



THE EMBODIED BRAIN: COMPUTATIONAL MECHANISMS OF INTEGRATED SENSORIMOTOR INTERACTIONS WITH A DYNAMIC ENVIRONMENT

EDITED BY: Mario Senden, Judith Peters, Florian Röhrbein, Rainer Goebel
and Gustavo Deco

PUBLISHED IN: Frontiers in Computational Neuroscience, Frontiers in Neurorobotics
and Frontiers in Systems Neuroscience





frontiers

Frontiers eBook Copyright Statement

The copyright in the text of individual articles in this eBook is the property of their respective authors or their respective institutions or funders. The copyright in graphics and images within each article may be subject to copyright of other parties. In both cases this is subject to a license granted to Frontiers.

The compilation of articles constituting this eBook is the property of Frontiers.

Each article within this eBook, and the eBook itself, are published under the most recent version of the Creative Commons CC-BY licence.

The version current at the date of publication of this eBook is CC-BY 4.0. If the CC-BY licence is updated, the licence granted by Frontiers is automatically updated to the new version.

When exercising any right under the CC-BY licence, Frontiers must be attributed as the original publisher of the article or eBook, as applicable.

Authors have the responsibility of ensuring that any graphics or other materials which are the property of others may be included in the CC-BY licence, but this should be checked before relying on the CC-BY licence to reproduce those materials. Any copyright notices relating to those materials must be complied with.

Copyright and source acknowledgement notices may not be removed and must be displayed in any copy, derivative work or partial copy which includes the elements in question.

All copyright, and all rights therein, are protected by national and international copyright laws. The above represents a summary only. For further information please read Frontiers' Conditions for Website Use and Copyright Statement, and the applicable CC-BY licence.

ISSN 1664-8714

ISBN 978-2-88963-910-6

DOI 10.3389/978-2-88963-910-6

About Frontiers

Frontiers is more than just an open-access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers Journal Series

The Frontiers Journal Series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the Frontiers Journal Series operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to Quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews.

Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the Frontiers Journals Series: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area! Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers Editorial Office: researchtopics@frontiersin.org

THE EMBODIED BRAIN: COMPUTATIONAL MECHANISMS OF INTEGRATED SENSORIMOTOR INTERACTIONS WITH A DYNAMIC ENVIRONMENT

Topic Editors:

Mario Senden, Maastricht University, Netherlands

Judith Peters, Maastricht University, Netherlands

Florian Röhrbein, Independent researcher, Germany

Rainer Goebel, Maastricht University, Netherlands

Gustavo Deco, Pompeu Fabra University, Spain

Citation: Senden, M., Peters, J., Röhrbein, F., Goebel, R., Deco, G., eds. (2020). The Embodied Brain: Computational Mechanisms of Integrated Sensorimotor Interactions with a Dynamic Environment. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-88963-910-6

Table of Contents

- 05 Editorial: The Embodied Brain: Computational Mechanisms of Integrated Sensorimotor Interactions With a Dynamic Environment**
Mario Senden, Judith Peters, Florian Röhrbein, Gustavo Deco and Rainer Goebel
- 08 Spike Timing Neural Model of Motion Perception and Decision Making**
Petia D. Koprinkova-Hristova, Nadejda Bocheva, Simona Nedelcheva and Mirsolava Stefanova
- 15 Running Large-Scale Simulations on the Neurorobotics Platform to Understand Vision – The Case of Visual Crowding**
Alban Bornet, Jacques Kaiser, Alexander Kroner, Egidio Falotico, Alessandro Ambrosano, Kepa Cantero, Michael H. Herzog and Gregory Francis
- 29 Complex Electroresponsive Dynamics in Olivocerebellar Neurons Represented With Extended-Generalized Leaky Integrate and Fire Models**
Alice Geminiani, Claudia Casellato, Egidio D'Angelo and Alessandra Pedrocchi
- 41 Corrigendum: Complex Electroresponsive Dynamics in Olivocerebellar Neurons Represented With Extended-Generalized Leaky Integrate and Fire Models**
Alice Geminiani, Claudia Casellato, Egidio D'Angelo and Alessandra Pedrocchi
- 42 The Embodied Brain of SOVEREIGN2: From Space-Variant Conscious Percepts During Visual Search and Navigation to Learning Invariant Object Categories and Cognitive-Emotional Plans for Acquiring Valued Goals**
Stephen Grossberg
- 74 The Energy Homeostasis Principle: Neuronal Energy Regulation Drives Local Network Dynamics Generating Behavior**
Rodrigo C. Vergara, Sebastián Jaramillo-Riveri, Alejandro Luarte, Cristóbal Moënné-Loccoz, Rómulo Fuentes, Andrés Couve and Pedro E. Maldonado
- 92 A Closed-Loop Toolchain for Neural Network Simulations of Learning Autonomous Agents**
Jakob Jordan, Philipp Weidel and Abigail Morrison
- 103 A Biomimetic Control Method Increases the Adaptability of a Humanoid Robot Acting in a Dynamic Environment**
Marie Claire Capolei, Emmanouil Angelidis, Egidio Falotico, Henrik Hautop Lund and Silvia Tolu
- 121 Combining Evolutionary and Adaptive Control Strategies for Quadruped Robotic Locomotion**
Elisa Massi, Lorenzo Vannucci, Ugo Albanese, Marie Claire Capolei, Alexander Vandesompele, Gabriel Urbain, Angelo Maria Sabatini, Joni Dambre, Cecilia Laschi, Silvia Tolu and Egidio Falotico
- 140 Generating Pointing Motions for a Humanoid Robot by Combining Motor Primitives**
J. Camilo Vasquez Tieck, Tristan Schnell, Jacques Kaiser, Felix Mauch, Arne Roennau and Rüdiger Dillmann

- 149** *Response Dynamics in an Olivocerebellar Spiking Neural Network With Non-linear Neuron Properties*
Alice Geminiani, Alessandra Pedrocchi, Egidio D'Angelo and Claudia Casellato
- 164** *Embodied Synaptic Plasticity With Online Reinforcement Learning*
Jacques Kaiser, Michael Hoff, Andreas Konle, J. Camilo Vasquez Tieck, David Kappel, Daniel Reichard, Anand Subramoney, Robert Legenstein, Arne Roennau, Wolfgang Maass and Rüdiger Dillmann
- 175** *Autonomous Sequence Generation for a Neural Dynamic Robot: Scene Perception, Serial Order, and Object-Oriented Movement*
Jan Tekülve, Adrien Fois, Yulia Sandamirskaya and Gregor Schöner
- 193** *Experimental and Computational Study on Motor Control and Recovery After Stroke: Toward a Constructive Loop Between Experimental and Virtual Embodied Neuroscience*
Anna Letizia Allegra Mascaro, Egidio Falotico, Spase Petkoski, Maria Pasquini, Lorenzo Vannucci, Núria Tort-Colet, Emilia Conti, Francesco Resta, Cristina Spalletti, Shravan Tata Ramalingasetty, Axel von Arnim, Emanuele Formento, Emmanouil Angelidis, Camilla H. Blixhavn, Trygve B. Leergaard, Matteo Caleo, Alain Destexhe, Auke Ijspeert, Silvestro Micera, Cecilia Laschi, Viktor Jirsa, Marc-Oliver Gewaltig and Francesco S. Pavone



Editorial: The Embodied Brain: Computational Mechanisms of Integrated Sensorimotor Interactions With a Dynamic Environment

Mario Senden^{1,2*†}, Judith Peters^{1,2,3†}, Florian Röhrbein^{4†}, Gustavo Deco^{5,6} and Rainer Goebel^{1,2,3}

¹ Department of Cognitive Neuroscience, Faculty of Psychology and Neuroscience, Maastricht University, Maastricht, Netherlands, ² Maastricht Brain Imaging Center (M-BIC), Maastricht University, Maastricht, Netherlands, ³ Department of Vision and Cognition, Netherlands Institute for Neuroscience, Royal Netherlands Academy of Arts and Sciences (KNAW), Amsterdam, Netherlands, ⁴ Institut für Informatik VI, Technische Universität München, Munich, Germany, ⁵ Center for Brain and Cognition, Computational Neuroscience Group, Department of Information and Communication Technologies, Universitat Pompeu Fabra, Barcelona, Spain, ⁶ Institució Catalana de la Recerca i Estudis Avançats (ICREA), Universitat Pompeu Fabra, Barcelona, Spain

Keywords: sensorimotor integration, embodiment, neurorobotics, motor control, reinforcement learning and plasticity, neural computation

Editorial on the Research Topic

The Embodied Brain: Computational Mechanisms of Integrated Sensorimotor Interactions With a Dynamic Environment

OPEN ACCESS

Edited and reviewed by:

Si Wu,
Peking University, China

*Correspondence:

Mario Senden
mario.senden@maastrichtuniversity.nl

[†]These authors have contributed
equally to this work

*Present address:

Florian Röhrbein,
Alfred Kärcher SE Co. & KG,
Winnenden, Germany

Received: 09 May 2020

Accepted: 15 May 2020

Published: 18 June 2020

Citation:

Senden M, Peters J, Röhrbein F,
Deco G and Goebel R (2020) Editorial:
The Embodied Brain: Computational
Mechanisms of Integrated
Sensorimotor Interactions With a
Dynamic Environment.
Front. Comput. Neurosci. 14:53.
doi: 10.3389/fncom.2020.00053

The paradigm shift toward an action-oriented view (Engel et al., 2013) stresses that cognition permits meaningful interactions with a dynamic environment and cannot be reduced to thinking-related mental representations. Consequently, the emerging field of embodied neuroscience has been inspired by recent achievements in robotics. At the same time, the fields of robotics and artificial intelligence increasingly turn to neuroscience to utilize insights on the neural underpinnings of sensorimotor interactions and embodied cognition.

As contribution to this integration of computational neuroscience, artificial intelligence, robotics and neurobiology, this Research Topic provides an overview of recent advances in sensorimotor integration and embodied cognition from a multidisciplinary perspective. A total of nine contributions present important scientific insights into embodied sensorimotor systems while another four contributions present comprehensive frameworks and toolchains that support the interdisciplinary study of embodied agents.

EMBODIED SENSORIMOTOR SYSTEMS

Embodied agents need to be able to autonomously and adaptively interact with their environment. Grossberg presents a large-scale visuomotor architecture: the Self-Organizing, Vision, Expectation, Recognition, Emotion, Intelligent, Goal-oriented Navigation model (SOVEREIGN; Gnadt and Grossberg, 2008). This architecture consists of several sensory, motor and memory components and is able to perform motor sequences under different motivational states as well as to learn more efficient sequences in response to rewards. Grossberg reviews the SOVEREIGN architecture as well as advancements in the field over the past decade and presents an updated version of the architecture, SOVEREIGN2. SOVEREIGN2 incorporates resonant dynamics which allow new perceptual, cognitive and navigational properties to emerge.

One highly complex cognitive aspect of sensorimotor integration, involving the recruitment and concerted interplay among a large number of cortical and subcortical brain regions, is action selection. Koprinkova-Hristova et al. capture this complexity with a biologically plausible large-scale architecture able to generate eye movement decisions. This architecture, implemented as a hierarchical spiking neural network (SNN), consists of multiple layers including the retina, several thalamic nuclei as well as cortical regions along the dorsal stream from V1 to the lateral intraparietal cortex. When probed with stimuli mimicking optic flow patterns of forward self-motion, the model selects eye movements that correctly align its gaze with the direction of self-motion.

Tekülve et al. approach a sequential pointing task from the perspective of dynamic field theory (Schöner and Spencer, 2016). Their contribution presents a spiking neural network (SNN) architecture comprised of: a perceptual subnetwork able to create a working memory representation of the visual scene, a motor subnetwork able to generate movement commands for a robotic arm, and a cognitive subnetwork able to represent positions in a sequence as well as to initiate shifts between positions. This architecture allows a robot to memorize a sequence of distinct objects (presented by a human), and subsequently point at these objects for random spatial arrangements of these objects.

Another robotic agent able to perform pointing movements is presented by Tieck et al. They developed an SNN of the primary motor cortex that is able to adaptively combine motor primitives, a low-dimensional vocabulary of motor actions (Rizzolatti et al., 1988; Santello et al., 1998; Ciocarlie et al., 2007). A humanoid robot, utilizing this network, could successfully point at different targets marked on a plane.

The cerebellum is a key structure for sensorimotor control, as it coordinates voluntary movements through prediction and sensory feedback (Johansson and Westling, 1988; Wolpert and Flanagan, 2001; Xu-Wilson et al., 2009; Manto et al., 2012). Capolei et al. present a cerebellar microcircuit which, supplanted with a classic control method, allows for adaptive and robust control of a robot's movements as it balances a board with a rolling ball. The contributors show that cerebellar plasticity contributes to learning of dynamics related to arm-object interactions, and thus supports adaptive corrections to executed actions.

Inspired by the fact that evolution does not act on static, but rather on plastic systems learning from experiences in their environment, Massi et al. combine cerebellar plasticity with an evolutionary algorithm for optimizing quadruped robotic locomotion. Their control structure consists of a spinal central pattern generator (CPG) and a cerebellar adaptive controller able to learn online from feedback, while the parameters of the CPG are optimized offline via an evolutionary algorithm. Their results show that locomotion in a quadruped robot improves when the cerebellar controller is allowed to learn during evolutionary optimization as opposed to only afterwards. This suggests that parameters controlling the CPG need to be selected to benefit optimally from the adaptive controller.

The benefits conveyed by the cerebellum are intricately linked to its complex electroresponsive dynamics afforded by the plethora of cerebellar neuron types. Geminiani, Casellato et al.

present a novel point neuron model able to capture the dynamics of several neurons of the olivocerebellar circuit. Their Extended-Generalized Leaky Integrate-And-Fire (E-GLIF) neuron is optimized to capture the input-output relationships of Golgi cells, granule cells, Purkinje cells, molecular layer interneurons, deep cerebellar nuclei cells and inferior olivary cells. Geminiani, Pedrocchi, et al. utilize the E-GLIF to investigate how single neuron dynamics in conjunction with geometrical modular connectivity profiles shape the dynamics exhibited by cerebellar circuits involved in eye blink classical conditioning. Their simulations produce response properties in Purkinje and deep nuclei cells similar to those reported *in vivo* when relying on the E-GLIF neuron model, but not when using simplified point neuron models.

This highlights the significance of neuron dynamics. Importantly, these dynamics are not only affected by neuron morphology. Vergara et al. argue that the balance between energy income, expenditure and availability determine neural dynamics to a significant extent. Importantly, the contributors argue, the effects of these factors manifest themselves at all levels from molecular to behavioral. In arguing their case, the contributors provide a comprehensive overview of energy demands of neurons culminating in the proposal of the Energy Homeostasis Principle.

TOOLCHAINS AND FRAMEWORKS

Constructing state-of-the-art embodied systems that are able to intelligently interact with their environment in a closed loop, requires the development of large-scale architectures incorporating several structural as well as functional components. The immensity of this task requires a high degree of collaboration among research disciplines. In order to facilitate such collaboration, universally available platforms, toolchains, and shared frameworks are indispensable.

One platform aiming to facilitate integration of several structural and functional components into an embodied agent is the neurorobotics platform (NRP; Falotico et al., 2017). Bornet et al. show how the NRP enables to connect models of diverse visual functions, developed by different research groups, into a coherent architecture. Their architecture, consisting of a retina model, a saliency model and a segmentation model, is able to explain visual crowding phenomena.

Jordan et al. present a novel toolchain for reinforcement learning in autonomous agents controlled by biologically plausible neural networks. This toolchain connects benchmarking tools from machine learning with network simulators from computational neuroscience. The collaborators demonstrate the functionality of the toolchain by implementing a rate neuron actor critic architecture in the NEST simulator (Gewaltig and Diesmann, 2007) and training on the grid world and mountain car environments.

The possibility to perform online reward-based learning with spiking neurons in the NEST simulator is provided by the Synaptic Plasticity with Online Reinforcement learning (SPORE) framework (Kappel et al., 2015, 2017, 2018). Kaiser et al. utilize the NRP to evaluate SPORE for training robotic agents on a closed loop reaching and lane-following task. The contributors

show that SPORE was capable of learning shallow feedforward policies online for moderately difficult embodied tasks.

Mascaro et al. present an iterative loop between experiment and model simulation to refine and validate models with experimental data as well as adjust experiments based on simulations. The contributors demonstrate the feasibility of their iterative loop for two separate scenarios. In the first, the iterative loop allowed them to replicate the evolution of functional connectivity in the mouse brain after stroke using neural mass model simulations. In the second, the contributors integrated their iterative loop with the NRP to embody a spinal cord model of the mouse and were able to reproduce goal-directed forelimb movements. Such a framework that simulates all relevant components of an experimental study, facilitates the continuous integration of novel experimental results into model simulations. In turn, modeling results can contribute to ongoing improvements in experimental design.

CONCLUSION

Understanding how an embodied brain can meaningfully interact with its dynamic external environment while managing inner homeostatic requirements is a challenging task. Indeed, identifying the functional capacities that an embodied nervous system needs to implement, the physical

constraints it is subjected to as well as specifying representations, transformations and dynamics realizing these capacities requires input from computational neuroscientists, roboticists, machine learning experts, and neurobiologists. Contributions to this Research Topic reflect current advances in embodied action mechanisms across fields. However, for a comprehensive understanding of embodied cognition and its utilization in neurorobotics, it is essential that efforts become increasingly collaborative in the future. For this collaboration to be fruitful, support by an infrastructure enabling researchers to effectively integrate their empirical results and modeling efforts into large-scale closed-loop architectures will be indispensable. The frameworks and toolchains presented within the present Research Topic are an important step in that direction.

AUTHOR CONTRIBUTIONS

All authors acted as guest editors on the Research Topic. MS and JP wrote the Editorial.

ACKNOWLEDGMENTS

We thank all authors contributing with their work to this Research Topic. This work received funding from the Human Brain Project grant agreement no. 785907 (SGA2).

REFERENCES

- Ciocarlie, M., Goldfeder, C., and Allen, P. (2007). "Dexterous grasping via eigengrasps: a low-dimensional approach to a high-complexity problem" in *Robotics: Science and Systems*. Available online at: http://projects.csail.mit.edu/manipulation/rss07/paper_dexterous_grasping_via_eigengrasps_a_low_dimensional_approach_to_a_high_complexity_problem_ciocarlie.pdf
- Engel, A. K., Maye, A., Kurthen, M., and König, P. (2013). Where's the action? The pragmatic turn in cognitive science. *Trends Cognit. Sci.* 17, 202–209. doi: 10.1016/j.tics.2013.03.006
- Falotico, E., Vannucci, L., Ambrosano, A., Albanese, U., Ulbrich, S., Vasquez Tieck, J. C., et al. (2017). Connecting artificial brains to robots in a comprehensive simulation framework: the neurorobotics platform. *Front. Neurobot.* 11:2. doi: 10.3389/fnbot.2017.00002
- Gewaltig, M.-O., and Diesmann, M. (2007). NEST (NEural Simulation Tool). *Scholarpedia* 2:1430. doi: 10.4249/scholarpedia.1430
- Gnadt, W., and Grossberg, S. (2008). SOVEREIGN: An autonomous neural system for incrementally learning planned action sequences to navigate towards a rewarded goal. *Neural Netw.* 21, 699–758. doi: 10.1016/j.neunet.2007.09.016
- Johansson, R. S., and Westling, G. (1988). Programmed and triggered actions to rapid load changes during precision grip. *Exp. Brain Res.* 71, 72–86. doi: 10.1007/BF00247523
- Kappel, D., Habenschuss, S., Legenstein, R., and Maass, W. (2015). Network plasticity as bayesian inference. *PLoS Computat. Biol.* 11:e1004485. doi: 10.1371/journal.pcbi.1004485
- Kappel, D., Hoff, M., and Subramoney, A. (2017). *IGITUGraz/spore-nest-module: SPORE version 2.14.0 (version v2.14.0)*. Available online at: <https://zenodo.org/record/1043486>
- Kappel, D., Legenstein, R., Habenschuss, S., Hsieh, M., and Maass, W. (2018). A dynamic connectome supports the emergence of stable computational function of neural circuits through reward-based learning. *ENeuro* 5, 1–28. doi: 10.1523/ENEURO.0301-17.2018
- Manto, M., Bower, J. M., Conforto, A. B., Delgado-García, J. M., da Guarda, S. N. F., Gerwig, M., et al. (2012). Consensus paper: roles of the cerebellum in motor control—the diversity of ideas on cerebellar involvement in movement. *Cerebellum* 11, 457–487. doi: 10.1007/s12311-011-0331-9
- Rizzolatti, G., Camarda, R., Fogassi, L., Gentilucci, M., Luppino, G., and Matelli, M. (1988). Functional organization of inferior area 6 in the macaque monkey. *Exp. Brain Res.* 71, 491–507. doi: 10.1007/BF00248742
- Santello, M., Flanders, M., and Soechting, J. F. (1998). Postural hand synergies for tool use. *J. Neurosci.* 18, 10105–15. doi: 10.1523/JNEUROSCI.18-23-10105.1998
- Schöner, G., and Spencer, J. (2016). *Dynamic Thinking: A Primer on Dynamic Field Theory*. New York, NY: Oxford University Press. Available online at: https://books.google.co.in/books/about/Dynamic_Thinking.html?id=NA0HCwAAQBAJ&printsec=frontcover&source=kp_read_button&redir_esc=y#v=onepage&q&f=false
- Wolpert, D. M., and Flanagan, J. R. (2001). Motor prediction. *Curr. Biol.* 11, R729–R732. doi: 10.1016/S0960-9822(01)00432-8
- Xu-Wilson, M., Chen-Harris, H., Zee, D. S., and Shadmehr, R. (2009). Cerebellar contributions to adaptive control of saccades in humans. *J. Neurosci.* 29, 12930–12939. doi: 10.1523/JNEUROSCI.3115-09.2009

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2020 Senden, Peters, Röhrbein, Deco and Goebel. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Spike Timing Neural Model of Motion Perception and Decision Making

Petia D. Koprinkova-Hristova^{1*}, Nadejda Bocheva^{2*}, Simona Nedelcheva¹ and Mirsolava Stefanova²

¹ Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Sofia, Bulgaria, ² Institute of Neurobiology, Bulgarian Academy of Sciences, Sofia, Bulgaria

The paper presents a hierarchical spike timing neural network model developed in NEST simulator aimed to reproduce human decision making in simplified simulated visual navigation tasks. It includes multiple layers starting from retina photoreceptors and retinal ganglion cells (RGC) via thalamic relay including lateral geniculate nucleus (LGN), thalamic reticular nucleus (TRN), and interneurons (IN) mediating connections to the higher brain areas—visual cortex (V1), middle temporal (MT), and medial superior temporal (MST) areas, involved in dorsal pathway processing of spatial and dynamic visual information. The last layer—lateral intraparietal cortex (LIP)—is responsible for decision making and organization of the subsequent motor response (saccade generation). We simulated two possible decision options having LIP layer with two sub-regions with mutual inhibitory connections whose increased firing rate corresponds to the perceptual decision about motor response—left or right saccade. Each stage of the model was tested by appropriately chosen stimuli corresponding to its selectivity to specific stimulus characteristics (orientation for V1, direction for MT, and expansion/contraction movement templates for MST, respectively). The overall model performance was tested with stimuli simulating optic flow patterns of forward self-motion on a linear trajectory to the left or to the right from straight ahead with a gaze in the direction of heading.

Keywords: visual perception, self-motion, spike timing neuron model, visual cortex, LGN, MT, MST, LIP

OPEN ACCESS

Edited by:

Mario Senden,
Maastricht University, Netherlands

Reviewed by:

Jan Lauwereyns,
Kyushu University, Japan
Daya Shankar Gupta,
Camden County College,
United States

*Correspondence:

Petia D. Koprinkova-Hristova
pkoprinkova@bas.bg
Nadejda Bocheva
nadya@percept.bas.bg

Received: 01 December 2018

Accepted: 18 March 2019

Published: 05 April 2019

Citation:

Koprinkova-Hristova PD, Bocheva N,
Nedelcheva S and Stefanova M (2019)
Spike Timing Neural Model of Motion
Perception and Decision Making.
Front. Comput. Neurosci. 13:20.
doi: 10.3389/fncom.2019.00020

INTRODUCTION

Vision has to encode and interpret in real time the complex, ambiguous, and dynamic information from the environment in order to ensure successive interaction with it. In the process of evolution, in the mammalian brain have emerged areas with a specific type of functionality that can be regarded as a hierarchical structure processing the visual input. The incoming light is initially converted in the retina into electrical signal by retinal ganglion cells (RGC), passed through the relay station—lateral geniculate nucleus (LGN) and thalamic reticular nucleus (TRN)—to the primary visual cortex (V1) where the visual information splits in two parallel pathways involved in encoding spatial layout and motion (dorsal) and shape (ventral) information. Motion information encoding and interpretation pose serious challenges due to its different sources (self-motion, object motion, or eye movements), the need to integrate local measurements in order to resolve the ambiguities in the incoming dynamic stream of information, but also the need to segregate the signals coming from different objects. The motion information processing is

performed predominantly by the middle temporal area (MT) that encodes the speed and direction of the moving objects and the medial superior temporal area (MST) that extracts information about the self-motion of the observer.

Most of the existing motion information processing models are restricted to the interactions between the areas in the dorsal pathway: V1 and MT (e.g., Simoncelli and Heeger, 1998; Bayerl and Neumann, 2004; Bayerl, 2005; Chessa et al., 2016), V1, MT, and MST (Raudies et al., 2012) or MT and MST (Grossberg et al., 1999; Perrone, 2012). Many models consider only the feedforward interactions (e.g., Simoncelli and Heeger, 1998; Solari et al., 2015) disregarding the feedback connectivity; others employ rate-based equations (e.g., Grossberg et al., 2001; Raudies and Neumann, 2010) considering an average number of spikes in a population of neurons.

Here we present spike-timing neural network as an attempt to simulate realistically the interactions between all described processing stages of encoding of dynamic visual information in the human brain. To take into account the process of decision making based on perceived visual information and the preparation of a saccade to the desired location, we included the lateral intraparietal area (LIP) as the output layer. The model behavior was tested with simplified visual stimuli mimicking self-motion with gaze fixed, considering its output as a decision for saccade toward the determined heading direction.

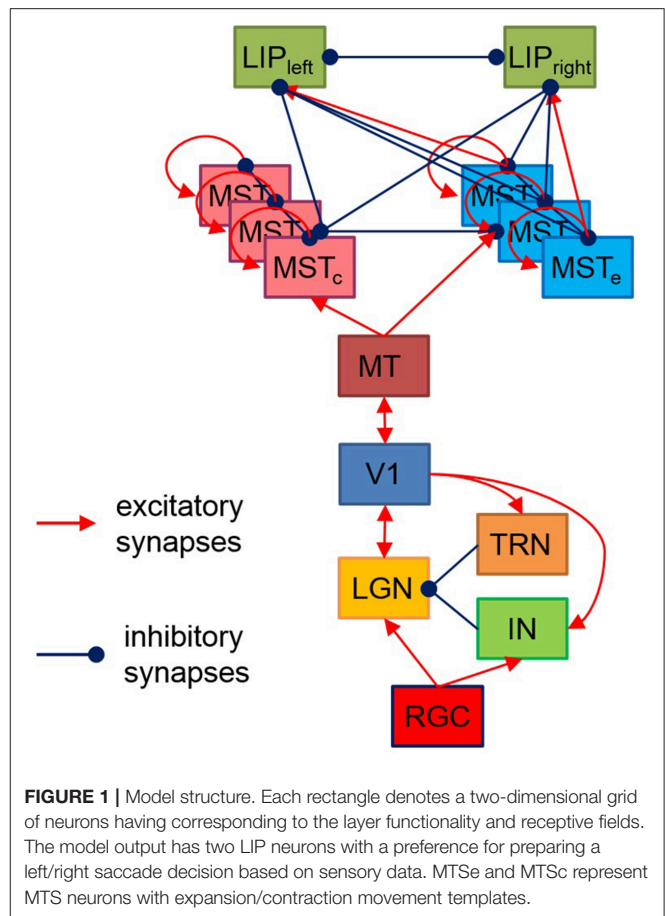
The model is implemented using NEST 2.12.0 simulator (Kunkel et al., 2017).

The paper is organized as follows: Section Model Structure describes briefly the overall model structure; Section Simulation Results reports results from its performance testing; Section Discussion presents a brief discussion of the model limitations and the directions of future work.

MODEL STRUCTURE

The proposed here hierarchical model, shown on **Figure 1**, is based on the available data about brain structures playing a role in visual motion information processing and perceptual decision making, as well as their connectivity. Each layer consists of neurons positioned in a regular two-dimensional grid. The receptive field of each neuron depends both on the function of the layer it belongs to and on its spatial position within its layer.

The reaction of RGC to luminosity changes is simulated by a convolution of a spatiotemporal filter with the images falling on the retina, following models from Troyer et al. (1998) and Kremkow et al. (2016). Its spatial component has a circular shape modeled by a difference of two Gaussians (DOG) while the temporal component has a bi-phasic profile determined by the difference of two Gamma functions. The model contains two layers of ON and OFF RGC and their corresponding LGN and IN/TRN neurons, having identical relative to visual scene positions and opposite [“on-center off-surround” (ON) and “off-center on-surround” (OFF)] receptive fields placed in reverse order like in Kremkow et al. (2016). Each layer consists of totally 400 neurons, positioned on 20×20 grid.



The continuous current generated by RGC is injected into LGN and IN via one-to-one connections. The structure of direct excitatory synaptic feedforward connectivity between LGN and V1 is also adopted from Kremkow et al. (2016). LGN also receives inhibitory feedback from V1 via IN and TRN according to (Ghodratia et al., 2017).

As in Kremkow et al. (2016), the neurons in V1 are separated into four groups—two exciting and two inhibiting, having a ratio of 4/1 exciting/inhibiting neurons (400/100 in our model) and connected via corresponding excitatory and inhibitory lateral connections. All exciting neurons are positioned at 20×20 grid while the 10×10 inhibiting neurons are dispersed among them. Being orientation sensitive, V1 neurons have elongated receptive fields defined by Gabor probability function as in Nedelcheva and Koprinkova-Hristova (2019). The “pinwheel structure” of the spatiotemporal maps of the orientations and phases of V1 neurons receptive fields was generated using a relatively new and easily implemented model (Sadeh and Rotter, 2014). An example of V1 orientation map (Nedelcheva and Koprinkova-Hristova, 2019) for a spatial frequency λ of the generating grating stimulus is shown in **Figure 2A**. Lateral connections in V1 are determined by Gabor correlations between the positions, phases, and orientations of each pair of neurons. As in Kremkow et al. (2016), neurons from inhibitory populations connect preferentially to neurons having a receptive field phase difference

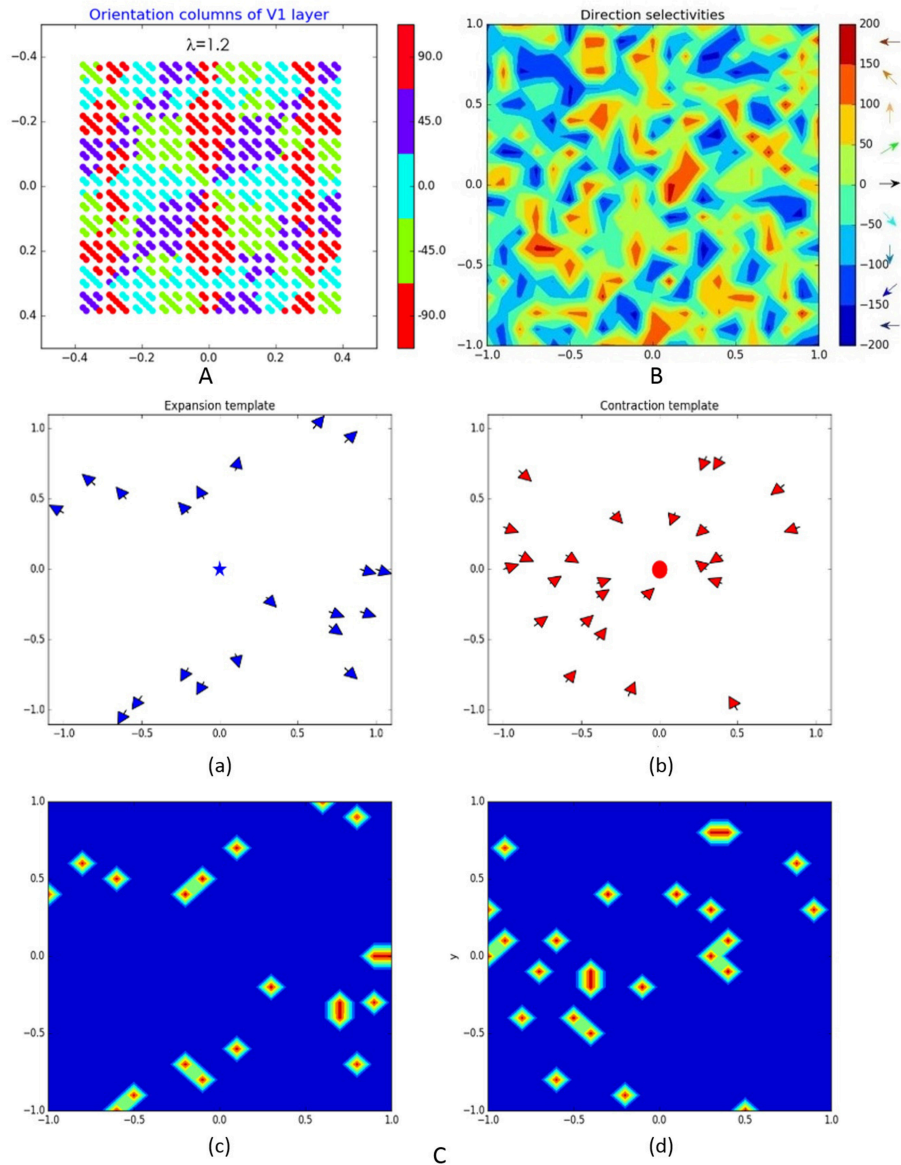


FIGURE 2 | Representation of some of the model layers and their connections: **(A)** orientation columns of V1 layer; **(B)** direction selectivity of MT layer; **(C)** expansion (a) and contraction (b) binary patterns $\mathbf{T}(\delta)$ of MT-MST connections (blue star and red dot denote the expansion/contraction focal points while the arrows show the direction selectivity of MT cells eligible to be connected to corresponding MST pattern) and corresponding to them connection templates $\mathbf{T}_{e(c)}$ on (c,d).

of around 180° . In our model, the frequencies, and standard deviations of Gabor filters for lateral connections were chosen so that all neurons in the layer have approximately circular receptive fields.

MT has identical to V1 size and structure and its lateral connections are designed in the same way while the connections from V1 cells depend on the angle φ_{ij} between the orientation preferences of each two cells like in Escobar et al. (2009):

$$w_{ij} = \begin{cases} k_c w_{cs} (x_i^{MT} - x_j^{V1}, y_i^{MT} - y_j^{V1}) \cos \varphi_{ij}, & 0 \leq \varphi_{ij} \leq \frac{\pi}{2} \\ 0, & \frac{\pi}{2} < \varphi_{ij} < \pi \end{cases}$$

Here k_c is amplification factor and w_{cs} is weight factor associated with the MT neuron receptive field, modeled as DOG function:

$$w_{cs} (x_i^{MT} - x_j^{V1}, y_i^{MT} - y_j^{V1}) = \frac{a_c e^{-\frac{\sqrt{(x_i^{MT} - x_j^{V1})^2 + (y_i^{MT} - y_j^{V1})^2}}{\sigma_c^2}}}{\sigma_c^2} - \frac{a_s e^{-\frac{\sqrt{(x_i^{MT} - x_j^{V1})^2 + (y_i^{MT} - y_j^{V1})^2}}{\sigma_s^2}}}{\sigma_s^2}$$

where a_c and a_s are the center and surround weights and σ_c and σ_s are the corresponding standard deviations. The orientation

and phase maps of this layer were generated in the same way as those of V1. An example of direction selectivity map of MT is shown on **Figure 2B**.

The MST consist of two layers, each one containing 400 neurons positioned on 20×20 grid, sensitive to expansion and contraction movement patterns, respectively, like in Layton and Fajen (2017). Each MST cell has assigned an expansion/contraction connection template $T_{e(c)}$ having a circular shape with width d and focal point $(x_{e(c)}, y_{e(c)})$ at MT as follows:

$$T_{e(c)}(x_{e(c)}, y_{e(c)}, x_{MT}, y_{MT}) = T(\delta) e^{-d((x_{e(c)} - x_{MT})^2 + (y_{e(c)} - y_{MT})^2)}$$

$$\delta = \arctg \frac{y_{e(c)} - y_{MT}}{x_{e(c)} - x_{MT}}$$

Here δ is the radial template angle determined by the position of each MT cell (x_{MT}, y_{MT}) and the given pattern expansion/contraction focal point. The binary pattern variable $T(\delta)$ is non-zero only if the corresponding MT cell has direction preference toward/against the contraction/expansion center of MST. **Figure 2C** shows examples of MT cells (with direction selectivity presented by arrows at corresponding positions) that are eligible for connection to corresponding expansion/contraction MST cells having focal points marked by blue star and red dot [(a) and (b)] and the corresponding connection templates [(c) and (d)].

The MST neurons have on-center receptive fields with standard deviation σ . Each MST neuron collects inputs from MT

cells corresponding to its pattern template as follows:

$$w_{e(c)}(x_{MT}, y_{MT}, x_{MST}, y_{MST}) = \frac{T_{e(c)}(x_{e(c)}, y_{e(c)}, x_{MT}, y_{MT})}{\frac{e^{-\frac{(x_{MT} - x_{MST})^2 + (y_{MT} - y_{MST})^2}{2\sigma_I^2}}}{\sqrt{2\pi\sigma_I^2}}}$$

Both layers have intra- and interlayer excitatory/inhibitory recurrent connections between cells having similar/different sensitivity as shown on **Figure 1**.

These lateral connections are determined based on neurons' positions and template similarities. All neurons have Gaussian receptive fields. Connections within expansion/contraction layers are excitatory or inhibitory in dependence on their focal points similarity as follows:

$$w_{e(c)}^{intra}(x_{MST}^1, y_{MST}^1, x_{MST}^2, y_{MST}^2) = \begin{cases} +e^{-\frac{(x_{MST}^1 - x_{MST}^2)^2 + (y_{MST}^1 - y_{MST}^2)^2}{2\sigma_{Is}^2}}, & \text{if } x_{e(c)}^1 = x_{e(c)}^2 \text{ and } y_{e(c)}^1 = y_{e(c)}^2 \\ -e^{-\frac{(x_{MST}^1 - x_{MST}^2)^2 + (y_{MST}^1 - y_{MST}^2)^2}{2\sigma_{Is}^2}}, & \text{otherwise} \end{cases}$$

Connections between expansion and contraction layers are all inhibitory and depend both on similarities of their positions and focal points as follows:

$$w_{e(c)}^{c(e)}(x_{MST}^e, y_{MST}^e, x_{MST}^c, y_{MST}^c) = -e^{-\frac{(x_{MST}^e - x_{MST}^c)^2 + (y_{MST}^e - y_{MST}^c)^2}{2\sigma_{Is}^2}} e^{-\frac{(x_{e(c)}^e - x_{e(c)}^c)^2 + (y_{e(c)}^e - y_{e(c)}^c)^2}{2\sigma_{Is}^2}}$$

In present work, we used only three focal points having identical vertical positions $y_{e(c)} = 0$.

Since our model aims to decide whether the expansion center of a moving dot stimulus is left or right from the stimulus center, here we proposed a task-dependent design of excitatory/inhibitory connections from MST expansion/contraction layers to the two LIP sub-regions whose increased firing rate corresponds to two taken decisions for two alternative motor responses—eye movement to the left or to the right. Both LIP areas are modeled by two neurons receiving excitatory input from MST expansion layer neurons having focal points corresponding to their decision responses (left or right) and inhibitory input from all other MST neurons. There are also lateral inhibitory connections between both LIP areas (**Figure 1**).

For the neurons in LGN conductance-based leaky integrate-and-fire neuron model as in Casti et al. (2008) (iaf_chxk_2008 in NEST) was adopted. For the rest of neurons, leaky integrate-and-fire model with exponential shaped postsynaptic currents according to Tsodyks et al. (2000) (iaf_psc_exp in NEST) was used. All connection parameters are the same as in the cited literature sources.

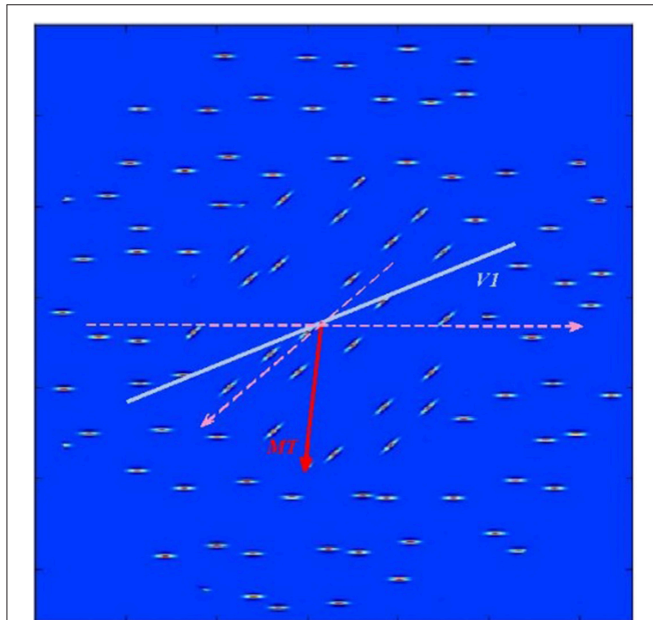


FIGURE 3 | Test stimulus consisting of horizontal and diagonal bars moving parallel to the bar orientations in each of the two stimulus regions as shown by dashed pink lines). The blue thick line shows estimated in V1 layer average orientation of the stimulus. The red arrow points toward estimated in MT layer average direction of bar movement within the stimulus.

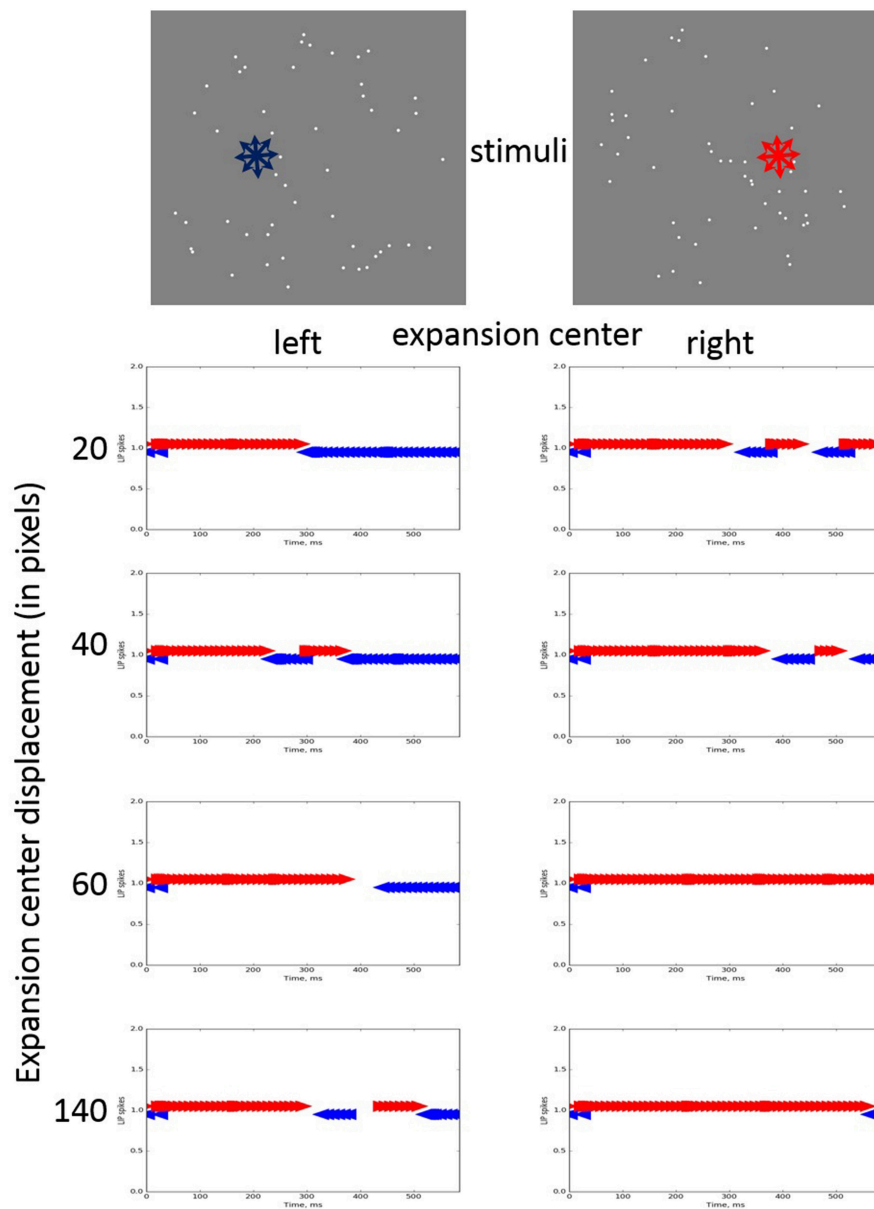


FIGURE 4 | LIP neurons spikes induced by moving dot stimuli (on the top) having expansion centers (denoted by stars indicating the predominant directions of motion of the dots) with varying displacements to the left or to the right from the screen center. The arrows pointing to the left and right denote the corresponding neuron spikes (blue for the left and red for the right, respectively).

SIMULATION RESULTS

In our previous work (Nedelcheva and Koprinkova-Hristova, 2019) we tested orientation selectivity of V1 in order to tune parameters of receptive fields of both LGN and V1 and the spatial frequency of V1 orientation columns using moving bar stimuli with two orientations. In Koprinkova-Hristova et al. (2018) we demonstrated that feedback inhibitory connections from V1 to LGN via TRN/IN modulates V1 neurons selectivity.

Further, we tested responses of MT using a stimulus composed of horizontal and diagonal bars moving with equal speed along different directions. To evaluate model responses, the

vector-averaged population decoding of V1, and MT was determined as in (Webb et al., 2010):

$$OR_{est} = \arctg \frac{\sum_i n_i \sin \theta_i}{\sum_i n_i \cos \theta_i}$$

where n_i is the total number of spikes generated by neurons having sensitivity to i -th orientation/direction. Estimated orientation and direction of stimulus shown on **Figure 3** in V1 and MT were 50.83° and 93.26° and correspond approximately to the mean values of the underlying stimulus characteristics.

The overall model was tested using visual stimulation simulating an observer's motion on a linear trajectory with eyes fixed in the heading direction. The stimuli consisted of 50 moving dots (36 of which moved radially and 14 with random movement directions) having expansion centers left or right from the visual scene center. Each dot lasted for 100 ms after which it was re-positioned randomly preserving its motion direction. On every frame, only one-third of the dots changed position. Variations of stimuli having seven expansion center positions ranging from 0.67 to 4.67° of arc (20–140 pixels) to the left or to the right of the screen center were generated. A detailed description of the experiment and the results with human subjects are given in Bocheva et al. (2018).

Spike trains generated by both LIP neurons (left and right) in response to the stimuli with varying center displacements (in pixels) moving for a duration of 600 ms are presented on **Figure 4**.

The simulation data showed that in all cases after a period of uncertainty the firing rate in the LIP area corresponding to the correct expansion center position is higher. The moment when correct decision starts to prevail depends on the task difficulty, i.e., the displacement magnitude. The LIP neuron reaching the correct decision has a shorter period of uncertainty with length inversely proportional to the center displacement magnitude. We also observed asymmetrical behavior of left/right LIP areas: the right decision is taken faster while for the left the model needed 300–400 ms to switch to the correct decision for intermediate displacements and longer time for the largest one.

DISCUSSION

The model has several limitations. We have focused only on the dorsal pathway and disregarded the interactions between the two visual pathways. However, the stimulation we used for model testing does not require additional complication even though its performance might be better at the MT stage if the information about the motion boundaries between the two regions of the stimulus configuration were extracted and supplied by the ventral pathway. The model parameters are based predominately on the data published in the literature. They have to be additionally tuned to represent the human performance in behavioral experiments with the same type of stimuli, as those reported by Bocheva et al. (2018).

The simulation data were obtained for fixed stimulus duration and suggest that the correct choice is achieved in <600 ms. However, the human observers, especially the older ones, needed more time to make a response. Only about 10 percent of the responses were shorter than 600 ms and only 53.4% of these

short responses were correct. While this suggests that the model outperforms the observers in accuracy and speed and is more effective in integrating the spatial and temporal information than the human observers, it needs to be emphasized that the reaction time of the human observers contains also non-decision components that involve the preparation of the motor response. Indeed, our data show that the component of the reaction time not related to decision-making is on average 342 ms for the young age group, 520 ms for the middle aged and 825 ms for the elderly. This non-decision time could not be taken into account in the model as it simulates only the decision making based on the accumulation of sensory information. In the future, we will test the model for longer stimulus duration and implement an ability to make a choice after the stimulus extinction.

In spite of its limitations, our model reproduced certain characteristics of the behavioral data like the trend for increased response times with the decrease in expansion center displacement.

We need to emphasize also that more elaborated stimuli were used for model testing than the typically used gratings or random dot patterns with the supposition that if the model performs well with these stimuli, it will perform well with simpler stimuli as well. However, even though our stimuli are more complex than the typical ones, they are simplified versions of the stimulation experienced in natural conditions and tasks. Additional tests with a larger set of stimuli are needed in order to improve model behavior. This will allow adjusting model parameters so that they replicate the age differences in performance in different tasks in dynamic conditions. The involvement of other brain structures contributing to saccade programming is another direction in our future work.

AUTHOR CONTRIBUTIONS

PK-H and NB contributed conception and design of the study. NB and MS developed visual stimuli. PK-H and SN performed the programming of the model in NEST. NB wrote the introduction, stimulus description, and discussion sections of the manuscript. PK-H wrote the model description sections. All authors contributed to manuscript revision, read and approved the submitted version.

FUNDING

The reported work is supported by the project DN02/3/2016 Modeling of voluntary saccadic eye movements during decision making funded by the Bulgarian Science Fund.

REFERENCES

- Bayerl, P. (2005). *A Model of Visual Perception*. PhD Thesis, Ulm University, Germany.
- Bayerl, P., and Neumann, H. (2004). Disambiguating visual motion through contextual feedback modulation. *Neural Comput.* 16, 2041–2066. doi: 10.1162/0899766041732404
- Bocheva, N., Genova, B., and Stefanova, M. (2018). Drift diffusion modeling of response time in heading estimation based on motion and form cues. *Int. J. Biol. Biomed. Eng.* 12, 75–83.
- Casti, A., Hayot, F., Xiao, Y., and Kaplan, E. (2008). A simple model of retina-LGN transmission. *J. Comput. Neurosci.* 24, 235–252. doi: 10.1007/s10827-007-0053-7

- Chessa, M., Sabatini, S., and Solari, F. (2016). A systematic analysis of a V1-MT neural model for motion estimation. *Neurocomputing*. 173, 1811–1823. doi: 10.1016/j.neucom.2015.08.091
- Escobar, M.-J., Masson, G. S., Vieville, T., and Kornprobst, P. (2009). Action recognition using a bio-inspired feedforward spiking network. *Int. J. Comput. Vis.* 82, 284–301. doi: 10.1007/s11263-008-0201-1
- Ghodratia, M., Khaligh-Razavic, S.-M., and Lehky, S. R. (2017). Towards building a more complex view of the lateral geniculate nucleus: recent advances in understanding its role. *Progr. Neurobiol.* 156, 214–255. doi: 10.1016/j.pneurobio.2017.06.002
- Grossberg, S., Mingolla, E., and Pack, C. (1999). A neural model of motion processing and visual navigation by cortical area MST. *Cereb. Cortex* 9, 878–895. doi: 10.1093/cercor/9.8.878
- Grossberg, S., Mingolla, E., and Viswanathan, L. (2001). Neural dynamics of motion integration and segmentation within and across apertures. *Vision Res.* 41, 2521–2553. doi: 10.1016/S0042-6989(01)00131-6
- Koprinkova-Hristova, P., Bocheva, N., and Nedelcheva, S. (2018). “Investigation of feedback connections effect of a spike timing neural network model of early visual system,” in *Innovations in Intelligent Systems and Applications (INISTA)*, 3–5 July 2018. Thessaloniki. doi: 10.1109/INISTA.2018.8466292
- Kremkow, J., Perrinet, L. U., Monier, C., Alonso, J.-M., Aertsen, A., Fregnac, Y., et al. (2016). Push-pull receptive field organization and synaptic depression: mechanisms for reliably encoding naturalistic stimuli in V1. *Front. Neural Circuits* 10:37. doi: 10.3389/fncir.2016.00037
- Kunkel, S., Morrison, A., Weidel, P., Eppler, J. M., Sinha, A., Schenck, W., et al. (2017). NEST 2.12.0. *Zenodo*. doi: 10.5281/zenodo.259534
- Layton, O. W., and Fajen, B. R. (2017). Possible role for recurrent interactions between expansion and contraction cells in MSTd during self-motion perception in dynamic environments. *J. Vis.* 17, 1–21. doi: 10.1167/17.5.5
- Nedelcheva, S., and Koprinkova-Hristova, P. (2019). “Orientation selectivity tuning of a spike timing neural network model of the first layer of the human visual cortex,” in *Advanced Computing in Industrial Mathematics. BGSIAM 2017. Studies in Computational Intelligence* 793, eds K. Georgiev, M. Todorov, and I. Georgiev (Cham: Springer), 291–303. doi: 10.1007/978-3-319-97277-0_24
- Perrone, J. (2012). A neural-based code for computing image velocity from small sets of middle temporal (MT/V5) neuron inputs. *J. Vis.* 12:1. doi: 10.1167/12.8.1
- Raudies, F., Mingolla, E., and Neumann, H. (2012). Active gaze control improves optic flow-based segmentation and steering. *PLoS ONE* 7:38446. doi: 10.1371/journal.pone.0038446
- Raudies, F., and Neumann, H. (2010). A neural model of the temporal dynamics of figure-ground segregation in motion perception. *Neural Networks* 23, 160–176. doi: 10.1016/j.neunet.2009.10.005
- Sadeh, S., and Rotter, S. (2014). Statistics and geometry of orientation selectivity in primary visual cortex. *Biol. Cybern.* 108, 631–653. doi: 10.1007/s00422-013-0576-0
- Simoncelli, E., and Heeger, D. (1998). A model of neuronal responses in visual area MT. *Vision Res.* 38, 743–761. doi: 10.1016/S0042-6989(97)00183-1
- Solari, F., Chessa, M., Medathati, K., and Kornprobst, P. (2015). What can we expect from a V1-MT feedforward architecture for optical flow estimation? *Signal Process. Image Commun.* 39, 342–354. doi: 10.1016/j.image.2015.04.006
- Troyer, T. W., Krukowski, A. E., Priebe, N. J., and Miller, K. D. (1998). Contrast invariant orientation tuning in cat visual cortex: thalamocortical input tuning and correlation-based intracortical connectivity. *J. Neurosci.* 18, 5908–5927. doi: 10.1523/JNEUROSCI.18-15-05908.1998
- Tsodyks, M., Uziel, A., and Henry, M. (2000). Synchrony generation in recurrent networks with frequency-dependent synapses. *J. Neurosci.* 20:RC50. doi: 10.1523/JNEUROSCI.20-01-j0003.2000
- Webb, B. S., Ledgeway, T., and McGraw, P. V. (2010). Relating spatial and temporal orientation pooling to population decoding solutions in human vision. *Vision Res.* 50, 2274–2283. doi: 10.1016/j.visres.2010.04.019

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Koprinkova-Hristova, Bocheva, Nedelcheva and Stefanova. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Running Large-Scale Simulations on the Neurobotics Platform to Understand Vision – The Case of Visual Crowding

Alban Bornet^{1*}, Jacques Kaiser², Alexander Kroner³, Egidio Falotico⁴,
Alessandro Ambrosano⁴, Kepa Cantero⁵, Michael H. Herzog¹ and Gregory Francis⁶

¹ Laboratory of Psychophysics, Brain Mind Institute, Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, ² FZI Research Center for Information Technology, Karlsruhe, Germany, ³ Department of Cognitive Neuroscience, Maastricht University, Maastricht, Netherlands, ⁴ The BioRobotics Institute, Scuola Superiore Sant'Anna, Pontedera, Italy, ⁵ Fortiss GmbH, Munich, Germany, ⁶ Department of Psychological Sciences, Purdue University, West Lafayette, IN, United States

OPEN ACCESS

Edited by:

Gustavo Deco,
Universitat Pompeu Fabra, Spain

Reviewed by:

Michael Beyeler,
University of Washington,
United States
Leslie Samuel Smith,
The University of Stirling,
United Kingdom

*Correspondence:

Alban Bornet
alban.bornet@epfl.ch

Received: 01 March 2019

Accepted: 14 May 2019

Published: 29 May 2019

Citation:

Bornet A, Kaiser J, Kroner A,
Falotico E, Ambrosano A, Cantero K,
Herzog MH and Francis G (2019)
Running Large-Scale Simulations on
the Neurobotics Platform
to Understand Vision – The Case
of Visual Crowding.
Front. Neurobot. 13:33.
doi: 10.3389/fnbot.2019.00033

Traditionally, human vision research has focused on specific paradigms and proposed models to explain very specific properties of visual perception. However, the complexity and scope of modern psychophysical paradigms undermine the success of this approach. For example, perception of an element strongly deteriorates when neighboring elements are presented in addition (visual crowding). As it was shown recently, the magnitude of deterioration depends not only on the directly neighboring elements but on almost all elements and their specific configuration. Hence, to fully explain human visual perception, one needs to take large parts of the visual field into account and combine all the aspects of vision that become relevant at such scale. These efforts require sophisticated and collaborative modeling. The Neurobotics Platform (NRP) of the Human Brain Project offers a unique opportunity to connect models of all sorts of visual functions, even those developed by different research groups, into a coherently functioning system. Here, we describe how we used the NRP to connect and simulate a segmentation model, a retina model, and a saliency model to explain complex results about visual perception. The combination of models highlights the versatility of the NRP and provides novel explanations for inward-outward anisotropy in visual crowding.

Keywords: visual crowding, neurobotics, modeling, large-scale simulation, vision

INTRODUCTION

Within the classic framework, vision starts with the analysis of basic features such as oriented edges. These basic features are then pooled along a feed-forward visual hierarchy to form more complex feature detectors until neurons respond to objects. A strength of modeling visual perception as a feed-forward process is that it breaks down the complexity of vision into mathematically treatable sub-problems. Whereas this approach has proven capable of explaining simple paradigms, it often fails when put in broader contexts (Oberfeld and Stahn, 2012; Clarke et al., 2014; Herzog et al., 2016; Overvliet and Sayim, 2016; Saarela et al., 2010). To fully understand vision, one needs to build complex models that process large parts of the visual field. At

such scale, many aspects of vision potentially become relevant. For example, it is well known that spatial resolution is highest in the fovea and strongly declines toward the periphery of the visual field (Daniel and Whitteridge, 1961; Cowey and Rolls, 1974). In addition, analysis of the visual field occurs by successive eye movements, which often brings the most salient aspects of the visual image into the center of fixation (Koch and Ullman, 1985; Itti et al., 1998). Moreover, the brain is also able to covertly attend to salient parts of the visual field and detect peripheral objects, without requiring eye movements (Eriksen and Hoffman, 1972; Posner, 1980; Wright and Ward, 2008). Hence, a full model of vision needs many functions that each requires sophisticated modeling, but these many functions are not easy to achieve within one research lab. To utilize different aspects of vision in one coherent system, we need a platform where many experts in the various subfields of vision can combine their models and test them in experimental conditions.

Efforts to simulate many models for different functions of perception as a single system can encounter many challenges, including the following.

Frameworks

Different models often come with very different computational frameworks. For example, one of the models might be a spiking neural network and another might be an algorithm involving a set of spatial convolutions. The models need a common simulation ground to talk to each other efficiently.

Emulation

Even if models coming from different research groups are simple, producing computer code to efficiently and reliably emulate models can be a daunting task. Few labs have the expertise needed to produce (or reproduce) models that address rather different parts of the visual system.

Analysis of the System

It is necessary, but often complicated, to determine the contribution of each model to the general output of the system. Moreover, competing models and hypotheses might be tested on the same data. To address these challenges, models should be treated as modules that can be easily removed from or added to the system. In the same vein, it is important to have a common visualization interface for the output of all simulated models.

Synchronization

It might be difficult to synchronize all the models in a common simulation. For example, one model might be a simple feed-forward input-output transformation, and another model might be a recurrent neural network that evolves through time even for a constant stimulus. It is important to make sure that interactions between those models are consistent with their states at every time-step.

Scalability

For many models, it is not straightforward to simulate the system efficiently and adapt the resource management to the workload of the simulation.

Reproducibility

It is important for scientists to be able to reproduce and extend simulation results. This means not only access to model code but also the ability to reproduce stimuli. Contextual elements such as lighting, distance to the stimulus, stimulus eccentricity or even the display screen, might matter in a complex model system. The simulated environment should ensure a common set of stimuli for all scientists.

The NRP, developed within the Human Brain Project, aims to address these challenges. The NRP provides an interface to study the interactions between an agent (a virtual robot) and a virtual environment through the simulation of a brain model (Falotico et al., 2017). The platform provides tools to enable the simulation of a full experiment, from sensory processing to motor execution. The simulated brain can comprise many functions, as long as the interactions between the various functions are defined in a specified python format (**Figure 1**). The main brain simulator of the platform is NEST (Gewaltig and Diesmann, 2007) but the platform also supports various mathematical libraries, such as TensorFlow (Abadi et al., 2016), to implement rate based neural networks. The virtual environment, the robot, and its sensors are simulated using Gazebo (Koenig and Howard, 2004). During the simulation, the platform provides an interactive visualization of the environment and of the output of all models that constitute the brain. Importantly, the user does not have to worry about the multiple synchronizations occurring during the simulation. The platform implements a closed loop that takes care of data exchanges and synchronizations between the virtual environment, the robot, and the brain models.

Here, we show that the NRP can easily combine different visual modules, even those programmed by different research groups. We show that these combined components can explain complex observations about visual perception, taking visual crowding as an example. We made the code publicly available at https://bitbucket.org/albornet/crowding_asymmetry_nrp. In the next section, we define visual crowding and the challenges that is

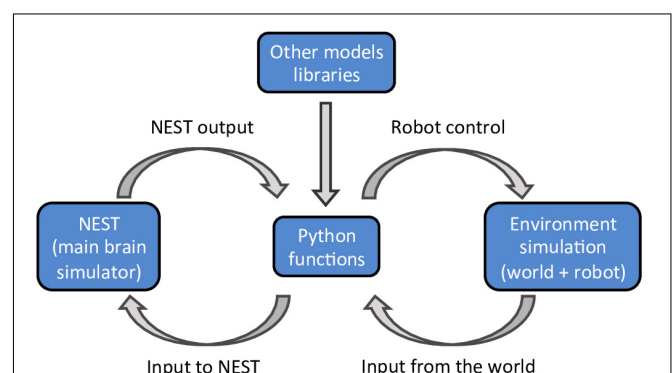


FIGURE 1 | A schematic outline of the NRP components. The platform can simulate a virtual environment (right) and a NEST brain model (left). Interactions between the brain and the virtual environment are set in python functions (center). These functions also take care of models that are not simulated in NEST, importing the required libraries as python packages.

addresses to vision research. Then, we describe the models that are combined in our visual system and their interactions. Next, we present the results of the simulation of the visual system that we built on the NRP. Finally, we discuss the results, followed by a conclusion.

THE CASE OF VISUAL CROWDING

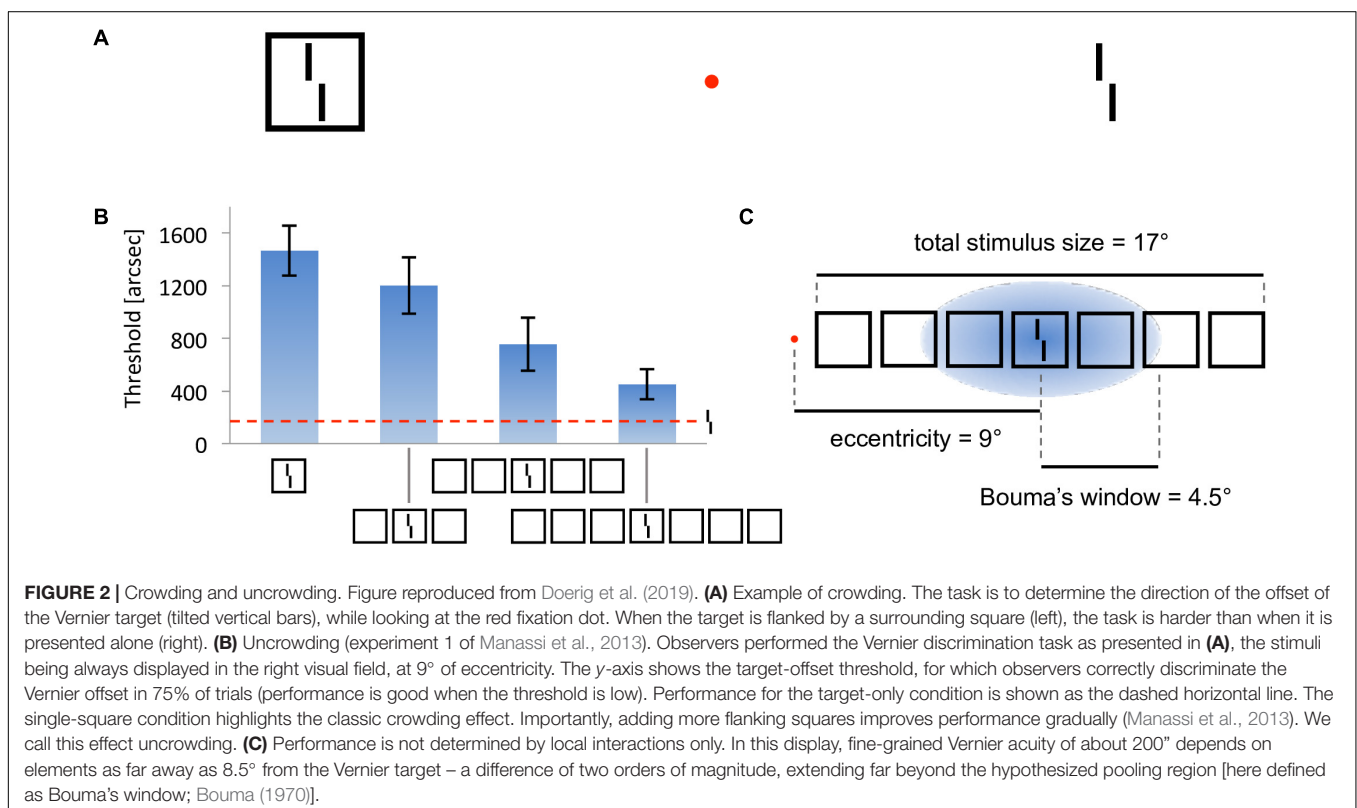
In crowding, perception of a target strongly deteriorates when it is presented together with surrounding elements (called flankers) that share similar features with the target (**Figure 2A**, Bouma, 1973). As for many other phenomena, crowding was traditionally explained by local mechanisms within the framework of object recognition (Wilson, 1997; Parkes et al., 2001; Pelli, 2008; Nandy and Tjan, 2012). In this view, crowding occurs when flanking elements are pooled with target information along the processing hierarchy. Pooling can explain crowding when a few flankers are present but fails to match human behavior when more flankers are presented. For example, pooling models predict that flankers beyond the pooling region should not influence performance on the target, and that adding flankers can only increase crowding. Both predictions have been shown to be wrong. Adding flankers up to a very large distance from the target can improve performance and even fully undo crowding (**Figures 2B–C**; Manassi et al., 2012, 2013). Another feature of crowding that remains unexplained by pooling models is inward-outward anisotropy, which is the tendency for flankers that lie between

the fixation point and the target to produce less crowding than remote flankers (**Figure 3**; Bouma, 1973; Petrov et al., 2007; Farzin et al., 2009; Petrov and Meleshkevich, 2011; Manassi et al., 2012).

Local models cannot explain these aspects of vision (Herzog and Manassi, 2015; Herzog et al., 2015; Manassi et al., 2015; Doerig et al., 2019). To fully explain crowding, one needs to take the spatial configuration of large parts of the visual field into account. Francis et al. (2017) recently explained crowding and uncrowding with a complex dynamical model that segments an input image into several distinct perceptual groups and computes illusory contours from the edges in the image. In the model, a group is defined by a set of edges that are linked by actual or illusory contours. Interference only occurs within each group, and the target is released from crowding if the flankers make a group on their own, as described in more detail below (**Figure 4**). However, the model does not generate inward-outward anisotropy, because it does not contain any source of asymmetry. To determine whether the grouping explanation can account for inward-outward anisotropy, we propose to incorporate the model in a more complex and realistic visual system, described in the next section.

MATERIALS AND METHODS

In this section, we describe the models that we connected, using the NRP, to explain inward-outward anisotropy in crowding. Then, we describe how the models interact with each other.



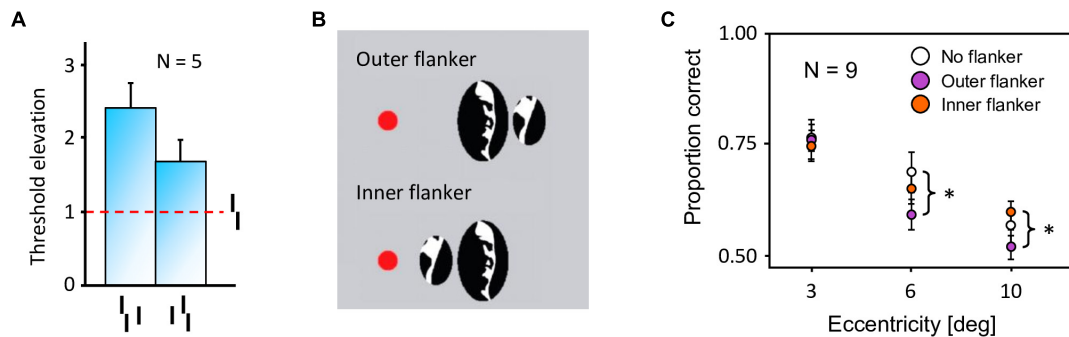


FIGURE 3 | Inward-outward anisotropy in visual crowding. **(A)** Inward-outward anisotropy in a Vernier discrimination task (experiment 1b of Manassi et al., 2012). Observers performed the Vernier discrimination task as presented in **Figure 2A**. The stimuli were always presented in the right visual field. The y-axis shows the target-offset threshold elevation, which is defined as the threshold of the condition divided by the threshold of the unflanked condition. Stronger crowding is observed for an outer flanker than for an inner flanker. **(B)** Stimuli used in the crowding paradigm of experiment 5 of Farzin et al. (2009), measuring inward-outward anisotropy with Mooney faces. The red dot is the fixation point. In this paradigm, the target Mooney face is shown either in the left or the right visual hemi-field, together with either an inner flanker, an outer flanker, or with no flanker and at different eccentricities of 3°, 6°, and 10° (one block per eccentricity and per flanker configuration). Observers were asked to discriminate an upright from an inverted target Mooney face (2-AFC discrimination task). **(C)** Data from experiment 5 of Farzin et al. (2009). Note that the y-axis is the proportion of correct discrimination, and that a high value means a good discrimination performance. The stars indicate significant differences between conditions. The amount of inward-outward anisotropy (how much the inner-flanker condition produces better performance than the outer-flanker condition) interacts with the stimulus eccentricity.

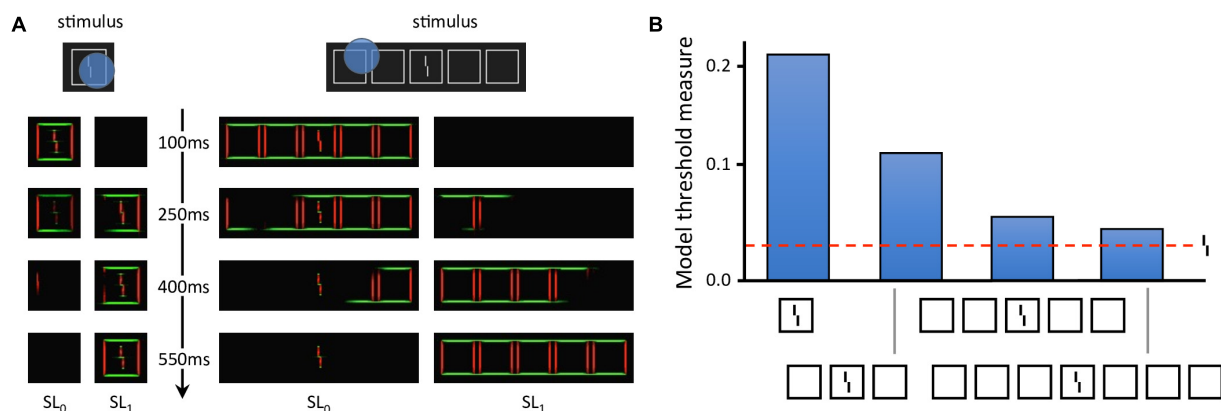


FIGURE 4 | Laminart model. **(A)** Activity in the segmentation model. The intensity of each pixel corresponds to the activity of an orientation-selective neuron encoding the stimulus as a local feature detector. The color of the pixel represents the orientation of the most active neuron at that location (red: vertical, green: horizontal). Visual elements linked together by illusory contours form a potential group. The blue circles mark example locations at which the segmentation dynamics are initiated after stimulus onset. From these locations, thanks to recurrent processing, segmentation propagates along connected (illusory or real) contours, until the stimulus is represented by several distinct neural populations, called segmentation layers (two here: SL_0 and SL_1). Each segmentation layer represents a perceptual group. Crowding is high if other elements are grouped in the same population as the Vernier target, and low if the target is alone. On the left, the flanker is hard to segment because of its proximity to the target. Across the trials, the selection signals often overlap with the whole stimulus, considered as a single group. Therefore, the flanker interferes with the target in most trials, and crowding is high. On the right, the flankers are linked by illusory contours and form a group that spans a large surface. In this case, the selection signal can easily hit the flankers group without hitting the target. The Vernier target thus ends up alone in its layer in most trials and crowding is low. **(B)** Threshold measurement from the segmentation model's output for all conditions of **Figure 2B**. The model threshold is measured by matching the output of the model to a target template over 20 segmentation trials, and then plotting the mean of the template match on a reversed axis [see Francis et al. (2017) for more details]. The segmentation model generates uncrowding and fits the behavioral data well.

The visual system is composed of the segmentation model of Francis et al. (2017), a retina model inspired by Ambrosano et al. (2016), and a saliency model, which is a simplified version of the model introduced by Kroner et al. (2019). These specific parts of human vision were chosen because the segmentation model already explains many features of visual crowding (**Figure 4**) and because retinal processing, as well as saliency computation, are potential sources of anisotropy for the segmentation output.

Indeed, the retina model is equipped with retinal magnification and the saliency model produces a central bias. In our simulated visual system, the visual environment is first processed by the retina model and its output is sent to the segmentation model. In parallel, saliency is computed as a 2-dimensional array which corresponds to the probabilities of making an eye movement to locations in the visual field. The current simulations do not contain any eye movement, but rather use the output of the

saliency model as a proxy for covert attention to determine the location where segmentation is initiated in the segmentation model. Finally, we measure crowding from the output of the segmentation model. We explain the model interactions and the crowding measurement process in more details further below.

Cortical Model for Segmentation

The Laminart model by Cao and Grossberg (2005) is a neural network that explains a wide variety of visual properties. A critical property is the creation of illusory contours between collinear lines. Francis et al. (2017) augmented the model with a segmentation mechanism, in which elements linked by contours (illusory or real) are grouped together by dedicated neural populations. The goal was to provide a two-stage model of crowding, with a strong grouping component: stimuli are first segmented into different groups and, subsequently, elements within a group interfere. After dynamical processing, different groups are represented by distinct neural populations. Crowding is determined by matching the model's output to a target template. Importantly, crowding is weak when the target is alone in its group (i.e., when the population representing the target does not also represent other elements) and strong otherwise.

The segmentation process is triggered by local selection signals that spread along connected contours (Figure 4). The location of the selection signals determines the output of the segmentation process. Uncrowding occurs when a selection signal touches a group of flankers without touching the target. In the original version of the model, the location of each selection signal followed a spatial distribution tuned to maximize successful segmentation of the target from the flanker in the crowding paradigm. This assumption follows the idea that, in psychophysical paradigms, an observer does the best job possible to succeed in the task. Here, we try a different approach by using the output of the saliency model to bias the location of the selection signal toward interesting regions of the visual field, as described further below.

Retina Model

Previous work has integrated a retina model as part of a neurobotic experiment in the NRP (Ambrosano et al., 2016) by using the COREM (Computational Retina Modeling framework; Martínez-Cañada et al., 2015, 2016). COREM is a set of building blocks that are often used to describe the behavior of the retina at different levels of detail. The system includes a variety of retina microcircuits, such as spatial integration filters, temporal linear filters, and static non-linearities. The retina model that is adopted for this work is an adaptation of a model of the X cells in the cat retina as described by Wohrer and Kornprobst (2009). We also use the COREM framework to simulate the retina model in the NRP. The model uses feedback loops between retinal layers to control contrast gain (Shapley and Victor, 1978). The X cells are chosen in this work because of their tonic and fine-grained response, as our paradigm involves highly detailed stimuli.

In addition, we include space variant Gaussian filters provided by COREM that mimic retinal magnification. Along the retinal layers, visual information is pooled with less spatial precision in the periphery than in foveal locations because the Gaussian

integration filters are broader with eccentricity. Finally, the output of the retina, i.e., the activity array of the ON- and OFF-centered ganglion cells, is distorted by a log-polar transform to mimic the magnification that results from the mapping of the retina neurons to the visual cortex. An example of the model's output is shown in Figure 5.

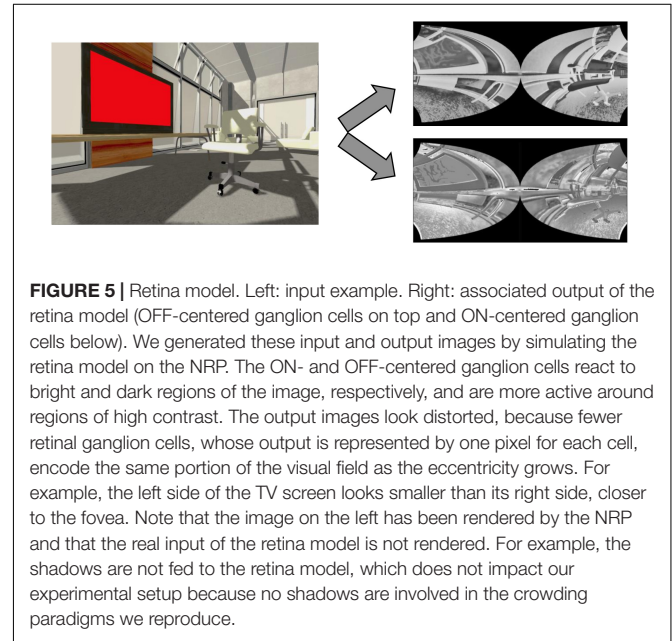


FIGURE 5 | Retina model. Left: input example. Right: associated output of the retina model (OFF-centered ganglion cells on top and ON-centered ganglion cells below). We generated these input and output images by simulating the retina model on the NRP. The ON- and OFF-centered ganglion cells react to bright and dark regions of the image, respectively, and are more active around regions of high contrast. The output images look distorted, because fewer retinal ganglion cells, whose output is represented by one pixel for each cell, encode the same portion of the visual field as the eccentricity grows. For example, the left side of the TV screen looks smaller than its right side, closer to the fovea. Note that the image on the left has been rendered by the NRP and that the real input of the retina model is not rendered. For example, the shadows are not fed to the retina model, which does not impact our experimental setup because no shadows are involved in the crowding paradigms we reproduce.

Saliency Model

Computational models of saliency aim to identify image regions that attract human eye movements when viewing complex natural scenes. The contribution of stimulus features to the allocation of overt attention can then best be captured in a task-free experimental scenario. As a model of saliency computation, we used a deep convolutional neural network, simulated in TensorFlow (Abadi et al., 2016), that automatically learns useful image representations to accurately predict empirical fixation density maps. Compared to early approaches based on biologically motivated feature channels, such as color, intensity, and orientation (Itti et al., 1998), the architecture extracts information at increasingly complex levels along its hierarchy.

The model is an encoder-decoder network that learned a non-linear mapping from raw images to topographic fixation maps. It constitutes a simplified version of the model introduced by Kroner et al. (2019), pruning the contextual layers to achieve computationally more efficient image processing. The VGG16 architecture (Simonyan and Zisserman, 2014), pre-trained on a visual classification task, serves as the model backbone to detect high-level features in the input space. Activation maps from the final convolutional encoding layer are then forwarded to the decoder, which restores the input resolution by applying bilinear up-sampling followed by a 3×3 convolution repeatedly. The task of saliency prediction is defined in a probabilistic framework and therefore aims to minimize the statistical distance between

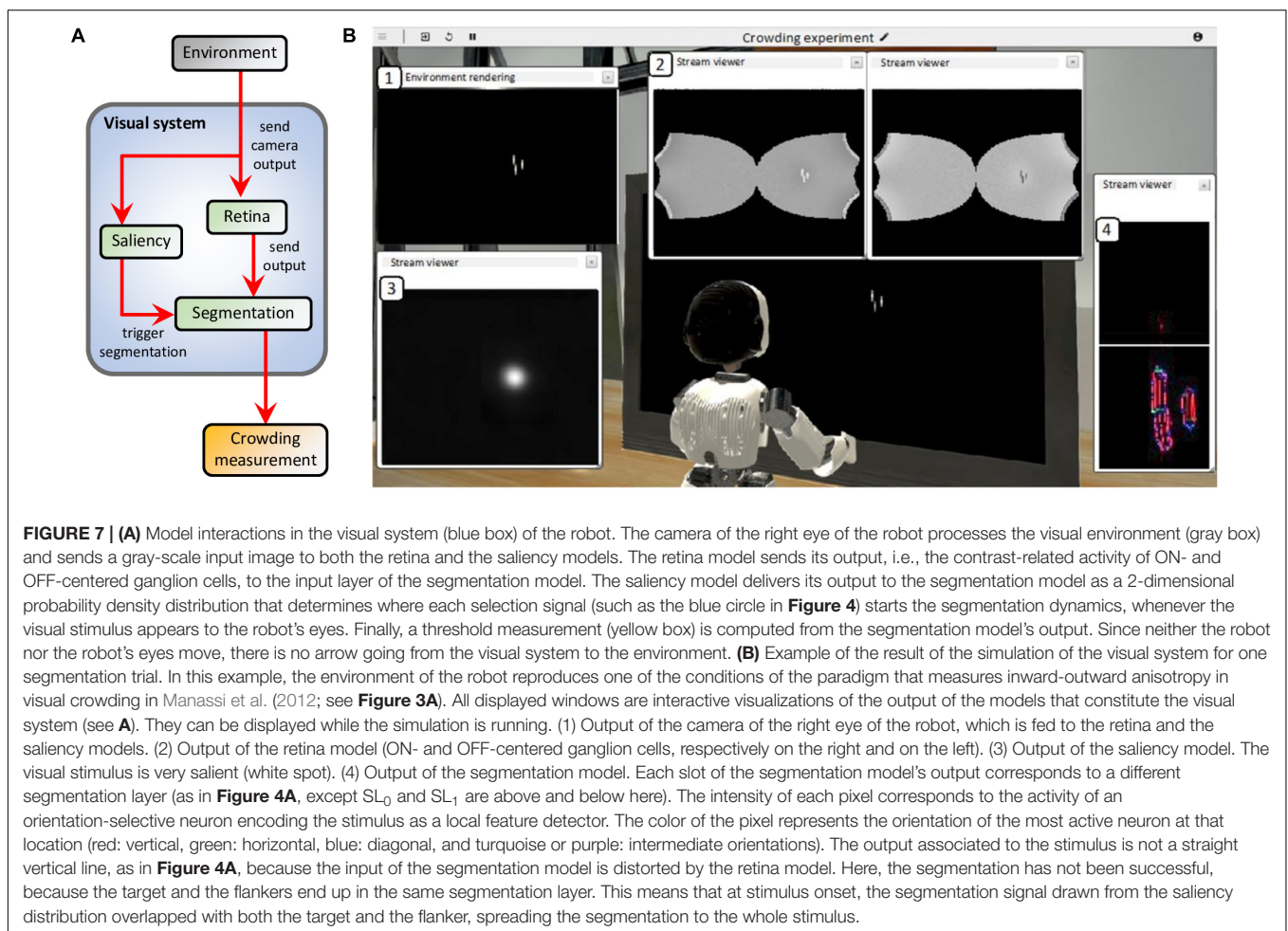
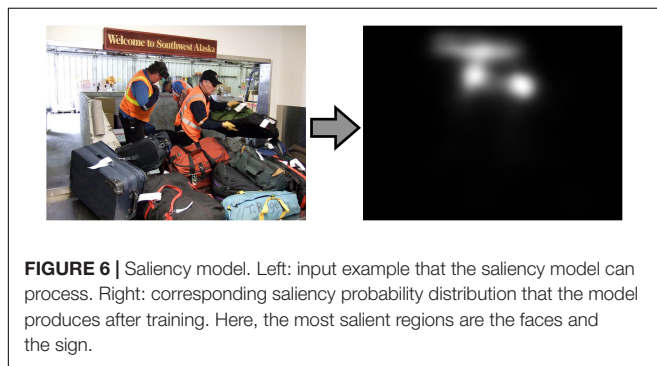
the estimated distribution and the ground truth. The model we used in this work was trained on the large-scale SALICON data set (Jiang et al., 2015), used as a proxy for eye tracking data. After training, the model produces a saliency map for any input image, such as in **Figure 6**. In our visual system, the saliency model output determines where the segmentation model selects objects of interest. The local selection signals that trigger segmentation in the model follow the saliency output as a probability density distribution. Although the saliency

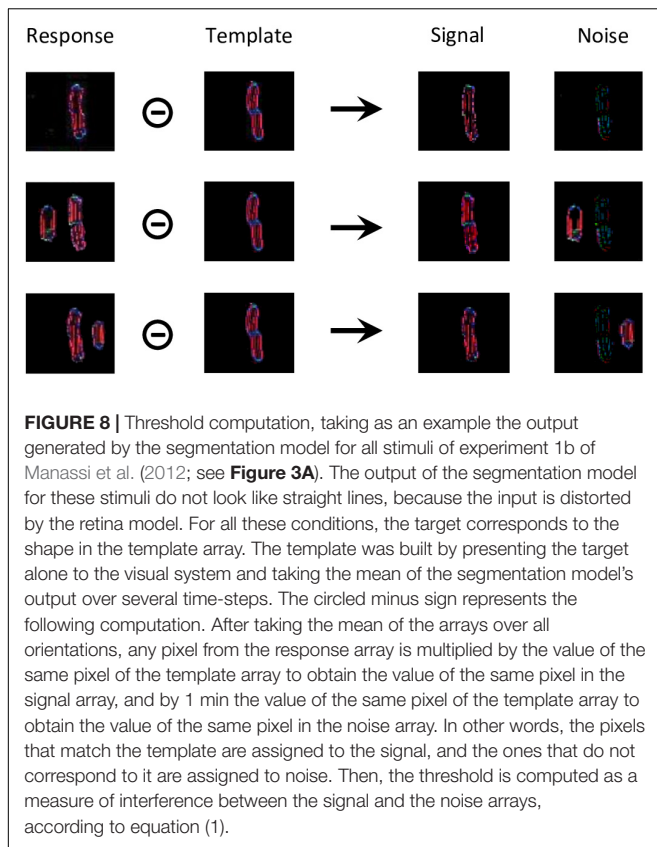
network models the empirical distribution of overt attention across images, we use it as a proxy of covert attention to select interesting objects from the background.

Virtual Experiment and Model Interactions

The virtual environment reproduces the conditions of two experiments that measure inward-outward anisotropy in visual crowding (see **Figure 3**): experiment 1b of Manassi et al. (2012) and experiment 5 of Farzin et al. (2009). A screen displays the visual stimulus (flankers and target) to the eyes of an iCub robot at a specific distance and a specific eccentricity, depending on the conditions of the simulated experiment. In all simulated conditions, the task of the robot is to give a measure of crowding associated to the stimulus, by trying to segment the flanker from the target over many trials. For each trial, the stimulus appears in the periphery of the right visual field of the robot, while the integrated camera of the right eye of the robot constantly records its visual environment and sends its output to the visual system. To process the visual stimulus, the models of the visual system are connected to each other according to the scheme in **Figure 7A**.

Figure 7B shows the result of an example trial simulated with the NRP and highlights the output of all models of the visual system. When the visual stimulus (the target with either



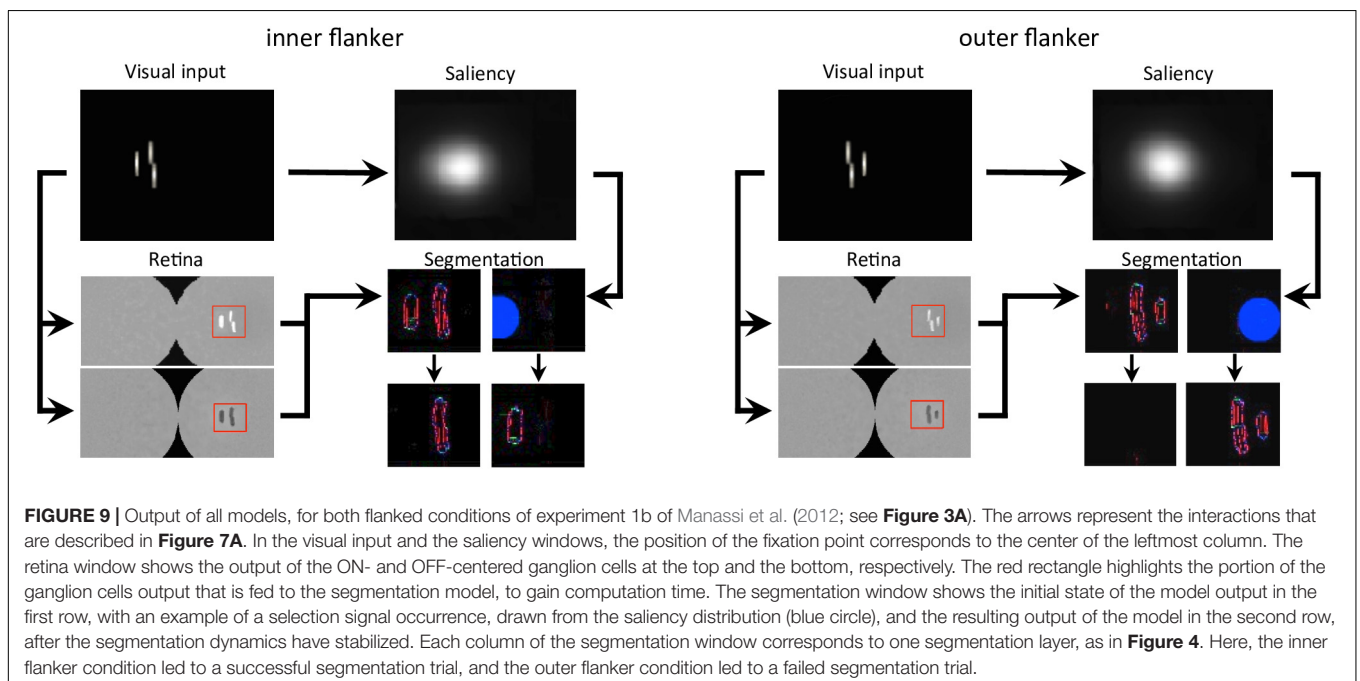


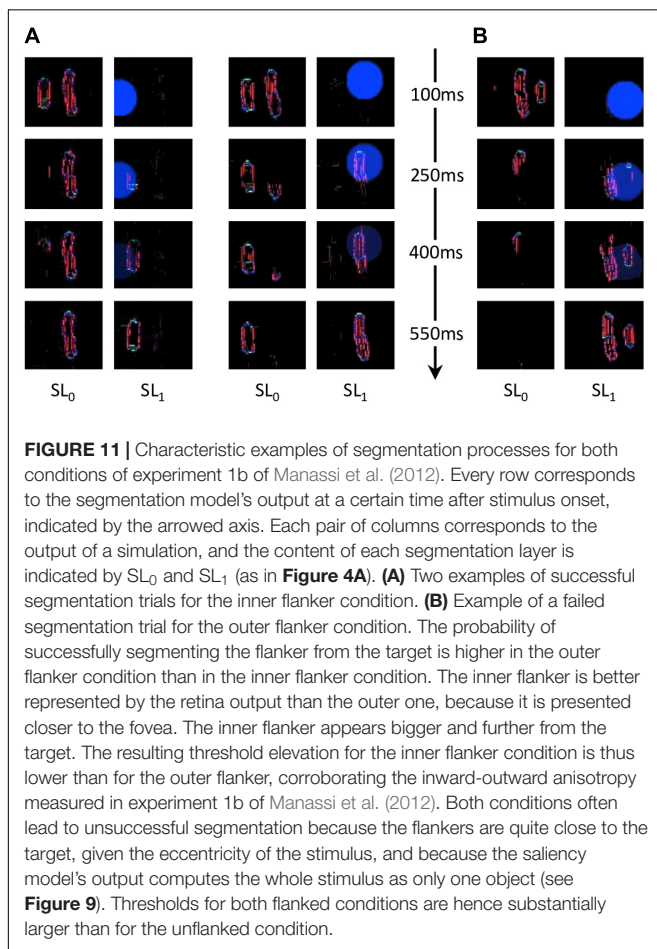
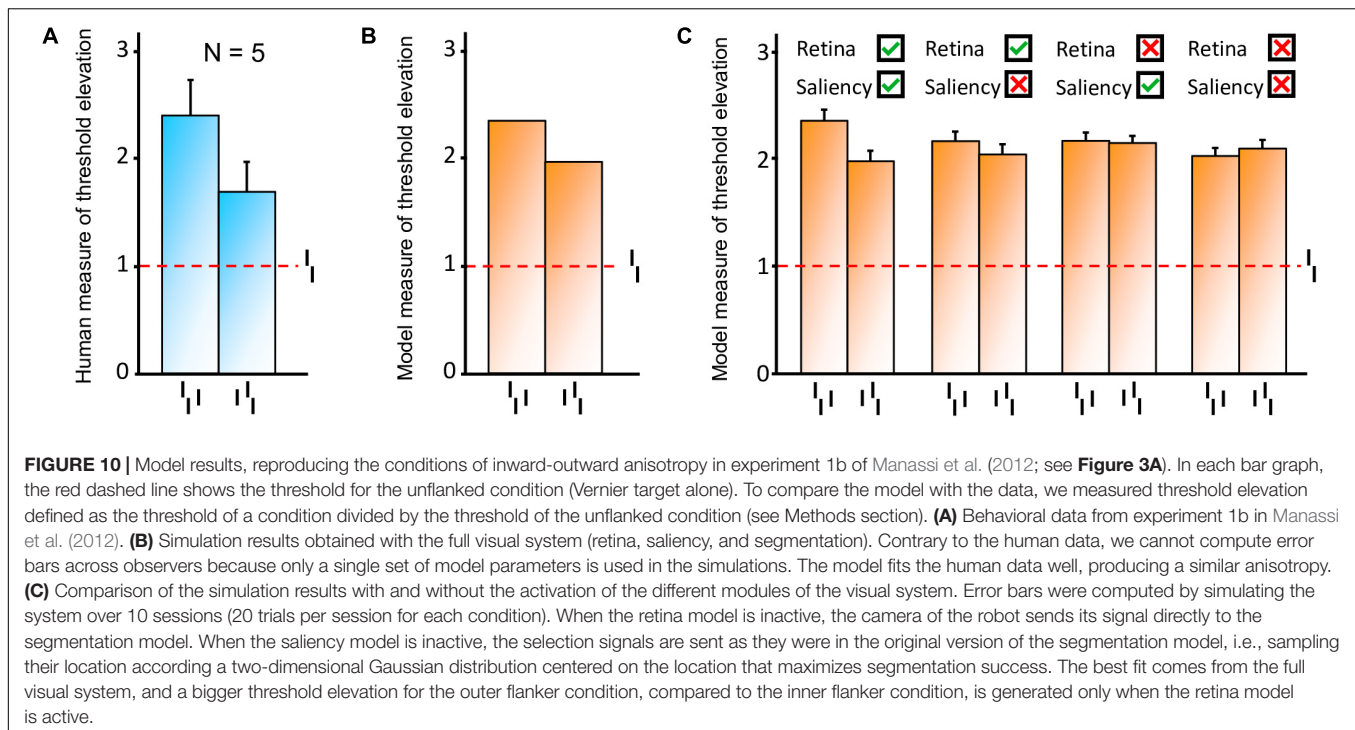
an inner flanker, an outer flanker, or unflanked) appears on the screen, the camera of the robot sends its output to the retina model whose output is delivered to the segmentation model.

Because of the magnification applied by the retina model, the segmentation model represents elements in the visual field with less precision if they appear in the periphery than if they appear near the fovea. At the same time, the saliency model is also fed with the output of the camera. The saliency model is not fed with the output of the retina model because it has been trained on undistorted images. In the simulation, the output of the saliency model corresponds to a probability density distribution of the selection signals that are sent to the segmentation model (see blue circle in **Figure 4**). After stimulus onset, a selection signal, whose location is sampled from the saliency map intensity, starts the segmentation dynamics of the segmentation model. The selection signal is sent to locations near the visual stimulus, because it is very salient. After some processing time, the segmentation stabilizes (groups are formed in the segmentation layers). The location of the selection signal drives the output of the segmentation. If it overlaps with both the target and the flanker, the segmentation is unsuccessful because the flanker and the target interact. If not, the segmentation is successful because the target ends up alone in its segmentation layer. When the target disappears, the activity of the segmentation model is reset by an overall inhibition signal, and the loop starts over.

For each condition of experiment 1b of Manassi et al. (2012) and experiment 5 of Farzin et al. (2009; **Figure 3**), we simulate the visual system of the robot for 20 trials. For each trial, we record a threshold measurement, based on the output of the segmentation model. First, we compare the output array to a target template to separate it into a signal and a noise array (**Figure 8**). The target template is the mean of the segmentation model's output over several time-steps that is generated when the target is presented alone.

Those signal and noise arrays are then used to measure the match M between the output of the segmentation model and the



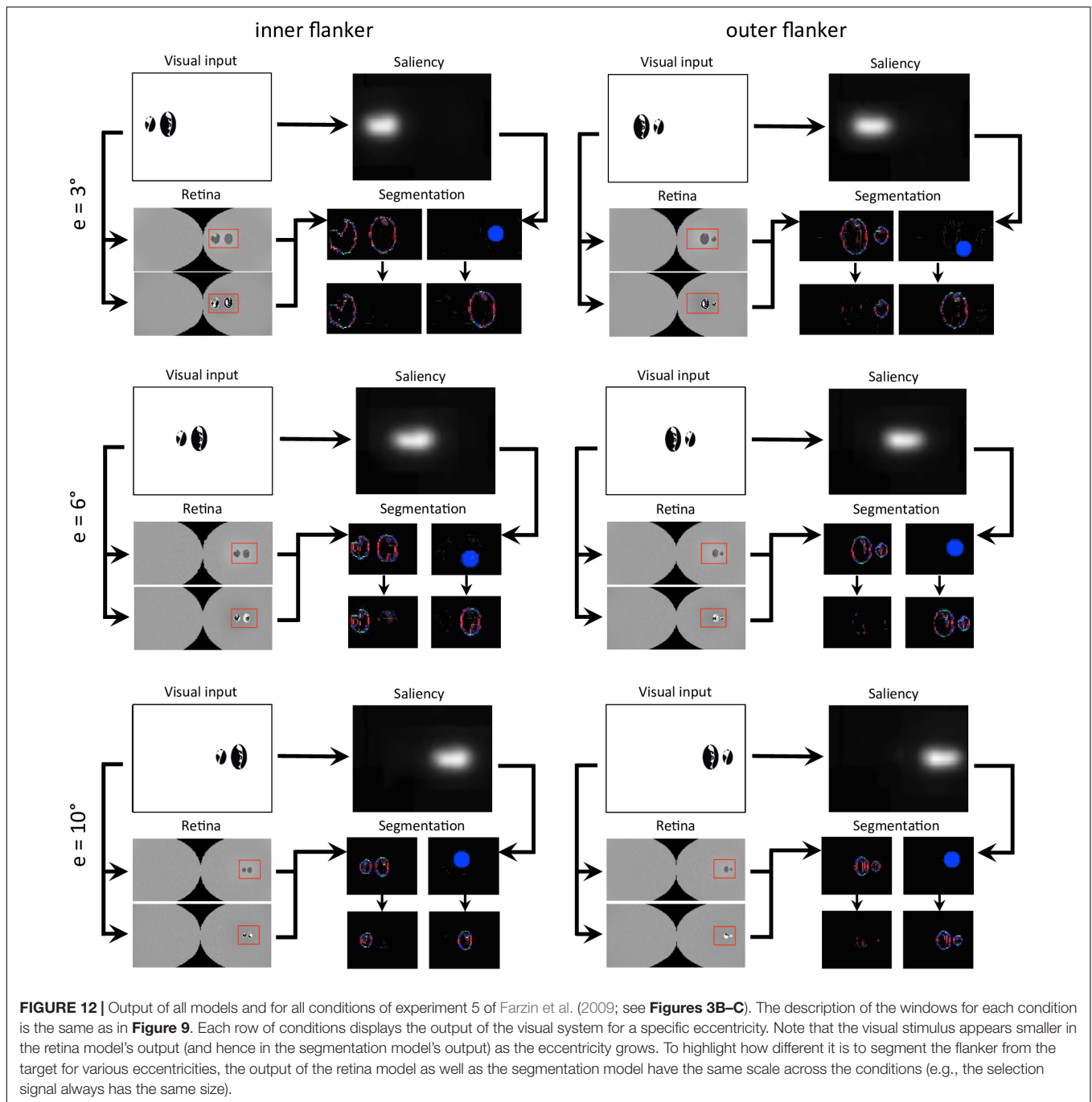


target template, according to equation (1).

$$M = \sum_{i,j} (s_{ij} - \sum_{k,l} n_{kl} \cdot I_0 \cdot e^{-\frac{\sqrt{(i-k)^2 + (j-l)^2}}{\sigma}}) \quad (1)$$

The intensity of pixel (i, j) of the signal array is denoted by s_{ij} and the intensity of pixel (k, l) of the noise array by n_{kl} . The weight of interference between those two pixels decreases exponentially with the distance between them. I_0 is the strength of interaction and sigma is the rate of exponential decrease. I_0 is set to 10^{-3} , a value that was determined to generate sufficient interaction between the target and the flanker, without killing the signal completely. Sigma is set to 30 pixels, a value that was determined to follow approximately the pooling range defined by Bouma's window (Bouma, 1970). Given this fixed value, the pooling range increases with eccentricity in the image space. The more flanker elements, in addition to the target, that are in the segmentation layer, the smaller the match. Note that even for a fully successful segmentation trial, when the target ends up completely alone in one of the segmentation layers, the match is not perfect, because the representation of the target has intrinsic noise and dynamics and thus does not perfectly match the template (**Figure 8**, first row). Also note that a small target generates less signal, and thus a weaker match, than a larger version of the same target. Difficulty of judging Vernier direction is usually measured by identifying the threshold separation needed for an observer to be 75% correct. In the model, we suppose that the threshold is a negative linear function of the match value (the higher the match, the lower the threshold), exactly as in Francis et al. (2017).

Finally, for each condition, we take the mean of the thresholds (T_i) across the trials and divide this value by the mean thresholds



of the unflanked condition, where only the target is presented to the robot. We define this final number as the model measurement of the threshold elevation of the flanking configuration [see equation (2)].

$$E_i = \frac{\frac{1}{N} \sum_{n=1}^N T_i(n)}{\frac{1}{N} \sum_{n=1}^N T_u(n)} \quad (2)$$

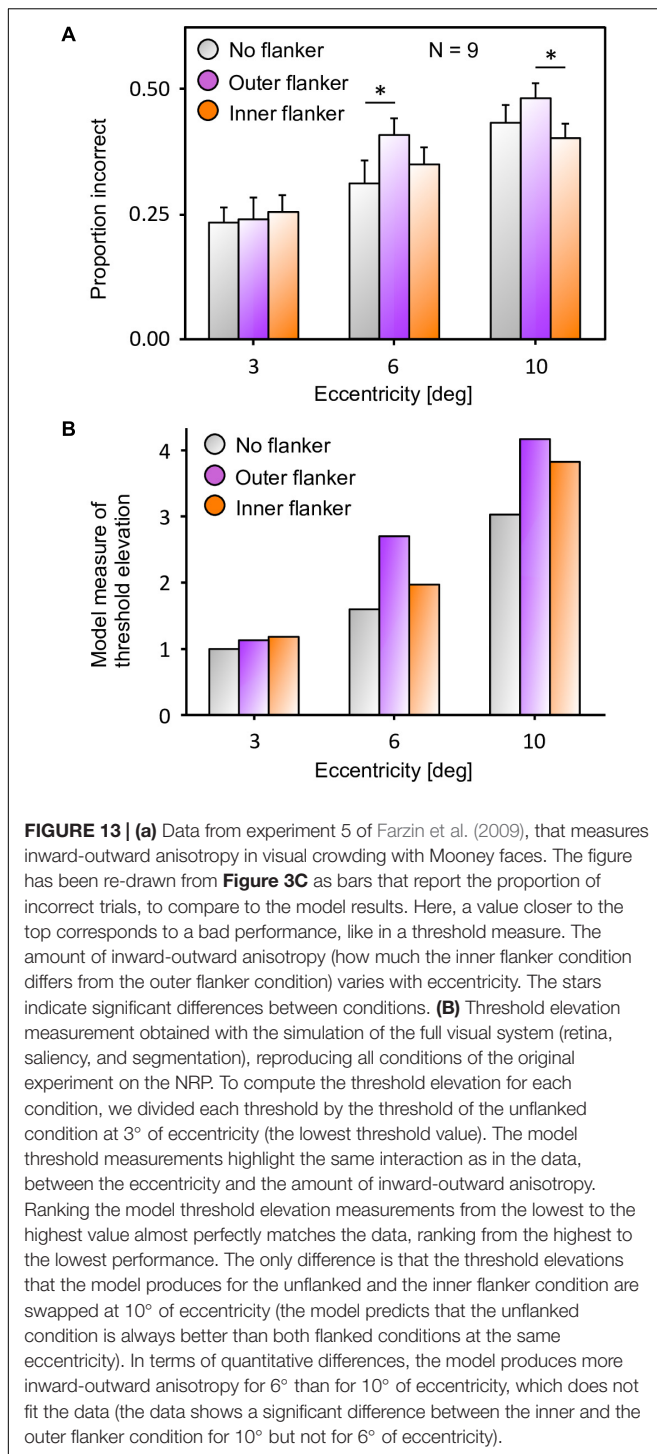
Where E_i is the threshold elevation of condition i , N is the number of trials, $T_i(n)$ is the threshold measurement associated to the segmented output of trial n for condition i , and $T_u(n)$ is

the threshold measurement associated to the segmented output of trial n for the unflanked condition.

RESULTS

Vernier Discrimination Task

First, we reproduced the crowding paradigm of experiment 1b of Manassi et al. (2012; see **Figure 3A**). This experiment measured inward-outward anisotropy in a Vernier discrimination task. In



the simulation, we showed a Vernier target at a fixed eccentricity of 3.89° from the fovea in the right visual field of the robot. The target was either flanked by a short bar on the left side, on the right side, or not flanked at all. Representative outputs of the retina model, the saliency model, and the segmentation model for both flanked conditions are presented in **Figure 9**. The threshold measurements for all conditions, coming from

the NRP simulation as well as from the behavioral data, are shown in **Figures 10A–B**. To investigate the role of each model in the general output of the system, we de-activated the different modules of the visual system and measured the corresponding model output thresholds (**Figure 10C**). Crucially, the simulation of the full visual system (retina, saliency and segmentation models) produces the best fit of the data (i.e., a larger threshold when the target was flanked by an outer bar than when flanked by an inner bar). De-activating only the saliency model in the visual system also generated the same kind of asymmetry as in the data, but to a smaller extent, suggesting that the retina is the main source of asymmetry in this paradigm. Indeed, an inner flanker is better represented by the retina model than an outer flanker, because it appears at a smaller eccentricity. When the flanker is presented on the foveal side, its representation is bigger and appears further from the target, and the segmentation model is more prone to segregate it from the target. This small but crucial difference between both flankers is illustrated in **Figure 11**.

Mooney Face Discrimination Task

Next, we reproduced the crowding paradigm of experiment 5 of Farzin et al. (2009; see **Figures 3B–C**). This experiment measured inward-outward anisotropy using Mooney faces. In this paradigm, the target Mooney face is shown either in the left or the right visual hemi-field, together with either an inner flanker, an outer flanker, or with no flanker and at different eccentricities of 3°, 6°, and 10° (one block per eccentricity and per flanker configuration). Observers were asked to discriminate an upright from an inverted target Mooney face (2-AFC discrimination task). We performed the same model measurements as in the previous simulations. We ran the visual system and collected threshold elevation results for all different eccentricities of the original experiment; presenting the Mooney face target together with either an inner or an outer flanker. The outputs of the retina model, of the saliency model, and of the segmentation model in response to all conditions are presented in **Figure 12**. The threshold measurements, coming from the NRP simulation as well as from the behavioral data, are shown in **Figure 13**. The simulation generates the same interaction between the eccentricity and the amount of inward-outward anisotropy that is found in the empirical data. A substantial difference of threshold elevation between the inner flanker and the outer flanker conditions is measured only for big eccentricities (6° and 10°). The reason is that for a small eccentricity (3°), the representation of the target generated by the retina model is so big that the segmentation is successful in almost every trial. For an inner flanker, the region to select only one of the objects is very large, and the selection signals thus have a very low probability of hitting both the target and the flanker at the same time. For an outer flanker, even if the flanker region gets substantially smaller, the target region is still very big, and most of the selection signals fall on the target, also leading to a very high segmentation success rate. In other words, the task is too easy to highlight any difference between the inner and the outer flanker conditions. For larger eccentricities, the size of the retina output associated with the stimulus becomes smaller, which makes the task more difficult.

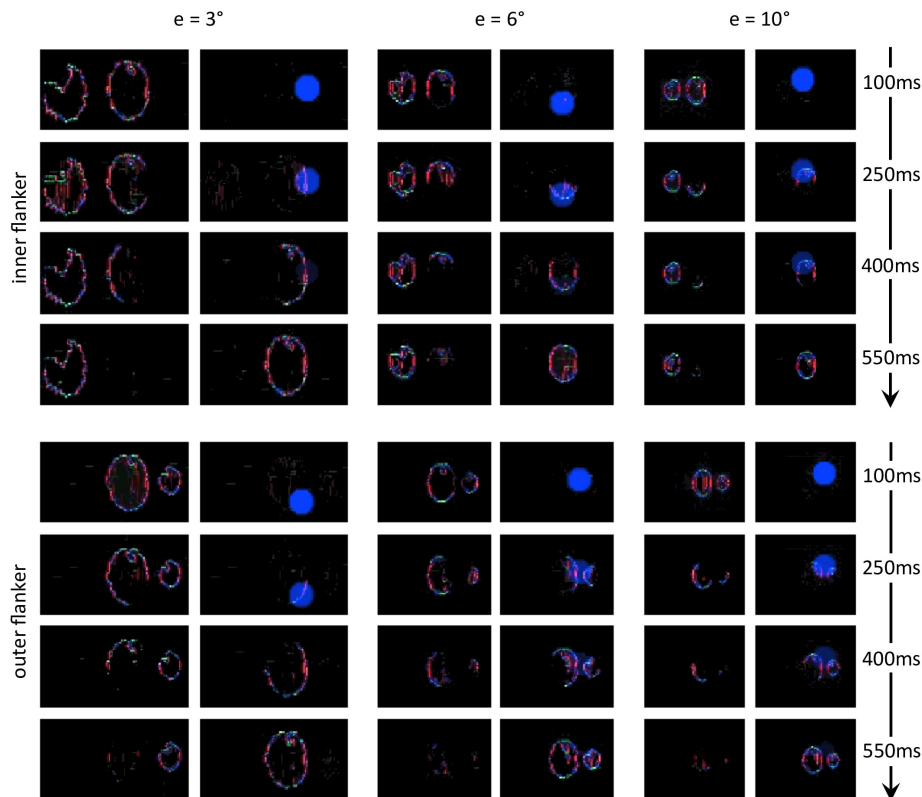


FIGURE 14 | Characteristic examples of segmentation processes for all conditions of experiment 5 of Farzin et al. (2009; see **Figures 3B–C**). The output of the segmentation model has the same scale across the conditions (e.g., the selection signals always have the same size). For both conditions at 3° of eccentricity, the retina model represents the stimulus with such details, that the task is easy even for an outer flanker, resulting in a high rate of segmentation success across the trials, and no substantial difference of threshold between those conditions. For larger eccentricities, the whole stimulus representation has a lower resolution. The target appears smaller, and the flankers appear closer to the target. Many trials thus lead to unsuccessful segmentation, because the selection signals have only a very small region to select only one of the objects. At the same time, the inner flanker is always better represented than the outer flanker. The model thus generates differences of thresholds between the inner-flanker and the outer-flanker conditions.

Over the trials, many selection signals can be unsuccessful (fall on both the target and the flanker) for both inner and outer flanker conditions, highlighting substantial differences in their threshold measurements. Those critical differences between the conditions are illustrated in **Figure 14**.

DISCUSSION

Using the NRP, we simulated a complex visual system composed of several models coming from different research labs. The platform provides satisfactory answers to many of the challenges described in the Introduction. Here, we summarize these issues and briefly explain how the NRP addresses them.

Frameworks

Even if the models that we use have different computational frameworks, the platform allows us to easily integrate them into a common visual system, define their interactions, and simulate them with a minimal amount of code. For example, the segmentation and the saliency models use NEST and TensorFlow, respectively, which the platform supports.

Emulation

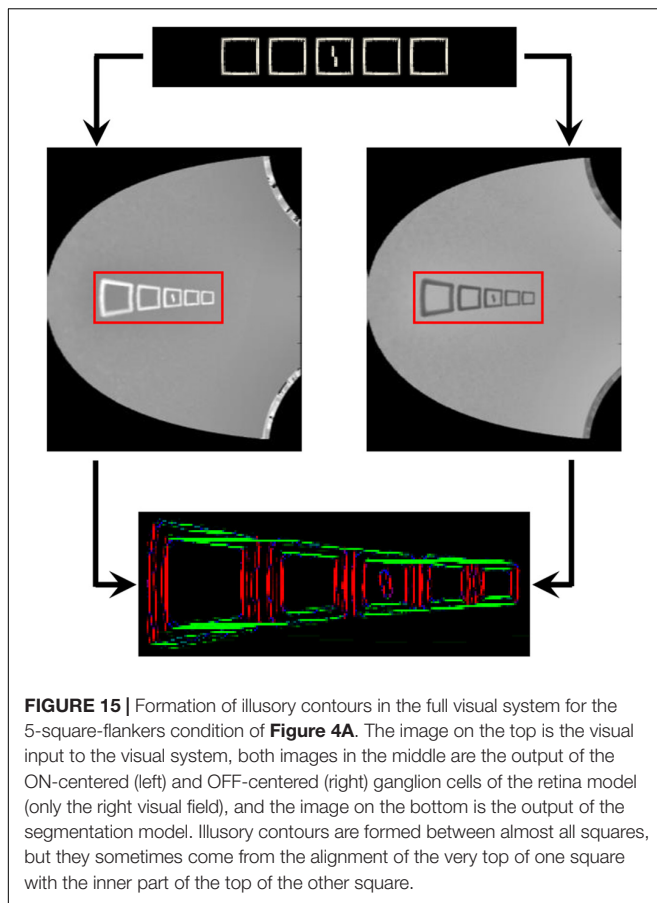
The collaborative aspect of the platform made it possible to quickly integrate the retina model to the simulation. The retina-modeling framework was already incorporated to the platform by other users (Ambrosano et al., 2016), together with some documentation and examples.

Analysis of the System

The NRP allows researchers to de-activate models, simply by commenting out a single line in the setup file of the virtual experiment. This is a powerful tool to investigate how each model contributes to the general output of the system (see **Figure 11C**), or to test competing hypotheses (e.g., compare how two competing models for the same function of vision fit some data).

Synchronization

The platform takes care of the synchronization between the simulated models. In our visual system, the segmentation model is a recurrent network and the saliency model is a feed-forward input-output transform and the NRP ensures that their respective inputs are always consistent. The models are first run in parallel



for a short amount of time. Then the platform collects data from the simulation and computes the relevant inputs for the next simulation step.

Scalability

However, some challenges were handled with less success. Simulating the whole visual system with the required input resolution required very long computational times (2 weeks to simulate all conditions). The platform is currently used online with servers that have rather limited resources. The platform is in development and will soon support high-performance computing.

Reproducibility

Because of the computational limitations, we could not reach the resolution that was required to identify the high-level features of some stimuli (e.g., “face-ness” of the Mooney faces). It would be interesting to check if the “face-ness” of the Mooney faces drastically changes the output of the saliency model and if the model threshold results substantially change.

Ultimately, simulating the visual system on the NRP allowed us to enhance understanding about visual crowding. We could show that the segmentation model that explains crowding and uncrowding (Manassi et al., 2012, 2013; Francis et al., 2017) is able to explain inward-outward anisotropy as well, if it

is connected to a retina model. Traditional explanations of crowding (e.g., pooling models) combined with retinal and cortical magnification would predict that an outer flanker produces less crowding than an inner flanker. The representation of an outer flanker in the visual cortex would appear smaller than the one of an inner flanker, thus causing less interaction with the target through pooling, whose range is expressed in cortical distance. Here, on the contrary, simulating the segmentation model of Francis et al. (2017) in a complex visual system, the prediction is exactly the opposite, thereby matching the data. Indeed, it becomes harder for the visual system to segment the flanker from the target, if the representation of the flanker is small. In other words, the visual system is more likely to treat the flanker and the target as a single object (or group). The grouping hypothesis of Francis et al. (2017) can thus explain uncrowding as well as inward-outward anisotropy. This gives more evidence to the idea that grouping is a central function of human vision (Manassi et al., 2012; Chaney et al., 2014; Harrison and Bex, 2016; Doerig et al., 2019).

The full model simulated with the NRP makes the prediction that inward-outward anisotropy can be observed only for a fixed range of eccentricities. If the eccentricity is too small (e.g., 3° for the paradigm of Farzin et al. (2009); see **Figures 13, 14**), no difference can be observed between the inner flanker and outer flanker conditions because the segmentation is almost always successful in both cases. Indeed, the retinal output related to the visual stimulus is substantially larger than the selection signals, and the probability that the signal covers both the target and the flanker is very low. If the eccentricity is too large (i.e., even bigger eccentricities than in **Figures 13, 14**, e.g., 13° , 16° , or 20°), an inner or an outer flanker becomes indistinguishable from the target, because the stimulus is represented as a tiny spot by the retina. The selection signal of the segmentation model would always cover the whole stimulus, segmenting the target and the flanker as a single group, thereby making no difference between an inner and an outer flanker. In **Figure 13**, the model produces a stronger inward-outward anisotropy for 6° than for 10° of eccentricity, which does not fit the human data. We attribute this discrepancy to a sub-optimal choice of the size of the selection signals in the segmentation model (the radius of the blue circles, e.g., in **Figure 14**). As said above, the radius of the selection signals directly affects the range of eccentricity at which inward-outward anisotropy is observed. If the signals were smaller, the eccentricity at which inward-outward anisotropy is maximal would be larger and vice versa. In general, this tells us that a more sophisticated mechanism should be used to trigger segmentation events. For example, at stimulus onset, the saliency output could instantiate a soft neural competition to determine the location and the size of the selection signal. A threshold, put on the time derivative of all pixel intensities of the saliency output, could even be used to determine when and where to trigger such a competition.

Furthermore, it would be interesting to test how inward-outward anisotropy interacts with uncrowding. A new interesting paradigm would be to continue the experiment 1b of Manassi et al. (2012) with different numbers of short flanking bars. Previously, it has been shown that crowding weakens when

adding more bars on both sides of the target, if they are aligned with each other (experiment 1a of Manassi et al. (2012)). To simulate such paradigms, we need to investigate whether our model of the visual system allows the creation of illusory contours between aligned flankers, such as between the squares of **Figure 4A**, to produce uncrowding. We expect that the distortion due to the retina model impairs the formation of illusory contours between aligned edges, because the segmentation model assumes that spatial pixels correspond to retinal pixels (see Francis et al. (2017) for the exact mechanism). We reproduced the 5-square-flankers condition of **Figure 4A** in the NRP and we simulated the model visual system (**Figure 15**). The segmentation model still generates illusory contours but to a lesser extent. We suspect that the mechanisms need not be changed but the way an aligned neighbor is encoded in the model should be redefined. This simulation highlights how challenging it is to merge different models. The NRP forces us to recognize a challenge in integrating the retina and the segmentation model. Future work is thus needed in order to simulate this kind of paradigm properly.

CONCLUSION

Breaking down the complexity of vision into simple mechanisms fails when the simple mechanisms are put in broader contexts. To fully understand human vision, one needs to build complex systems that process large parts of the visual field and combine many aspects of vision that all require sophisticated modeling. Using the NRP, we could start to simulate such a system by connecting a segmentation model, a saliency model, and a retina model, thereby providing explanations for complex results in visual crowding, such as inward-outward anisotropy. Crucially,

the explanation is in line with the grouping hypothesis of Francis et al. (2017) and predicts how much inward-outward anisotropy would be measured at bigger eccentricities. This early use of the NRP suggests that it provides a solution to some of the challenges that come with simulating big connected systems. We believe the system will prove useful beyond the specific models utilized here; and that it will provide a common platform for general purpose modeling of perception, cognition, and neuroscience.

DATA AVAILABILITY

No datasets were generated or analyzed for this study.

AUTHOR CONTRIBUTIONS

AB, JK, AK, and AA substantially contributed to conducting the underlying research. AB, AK, and AA provided the models descriptions to the manuscript writing process. KC provided the description of the Neurorobotics Platform to the manuscript writing process. AB wrote most of the manuscript and put all parts together. GF, MH, EF, JK, and AK gave substantial feedbacks to the writing process.

FUNDING

This project/research has received funding from the European Union's Horizon 2020 Framework Program for Research and Innovation under the Specific Grant Agreement No. 785907 (Human Brain Project SGA2).

REFERENCES

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). "TensorFlow: a system for large-scale machine learning." in *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, Savannah, GA, 265–283. doi: 10.1371/journal.pone.0181173
- Ambrosano, A., Vannucci, L., Albanese, U., Kirtay, M., Falotico, E., Martínez-Cañada, P., et al. (2016). "Retina color-opponency based pursuit implemented through spiking neural networks in the neurorobotics platform," in *Biomimetic and Biohybrid Systems*, eds N. F. Lepora, A. Mura, M. Mangan, P. F. M. J. Verschure, M. Desmulliez, and T. J. Prescott (New York, NY: Springer International Publishing), 16–27. doi: 10.1007/978-3-319-42417-0_2
- Bouma, H. (1970). Interaction effects in parafoveal letter recognition. *Nature* 226, 177–178. doi: 10.1038/226177a0
- Bouma, H. (1973). Visual interference in the parafoveal recognition of initial and final letters of words. *Vis. Res.* 13, 767–782. doi: 10.1016/0042-6989(73)90041-2
- Cao, Y., and Grossberg, S. (2005). A laminar cortical model of stereopsis and 3D surface perception: closure and da Vinci stereopsis. *Spat. Vis.* 18, 515–578. doi: 10.1163/156856805774406756
- Chaney, W., Fischer, J., and Whitney, D. (2014). The hierarchical sparse selection model of visual crowding. *Front. Integr. Neurosci.* 8:73. doi: 10.3389/fnint.2014.00073
- Clarke, A. M., Herzog, M. H., and Francis, G. (2014). Visual crowding illustrates the inadequacy of local vs. global and feedforward vs. feedback distinctions in modeling visual perception. *Front. Psychol.* 5:1193. doi: 10.3389/fpsyg.2014.01193
- Cowey, A., and Rolls, E. T. (1974). Human cortical magnification factor and its relation to visual acuity. *Exp. Brain Res.* 21, 447–454.
- Daniel, P. M., and Whitteridge, D. (1961). The representation of the visual field on the cerebral cortex in monkeys. *J. Physiol.* 159, 203–221. doi: 10.1113/jphysiol.1961.sp006803
- Doerig, A., Bornet, A., Rosenholtz, R., Francis, G., Clarke, A. M., and Herzog, M. H. (2019). Beyond Bouma's window: how to explain global effects of crowding? *PLoS Comput. Biol.* 15:e1006580. doi: 10.1371/journal.pcbi.1006580
- Eriksen, C. W., and Hoffman, J. E. (1972). Temporal and spatial characteristics of selective encoding from visual displays. *Percept. Psychophys.* 12, 201–204. doi: 10.3758/bf03212870
- Falotico, E., Vannucci, L., Ambrosano, A., Albanese, U., Ulbrich, S., Vasquez Tieck, J. C., et al. (2017). Connecting artificial brains to robots in a comprehensive simulation framework: the neurorobotics platform. *Front. Neurobot.* 11:2. doi: 10.3389/fnbot.2017.00002
- Farzin, F., Rivera, S. M., and Whitney, D. (2009). Holistic crowding of mooney faces. *J. Vis.* 9, 18.1–18.15. doi: 10.1167/9.6.18
- Francis, G., Manassi, M., and Herzog, M. H. (2017). Neural dynamics of grouping and segmentation explain properties of visual crowding. *Psychol. Rev.* 124, 483–504. doi: 10.1037/rev0000070
- Gewaltig, M.-O., and Diesmann, M. (2007). NEST (neural simulation tool). *Scholarpedia* 2:1430. doi: 10.4249/scholarpedia.1430
- Harrison, W. J., and Bex, P. J. (2016). Reply to pachai et al. *Curr. Biol.* 26, R353–R354. doi: 10.1016/j.cub.2016.03.024
- Herzog, M. H., and Manassi, M. (2015). Uncorking the bottleneck of crowding: a fresh look at object recognition. *Curr. Opin. Behav. Sci.* 1, 86–93. doi: 10.1016/j.cobeha.2014.10.006
- Herzog, M. H., Sayim, B., Chichero, V., and Manassi, M. (2015). Crowding, grouping, and object recognition: a matter of appearance. *J. Vis.* 15:5. doi: 10.1167/15.6.5

- Herzog, M. H., Thunell, E., and Ögmen, H. (2016). Putting low-level vision into global context: why vision cannot be reduced to basic circuits. *Vis. Res.* 126, 9–18. doi: 10.1016/j.visres.2015.09.009
- Itti, L., Koch, C., and Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* 20, 1254–1259. doi: 10.1109/34.730558
- Jiang, M., Huang, S., Duan, J., and Zhao, Q. (2015). “Salicon: saliency in context,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, Boston, FL, 1072–1080.
- Koch, C., and Ullman, S. (1985). Shifts in selective visual attention: towards the underlying neural circuitry. *Hum. Neurobiol.* 4, 219–227.
- Koenig, N., and Howard, A. (2004). “Design and use paradigms for gazebo, an open-source multi-robot simulator,” in *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No. 04CH37566)*, (Sendai: IEEE), 2149–2154.
- Kroner, A., Senden, M., Driessens, K., and Goebel, R. (2019). Contextual encoder-decoder network for visual saliency prediction. *arXiv*
- Manassi, M., Hermens, F., Francis, G., and Herzog, M. H. (2015). Release of crowding by pattern completion. *J. Vis.* 15:16. doi: 10.1167/15.8.16
- Manassi, M., Sayim, B., and Herzog, M. H. (2012). Grouping, pooling, and when bigger is better in visual crowding. *J. Vis.* 12:13. doi: 10.1167/12.10.13
- Manassi, M., Sayim, B., and Herzog, M. H. (2013). When crowding of crowding leads to uncrowding. *J. Vis.* 13:10. doi: 10.1167/13.10.10
- Martínez-Cañada, P., Morillas, C., Nieves, J. L., Pino, B., and Pelayo, F. (2015). “First stage of a human visual system simulator: the retina,” in *Computational Color Imaging*, eds A. Trémeau, R. Schettini, and S. Tominaga (New York, NY: Springer International Publishing), 118–127. doi: 10.1007/978-3-319-15979-9_12
- Martínez-Cañada, P., Morillas, C., Pino, B., Ros, E., and Pelayo, F. (2016). A computational framework for realistic retina modeling. *Int. J. Neural Syst.* 26:1650030. doi: 10.1142/S0129065716500301
- Nandy, A. S., and Tjan, B. S. (2012). Saccade-confounded image statistics explain visual crowding. *Nat. Neurosci.* 15, 463–469. doi: 10.1038/nn.3021
- Oberfeld, D., and Stahn, P. (2012). Sequential grouping modulates the effect of non-simultaneous masking on auditory intensity resolution. *PLoS One* 7:e48054. doi: 10.1371/journal.pone.0048054
- Overvliet, K. E., and Sayim, B. (2016). Perceptual grouping determines haptic contextual modulation. *Vis. Res.* 126, 52–58. doi: 10.1016/j.visres.2015.04.016
- Parkes, L., Lund, J., Angelucci, A., Solomon, J. A., and Morgan, M. (2001). Compulsory averaging of crowded orientation signals in human vision. *Nat. Neurosci.* 4, 739–744. doi: 10.1038/89532
- Pelli, D. G. (2008). Crowding: a cortical constraint on object recognition. *Curr. Opin. Neurobiol.* 18, 445–451. doi: 10.1016/j.conb.2008.09.008
- Petrov, Y., and Meleshkevich, O. (2011). Asymmetries and idiosyncratic hot spots in crowding. *Vis. Res.* 51, 1117–1123. doi: 10.1016/j.visres.2011.03.001
- Petrov, Y., Popple, A. V., and McKee, S. P. (2007). Crowding and surround suppression: not to be confused. *J. Vis.* 7:12. doi: 10.1167/7.2.12
- Posner, M. I. (1980). Orienting of attention. *Q. J. Exp. Psychol.* 32, 3–25.
- Saarela, T. P., Westheimer, G., and Herzog, M. H. (2010). The effect of spacing regularity on visual crowding. *J. Vis.* 10:17. doi: 10.1167/10.10.17
- Shapley, R. M., and Victor, J. D. (1978). The effect of contrast on the transfer properties of cat retinal ganglion cells. *J. Physiol.* 285, 275–298. doi: 10.1113/jphysiol.1978.sp012571
- Simonyan, K., and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv*
- Wilson, H. R. (1997). Rapid communication concentric orientation summation in human form vision. *Vis. Res.* 37, 2325–2330. doi: 10.1016/s0042-6989(97)00104-1
- Wohrer, A., and Kornprobst, P. (2009). Virtual retina: a biological retina model and simulator, with contrast gain control. *J. Comput. Neurosci.* 26, 219–249. doi: 10.1007/s10827-008-0108-4
- Wright, R. D., and Ward, L. M. (2008). *Orienting of Attention*. Oxford: Oxford University Press.

Conflict of Interest Statement: KC was employed by the company Fortiss GmbH. Fortiss GmbH is a public research institute financed by the Bavarian region. It is the principal developer of the NRP.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Bornet, Kaiser, Kroner, Falotico, Ambrosano, Cantero, Herzog and Francis. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Complex Electroresponsive Dynamics in Olivocerebellar Neurons Represented With Extended-Generalized Leaky Integrate and Fire Models

Alice Geminiani^{1*}, Claudia Casellato², Egidio D'Angelo^{2,3} and Alessandra Pedrocchi¹

¹ NEARLab, Department of Electronics, Information and Bioengineering, Politecnico di Milano, Milan, Italy, ² Department of Brain and Behavioral Sciences, University of Pavia, Pavia, Italy, ³ IRCCS Mondino Foundation, Pavia, Italy

The neurons of the olivocerebellar circuit exhibit complex electroresponsive dynamics, which are thought to play a fundamental role for network entraining, plasticity induction, signal processing, and noise filtering. In order to reproduce these properties in single-point neuron models, we have optimized the Extended-Generalized Leaky Integrate and Fire (E-GLIF) neuron through a multi-objective gradient-based algorithm targeting the desired input–output relationships. In this way, E-GLIF was tuned toward the unique input–output properties of Golgi cells, granule cells, Purkinje cells, molecular layer interneurons, deep cerebellar nuclei cells, and inferior olivary cells. E-GLIF proved able to simulate the complex cell-specific electroresponsive dynamics of the main olivocerebellar neurons including pacemaking, adaptation, bursting, post-inhibitory rebound excitation, subthreshold oscillations, resonance, and phase reset. The integration of these E-GLIF point-neuron models into olivocerebellar Spiking Neural Networks will allow to evaluate the impact of complex electroresponsive dynamics at the higher scales, up to motor behavior, in closed-loop simulations of sensorimotor tasks.

Keywords: neuronal modeling, point neuron, neuron model simplification, neuronal electroresponsiveness, olivocerebellar neurons

OPEN ACCESS

Edited by:

Mario Senden,
Maastricht University, Netherlands

Reviewed by:

Christian Hansel,
The University of Chicago,
United States

Paolo Bazzigaluppi,
Toronto Western Hospital, Canada

*Correspondence:

Alice Geminiani
alice.geminiani@polimi.it

Received: 15 April 2019

Accepted: 20 May 2019

Published: 06 June 2019

Citation:

Geminiani A, Casellato C,
D'Angelo E and Pedrocchi A (2019)
Complex Electroresponsive Dynamics
in Olivocerebellar Neurons
Represented With
Extended-Generalized Leaky Integrate
and Fire Models.
Front. Comput. Neurosci. 13:35.
doi: 10.3389/fncom.2019.00035

INTRODUCTION

The variety of neuron types and spiking patterns is thought to play a fundamental role for cerebellar signal processing (Llinás, 1988, 2014) and eventually for motor learning and control. By exploiting pacemaking, bursting, adaptation and more complex properties like oscillation and resonance, cerebellar neurons can precisely encode sensorimotor signals, induce plasticity, filter noise, and efficiently communicate with different cerebellar layers and extra-cerebellar circuits (D'Angelo et al., 2016a).

The electroresponsiveness of cerebellar neurons has been deeply characterized *in vitro* and *in vivo*, allowing to identify, for each neuron type, a set of electrophysiological properties, which can be used as a reference for tuning single neuron models (Table 1). All cerebellar cortical neurons except granule cells show autorhythmic activity that becomes irregular *in vivo* due to synaptic inputs. All cerebellar neurons show an almost linear relationship between input current and

firing rate, although with different slopes. In addition, the different cerebellar neurons show specific properties. The Golgi Cells (GoCs) show spike-frequency adaptation (SFA) when depolarized by prolonged currents, post-inhibitory rebound bursts, phase reset, sub-threshold oscillations (STO), and resonance in theta band (Solinas et al., 2007a,b). The granule cells (GRs) exhibit near-threshold oscillations and resonance in theta band (D'Angelo et al., 1998, 2001). The Purkinje Cells (PCs) show a discontinuous f - I_{stim} curve, hysteresis following current ramp stimulation and bistability emerging with high stimulus currents (intrinsic bursting) (McKay and Turner, 2005; Masoli et al., 2015; Buchin et al., 2016). Intrinsic bursting is characterized by a sequence of bursts (depolarized spiking states) and pauses (hyperpolarized quiescent states), which correlate with burst-pause responses observed *in vivo* during behavior (Loewenstein et al., 2005). PC responses consist of simple and complex spikes: simple spikes are high-frequency regular spikes, generated spontaneously or following Parallel Fiber (PF) activation. Complex spikes consist of a burst of action potentials or spikelets, followed by a pause, resulting from Climbing Fiber (CF) excitation (Miall et al., 1998; Rokni et al., 2009). Molecular Layer Interneurons (MLIs) fire spontaneously with an increased firing irregularity *in vivo* (Lachamp et al., 2009; Jörntell et al., 2010) and have no significant SFA (Galliano et al., 2013). These properties derive from the specific set of ionic channels and from their localization on neuronal dendrites, soma and axons, as well as from the specific nature of synaptic inputs.

The deep cerebellar nuclei cells (DCNs) express SFA and post-inhibitory rebound bursting, which is fundamental *in vivo* to modulate the motor output (Hoebeek et al., 2010; Uusisaari and Knöpfel, 2011; Ten Brinke et al., 2017). Based on the expression of marker proteins, two major types of DCN neurons have been identified, with different morphologies, electrophysiological properties, and connectivity patterns (Uusisaari et al., 2007). Large non-GABAergic DCNs (DCNnL) mainly project to pre-motor areas, adapting motor commands during learning tasks, while small GABAergic DCNs (DCNp) are connected to the Inferior Olive, providing feedback on the learning process (Uusisaari and Knöpfel, 2011).

The olivocerebellar circuit functioning strongly relies on the complex dynamics of Inferior Olive (IO) neurons. They exhibit a stereotyped response with slow STO undergoing phase-reset after impulse currents (Long et al., 2002; Kazantsev et al., 2004; Choi et al., 2010; Lefler et al., 2013). Following hyperpolarization, IO neurons generate rebound spikes (De Zeeuw et al., 2003), while when a depolarizing input is applied, single somatic action potentials are translated into bursts of axonal spikes at instantaneous frequency that can exceed 400 Hz (Maruta et al., 2007; Mathy et al., 2009). IO bursts elicit PC complex spikes and promote plasticity in the cerebellar cortex.

In this scenario, single neuron properties have been described in detailed models based on multi-compartment neurons for the different cerebellar layers (Solinas et al., 2007b; Steuber et al., 2011; De Gruilj et al., 2012; D'Angelo et al., 2013; Masoli and D'Angelo, 2017). However, representing this rich

set of electroresponsive patterns through simplified neuron models is fundamental to develop realistic multiscale Spiking Neural Networks (SNNs). To tackle this issue, we here exploited the Extended-Generalized Leaky Integrate and Fire (E-GLIF) point neuron that allows to model single-point neurons while keeping a realistic picture of multiple essential electrophysiological features such as autorhythm, bursting, adaptation, oscillations, and resonance (Geminiani et al., 2018). The E-GLIF, which was originally used to reproduce the GoC electroresponsiveness (Geminiani et al., 2018), was used here to optimize and test the other cerebellar neurons: GRs, PCs, MLIs, DCNs, and IO. The results shown here are fundamental in view of SNNs simulations where the impact of complex single neuron dynamics will be evaluated at the network and, eventually, at the behavioral level (D'Angelo et al., 2016a).

MATERIALS AND METHODS

Single Neuron Model

To reproduce the firing patterns described in the Section "Introduction," single neurons were modeled as E-GLIF point neurons. In previous work, E-GLIF proved able to generate the complete set of GoC spiking responses to different inputs, with a minimum number of equations and free parameters. This makes it the best candidate to be used in SNNs to optimize the compromise between biological plausibility and computational load (Geminiani et al., 2018).

Extended-Generalized Leaky Integrate and Fire couples time-dependent with event-driven algorithmic components and includes three linear Ordinary Differential Equations describing the time evolution of membrane potential (V_m) and of two intrinsic currents (I_{adapt} and I_{dep}). These three state variables are updated at spike events, which are generated according to a probabilistic threshold crossing.

The model is defined as follows:

$$\begin{cases} \frac{dV_m(t)}{dt} = \frac{1}{C_m} \left(\frac{C_m}{\tau_m} (V_m(t) - E_L) - I_{adapt}(t) + I_{dep}(t) + I_e + I_{stim} \right) \\ \frac{dI_{adapt}(t)}{dt} = k_{adapt} (V_m(t) - E_L) - k_2 I_{adapt}(t) \\ \frac{dI_{dep}(t)}{dt} = -k_1 I_{dep}(t) \end{cases}$$

Where:

I_{stim} = external stimulation current;

C_m = membrane capacitance;

τ_m = membrane time constant;

E_L = resting potential;

I_e = endogenous current;

k_{adapt}, k_2 = adaptation constants;

$k_1 = I_{dep}$ decay rate.

If the neuron is in the refractory period t_{ref} , spikes cannot be emitted. Otherwise, a spike is generated stochastically at time t_{spk} , according to an escape rate noise: the nearer V_m is to the threshold potential V_{th} , the higher the probability to have a spike, depending on an exponential function (Gerstner and Kistler, 2002; Jolivet et al., 2006).

TABLE 1 | Electroresponsive properties of cerebellar neurons.

	Auto-rhythm (Hz)	CV _{ISI} (<i>in vitro</i>)	f-I _{stim} slope (Hz/pA)	SFA	Post- inhibitory rebound	Phase reset	Resonance	STO
GoC Forti et al., 2006; Solinas et al., 2007b; D'Angelo et al., 2013	5–15	0.03	~0.2	✓	✓	✓	✓ (ϑ band)	✓ (ϑ band)
GR D'Angelo et al., 2001; Masoli et al., 2017	–	–	~4 ÷ 10	–	–	–	✓ (ϑ band)	✓ (ϑ band)
MLI Lachamp et al., 2009; Galliano et al., 2013	~8.5	0.36	~2.5	–	–	–	–	–
PC McKay and Turner, 2005; Molineux et al., 2006; Lennon et al., 2014; Masoli et al., 2015	40–80	0.04	~0.08	✓	✓	–	–	–
DCNnL Llinás, 1988; Aizenman and Linden, 1999; Uusisaari et al., 2007; De Schutter and Steuber, 2009	~30	0.06	~0.2	✓	✓	–	–	–
DCNp Uusisaari et al., 2007	~10	N.A.	~0.18	✓	✓	–	–	–
IO De Grujil et al., 2012; Lefler et al., 2013	~1	–	–	–	✓	✓	✓	✓ (1–7 Hz)

CV_{ISI}, coefficient of variation of inter-spike intervals; SFA, spike-frequency adaptation; STO, sub-threshold oscillations. Reference literature studies are reported in the first column.

At each spike event, the state variables are updated according to the rules:

$$\begin{cases} V_m(t_{spk}^+) = V_r \\ I_{adap}(t_{spk}^+) = I_{adap}(t_{spk}) + A_2 \\ I_{dep}(t_{spk}^+) = A_1 \end{cases}$$

Where:

t_{spk}^+ = time instant immediately following the spike time t_{spk} ;

V_r = reset potential;

A_2, A_1 = update constants of I_{adap} and I_{dep} , respectively.

Based on k_2 and k_{adap} values, the model exhibits exponential or oscillatory responses (**Figure 1A**). Elements in the model can be associated to different mechanisms that contribute to the spike patterns. The endogenous current, I_e , accounts for autorhythm and regulation of the intrinsic steady-state membrane potential; the adaptive current, I_{adap} , coupled with V_m accounts for intrinsic sub-threshold oscillations of the membrane potential and represents the slow hyperpolarizing sub-cellular currents, e.g., the K^+ channel currents; the spike-triggered current, I_{dep} , accounts for fast depolarizing mechanisms, e.g., the Na^+ and low threshold voltage activated Ca^{2+} channel currents. For neuron connections within SNNs, conductance-based synapses are used, with spike-triggered change of synaptic conductance, g_{syn} , according to an alpha function (Cavallari et al., 2014; Geminiani et al., 2018):

$$g_{syn}(t) = G_{syn} \frac{t - t_{spk}}{\tau_{syn}} e^{1 - \frac{t - t_{spk}}{\tau_{syn}}}$$

where G_{syn} is the maximum conductance change and τ_{syn} the synaptic time constant.

Neuron Model Optimization

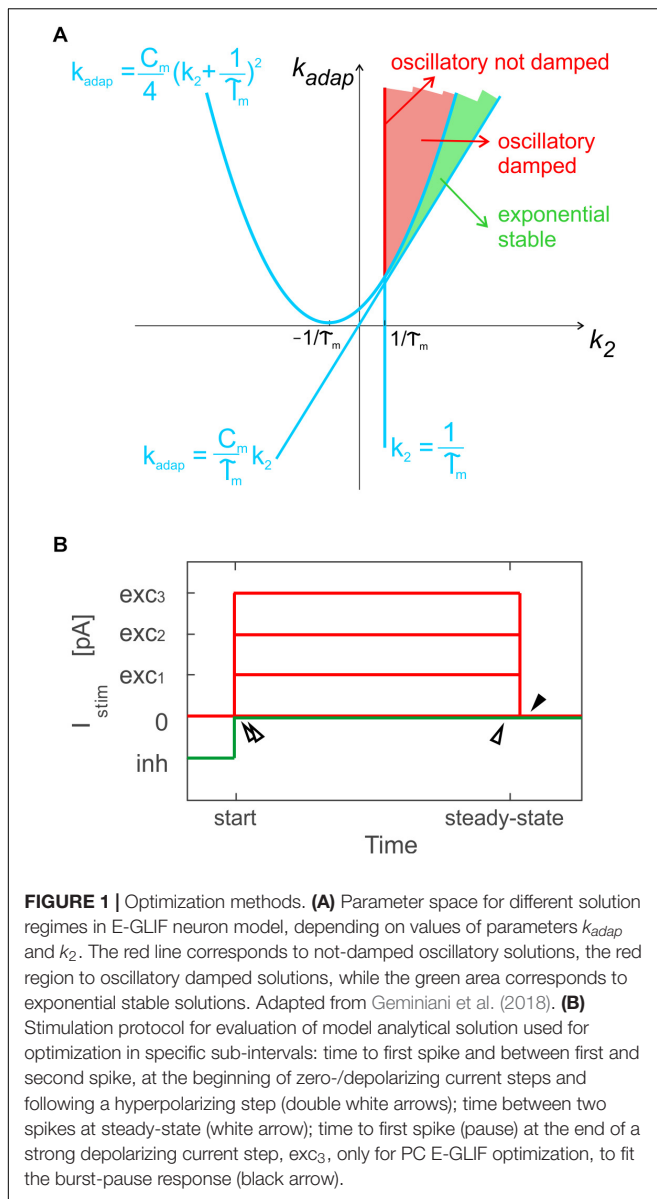
Analogously to the GoC E-GLIF optimization, for each cerebellar neuron we derived the parameters related to neurophysiological quantities (i.e., $C_m, \tau_m, E_L, \Delta t_{ref}, V_{th}, V_r$) from literature *in vitro* experiments (**Table 2**). For the remaining parameters (i.e., $k_{adap}, k_2, k_1, A_2, A_1, I_e$), we used the optimization strategy described in Geminiani et al. (2018), developed in MATLAB, where the cost and constraint functions were adapted to consider the electroresponsive properties of each neuron type as in **Table 1**.

Optimization Stimulation Protocol

Exploiting the analytical solution of the model, the optimization algorithm aimed at minimizing the error on spike times during three sub-intervals of a current step stimulation period, where the V_m solution could be computed: the time to the first spike, the time between first and second spike and the time between two steady-state spikes (**Figure 1B**). A multi-step stimulation protocol was considered for optimization, including: a zero-current phase, three phases with increasing depolarizing currents ($exc_1 < exc_2 < exc_3$), and a zero-current phase following a stimulation interval with a negative current, *inh*.

Cost Function

The cost function evaluated the error on the desired spike times (computed from desired output frequency), in order to fit cell-specific quantitative input–output relationships (**Supplementary Table S1**): (i) autorhythm frequency, when



$I_{stim} = 0$, (ii) response rates ($freq_1 < freq_2 < freq_3$), with increasing amplitudes of I_{stim} ($exc_1 < exc_2 < exc_3$), and (iii) rebound burst latency and initial frequency, following an inhibitory current step, inh . To take into account SFA during depolarizing current steps, the desired steady-state firing rate was obtained from desired frequencies ($freq_1 < freq_2 < freq_3$) multiplied by an attenuating factor ($factor_1, factor_2, factor_3$) based on experimental values.

In addition, only PCs exhibit the burst-pause response (Masoli et al., 2015): to account for this specific property, the PC cost function evaluated also the time to the first spike (i.e., the pause), just after the turning off of $I_{stim} = exc_3$ (Figure 1B).

Optimization Constraints

The cell-specific constraints (Supplementary Table S2) were customized to obtain:

- Neurophysiological ranges of currents in the model;
- Neurophysiological steady-state value of the membrane potential during inhibition (V_{m_inh});
- Oscillatory damped or not (red area in Figure 1A) or exponential (green area in Figure 1A) V_m dynamics (Geminiani et al., 2018), based on k_2 and k_{adap} ranges as in Figure 1A;
- Neurophysiological values of oscillation frequency, in case of oscillatory neurons, i.e., GRs and IOs;
- Sub-threshold value of the steady-state membrane potential ($V_{m_ss_tonic}$) and limited amplitude of oscillations (A_{osc_tonic}) to prevent spontaneous firing in oscillatory neurons without autorhythm – GRs and IOs, in case of zero external input.

The mathematical expression of the cost function, the fitted input-output quantitative patterns and the values of the constraints are reported with proper details in **Supplementary Material**.

Optimization Implementation

For each neuron type, we ran five optimizations with different random initializations of parameters within their ranges, to test the robustness of results with respect to initialization. We chose the optimal parameter set as the median of the final parameters in each optimization run.

Neuron Model Validation

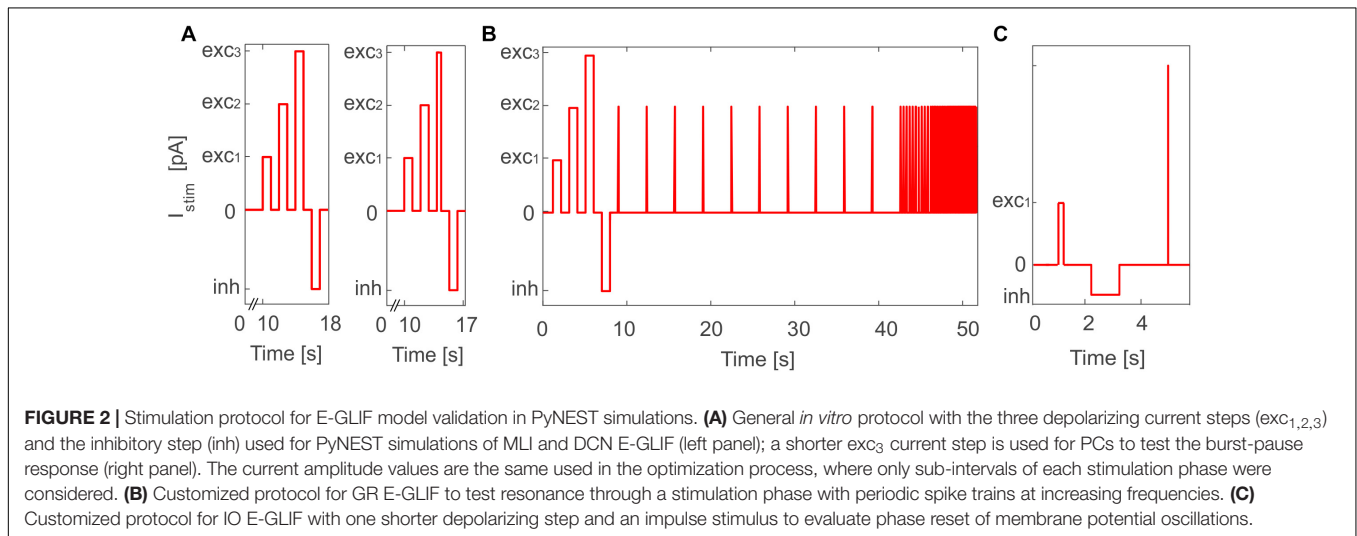
To validate the outcome of optimization and test the effective proper functioning of the model based on literature data, we simulated the E-GLIF responses during a continuous stimulation protocol with current steps in PyNEST (Diesmann and Gewaltig, 2002). This validation was fundamental to assess the result of optimization that was based on the evaluation of the neuron response only in sub-sampled intervals of a continuous simulation. In order to evaluate all the electroresponsive properties in Table 1, the stimulation protocol included a first phase with zero external current, where to measure autorhythm and irregular firing, followed by three depolarizing phases lasting 1 s and interleaved with 1-s zero-current intervals, to measure intrinsic excitability and adaptation. Afterward, a 1-s inhibitory current was applied and turned off in the subsequent step, to test rebound bursting (Figure 2A, left panel). The amplitudes of current steps in each phase were the same used during optimization, but the whole continuous response was here assessed, and not just the sub-intervals included in the optimization. The stimulation protocol was then customized with additional or modified phases for neurons with specific electroresponsive patterns:

- For PCs, we reduced the third depolarizing interval from 1 to 0.01 s (Figure 2A, right panel) to test the burst-pause response with high input currents (McKay and Turner, 2005) and evaluate the effect of current pulses (analogous to CF bursts);
- For GRs, we included an additional phase with input current step trains at increasing frequencies (0.3-3-6-9-12-15 Hz), to evaluate resonance (Figure 2B);

TABLE 2 | Electrophysiological passive properties chosen from literature for the different cerebellar neurons.

	C_m (pF)	τ_m (ms)	E_L (mV)	t_{ref} (ms)	V_r (mV)	V_{th} (mV)
GoC Forti et al., 2006; Solinas et al., 2007a; Tripathy et al., 2014	145 (145 ± 73)	44 (44 ± 22)	−62 (−62)	2 (2 ± 0.4)	−75 (−75)	−55 (−55 ± 1)
GR D'Angelo et al., 1998, 2001; Tripathy et al., 2014; Houston et al., 2017	7 (5.5 ± 0.5)	24.15 (24.15 ± 2)	−62 (−62 ± 11)	1.5 (1.5 ± 0.4)	−70 (−70)	−41 (−41 ± 3)
MLI Lennon et al., 2014	14.6 (14.6)	9.125 (9.125)	−68 (−68)	1.59 (1.59)	−78 (−78)	−53 (−53)
PC Hourez et al., 2011; Hoxha et al., 2012	334 (334 ± 106)	47 (47 ± 32)	−59 (−59 ± 6)	0.5 (0.5 ± 0.1)	−69 (−69)	−43 (−43 ± 2)
DCNnL Uusisaari et al., 2007	142 (142 ± 31)	33 (33 ± 18)	−45 (−45 ± 13)	1.5 (1.5 ± 0.2)	−55 (−55)	−36 (−36 ± 7)
DCNp Uusisaari et al., 2007	56 (56 ± 26)	56 (56 ± 30)	−40 (−40 ± 13)	3.02 (3.02 ± 0.3)	−55 (−55)	−39 (−39 ± 8)
IO Long et al., 2002; De Zeeuw et al., 2003; Van Der Giessen et al., 2008	189 (189 ± 12)	11 (11 ± 4)	−45 (−45)	1 (1)	−45 (−45)	−35 (−35)

Experimental reference values are reported in brackets as mean ± SD (Standard Deviation – when available), from literature reference studies reported in the first column.

**TABLE 3** | Optimized parameter sets of E-GLIF models for each neuron type.

	k_{adapt} (MH^{-1})	k_2 (ms^{-1})	A_2 (pA)	k_1 (ms^{-1})	A_1 (pA)	I_e (pA)
GoC (Geminiani et al., 2018)	0.217	0.023	178.01	0.031	259.988	16.214
GR	0.022	0.041	−0.94	0.311	0.01	−0.888
MLI	2.025	1.096	5.863	1.887	5.953	3.711
PC	1.491	0.041	172.622	0.195	157.622	742.534
DCNnL	0.408	0.047	3.477	0.697	13.857	75.385
DCNp	0.079	0.044	176.358	0.041	176.358	2.384
IO	1.928	0.091	1358.197	0.191	1810.923	−18.101

- For IOs, we considered only one depolarizing phase lasting 0.05 s, to adapt to literature reference protocols for *in vitro* experiments. Then, we tested the effect of different current amplitudes on burst response

properties and we evaluated phase reset of STO, following a current impulse (amplitude = 1 nA, duration = 5 ms), during a zero-current interval lasting 1.5 s (**Figure 2C**).

We ran 10 simulations for each neuron and computed the mean \pm Standard Deviation (SD) of activity parameters (see section “Validation Data Analysis”).

Validation Data Analysis

Significant parameters were extracted from spiking time instants to evaluate single neuron firing patterns in validation protocols:

- The tonic firing rate, f_{tonic} , as the inverse of the mean inter-spike interval (ISI), and the coefficient of variation of inter-spike intervals (CV_{ISI}) to quantify the irregularity of firing, during the zero-current phase;
- The firing rate, f , as the inverse of the mean ISI, during the first three spikes of each depolarizing phase;
- The steady-state firing rate, f_{ss} , as the inverse of the mean ISI, during the last six spikes of each 1-s depolarizing phase;
- The f - I_{stim} slope derived from initial responses to the excitatory step currents;
- The SFA gain, computed as the ratio between f and f_{ss} ;
- Latency and initial frequency (i.e., inverse of the first burst ISI), measured in the rebound burst after hyperpolarization (*lat_rebound* and *rebound_freq*, respectively). Post-inhibitory activity was considered a rebound burst if *lat_rebound* and *rebound_freq* were lower than the autorhythm ISI and higher than the autorhythm frequency, respectively.

To quantify resonance in GRs, we also computed the response speed as the inverse of the mean spike latency in each resonance step; the values from multiple simulation tests and frequencies were fitted through a smoothing spline in order to obtain the resonance curve (Gandolfi et al., 2013).

RESULTS

The single-point models of cerebellar neurons were generated using E-GLIF protocol (Geminiani et al., 2018) and were tuned toward their specific neurophysiological response patterns. For GoCs, we used the same optimal parameters reported in Geminiani et al. (2018). For the other neurons, after fixing the passive properties from literature data (Table 2), the optimization algorithm was used to tune the remaining model parameters toward specific electrophysiological features. In most cases, the algorithm converged to the same region of the parameter space over the five optimization runs (Supplementary Figures S1, S2). The resulting parameter sets achieved the optimal compromise between minimum cost function and constraint violation (below 1.0 and 0.1, respectively), best reproducing the electroresponsiveness of each neuron type (Table 3).

Tuned E-GLIF neurons were then tested in PyNEST simulations with the stimulation protocol described in the Section “Neuron Model Validation.” The model was able to capture the intrinsic excitability of all neurons, generating linearly increasing firing rates with depolarizing current steps. As shown in Figure 3, frequencies values and f - I_{stim} slope were close to the target values for all neurons or within

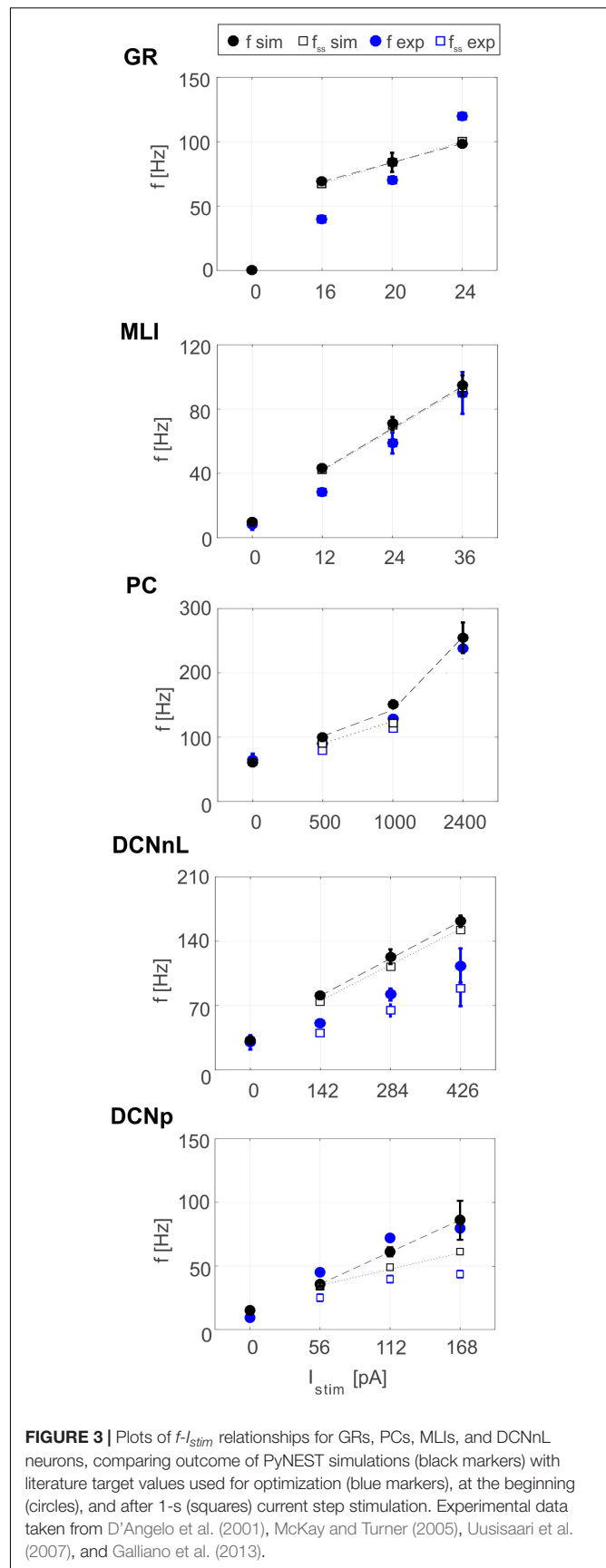


TABLE 4 | Intrinsic excitability properties of optimized E-GLIF neurons.

	f_{tonic} (Hz)	CV_{ISI} (mean)	$f-I_{\text{stim}}$ slope	$lat_rebound$ (ms)	$rebound_freq$ (Hz)
GR	–	–	3.70	–	–
MLI	9.51 ± 0.17	0.13	2.16	(172.96 ± 11.07)	(10.03 ± 1.54)
PC	60.96 ± 0.15	0.04	0.08	10.62 ± 0.15	183.01 ± 6.14
DCNnL	31.48 ± 0.16	0.06	0.28	23.95 ± 0.39	64.81 ± 4.49
DCNp	14.37 ± 0.1	0.09	0.4	69.32 ± 0.94	42.14 ± 3.54
IO	–	–	–	59.22 ± 1.96	–

Values are reported as mean \pm SD over the 10 PyNEST simulations for each neuron type.

acceptable ranges. For GRs the $f-I_{\text{stim}}$ slope was lower than in the reference study (D'Angelo et al., 1998) but still consistent with experimental ranges (Spanne et al., 2014; Masoli et al., 2017). In DCNnL, depolarization frequencies were higher than target values, but linearly increasing with an acceptable $f-I_{\text{stim}}$ slope (Table 4). SFA was present for PCs and DCNnL with average SFA gain of 1.1 at all I_{stim} values, close to the target values of adaptation gain from electrophysiological recordings (1.1 and 1.2, respectively) (Uusisaari et al., 2007; Kim et al., 2013). In DCNp, SFA was more pronounced, with an average gain of 1.3 for $I_{\text{stim}} = exc_{2,3}$ (Uusisaari et al., 2007). In absence of external stimuli, PC, MLI and both DCN E-GLIF produced irregular autorhythm at physiological frequencies, while GRs and IOs generated STO at 6 and 7 Hz, respectively (Figure 4). At the end of a hyperpolarizing current step, PCs and DCNs exhibited rebound excitation (doublets/bursts), which is fundamental for efficient signal transmission (Figure 4). In IOs, post-inhibitory rebound spikes were generated with 50% probability, as in experiments (De Zeeuw et al., 2003; Mathy et al., 2009). When stimulating PC with current pulses of 2.4 nA, the typical intrinsic bursting (burst-pause response) was generated. This was achieved thanks to the balance of model currents, I_{dep} and I_{adap} that accounted for subcellular mechanisms leading to PC complex spikes (De Zeeuw et al., 2011). A 10-ms pulse caused a burst at 254.58 ± 18.26 Hz followed by a pause of 23.47 ± 2.38 ms, longer than the tonic ISI (Figure 5A); with a 50-ms current step the neuron was silent for 32.46 ± 1.22 ms after a burst at 234.87 ± 2.70 Hz (Figure 5B; Grasselli et al., 2016). This spiking pattern well fits with the PC response to dendritic current injection; however, the typical PC bistable regime caused by a continuous high-amplitude stimulation could not be reproduced in the model without losing other electroresponsive properties (Masoli et al., 2015). Intrinsic STO in GRs lead to resonance at 6 Hz, when stimulating the GR neuron model with periodic spike trains at increasing frequencies (Figure 6A). Finally, the optimized E-GLIF model was able to generate also the typical IO bursting response (193.91 ± 24.58 Hz) in case of current step input, thanks to the rapid effect of I_{dep} at the beginning of stimulation and the slower accumulation of I_{adap} that blocked the firing (Figure 5B). Increased amplitudes of the input current caused a non-linear increase of the burst frequency, within physiological ranges; instead, lower currents (i.e., 200 pA) were not sufficient to activate bursts, but they only produced single spikes followed by a pause. Current pulses in the IO E-GLIF induced a spike and a subsequent phase reset of

STO, independent from the phase of the stimulus (Figure 6B). Consistently with experimental results, post-impulse STO phase in the model was $(0.87 \pm 0.02) \cdot T$ for pre-stimulus phases ranging from $0.06 \cdot T$ to $0.92 \cdot T$, being T the period of oscillations (Kazantsev et al., 2004; Lefler et al., 2013).

Therefore, the whole set of olivo-cerebellar cells could be modeled with E-GLIF neurons, generating realistic spiking patterns and capturing crucial electroresponsive properties for cerebellar functioning.

DISCUSSION

In this paper, the E-GLIF model (Geminiani et al., 2018), that was previously developed and validated for Golgi cells, was tuned toward the unique electroresponsive properties of granule cells, Purkinje cells, molecular layer interneurons, deep cerebellar nuclei cells and inferior olivary cells. In these neurons, E-GLIF effectively reproduced pacemaking, adaptation, bursting, post-inhibitory rebound excitation, subthreshold oscillations, resonance, and phase reset. Therefore, for the first time, a whole set of single point neurons is made available to investigate the functional dynamics of the olivocerebellar circuit (Voogd and Glickstein, 1998; Ruigrok, 2011; D'Angelo et al., 2013; Witter et al., 2013; Zhou et al., 2014). These include oscillations and resonance, which are thought to play a critical role for network entraining into large-scale brain oscillations (De Zeeuw et al., 2011; Courtemanche et al., 2013; Llinás, 2014), and long-term synaptic plasticity, which is considered the main mechanism underlying the cerebellar role in motor control and learning (Ito et al., 2014; D'Angelo et al., 2016b).

Modeled Single Neuron Dynamics

Extended-Generalized Leaky Integrate and Fire (Geminiani et al., 2018) is a simplified point-neuron based on a system of three linear ordinary differential equations and its analytical tractability allows to define different solution regimes and to tune model parameters through a generalizable optimization algorithm. In the current work, E-GLIF was able to simulate complex input-output relationships of cerebellar and IO neurons, generating cell-specific intrinsic excitability and non-linear firing properties that would not be possible using previous GLIF models (Mihalaş and Niebur, 2009).

For neurons with oscillatory V_m , the second order dynamics of the model allowed to simulate intrinsic self-sustained

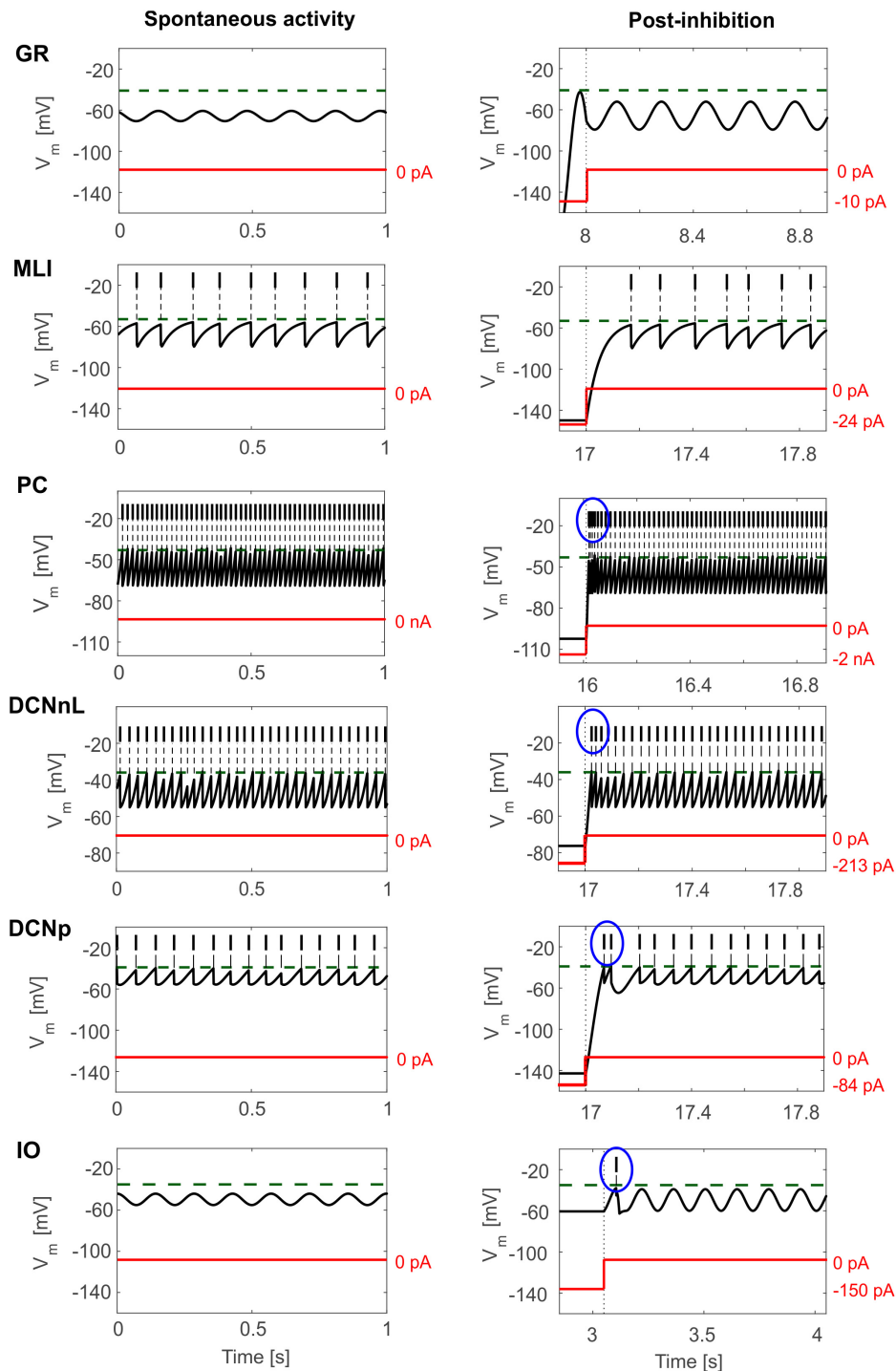


FIGURE 4 | E-GLIF responses to zero-input current (left column) and following a hyperpolarizing current step (right column) for the main olivo-cerebellar neurons. Zero-current inputs cause STO in GR and IO neurons, and autorhythm in the others. Post-inhibitory rebound excitation (burst or spike) is highlighted in the blue circle, where present.

STO. Second order dynamics allowed to reproduce also other non-linear electroresponsive behaviors like resonance in GRs and phase reset of STO in IO neurons. These properties have been measured in single-neuron experiments and are

probably amplified at network level (D'Angelo et al., 2001). Specifically, the feedback inhibitory loop from GoCs to GRs is supposed to contribute to resonance and oscillations in the Granular layer network, enhancing theta-band signals

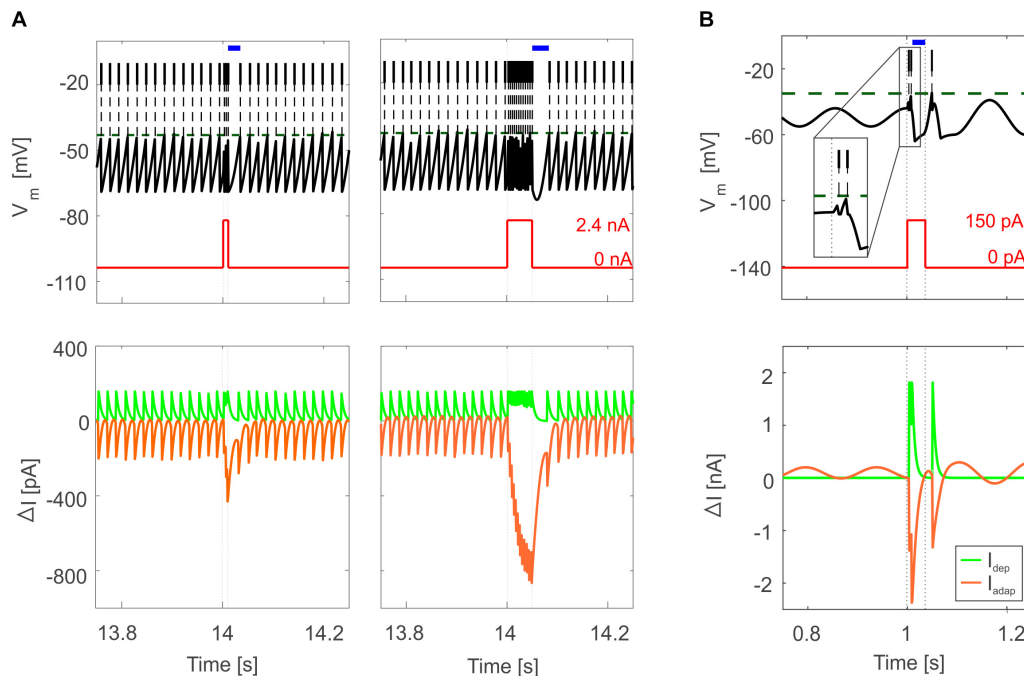


FIGURE 5 | Bursting responses in E-GLIF simulations. **(A)** Burst-pause in PC E-GLIF with a 10-ms input current step (left panel) and 50-ms input current step (right panel). V_m and input current traces are reported in top panels, showing the burst during the stimulation phase and the subsequent pause (blue segment) when the current goes back to 0 nA. Model current traces are reported in bottom panels, with respect to their steady-state value (ΔI). The I_{adapt} current is reported in negative values as it has a hyperpolarizing effect in the neuron model. At the end of the stimulation, the accumulated inhibitory effect of I_{adapt} causes the pause, until it decays, and the tonic balance of currents is restored. **(B)** Bursting response in IO E-GLIF during a 50-ms current step stimulus, showing a first doublet (zoom in the inset) followed by a pause (blue segment); even in this case, the intrinsic model currents drive the V_m response (bottom panel).

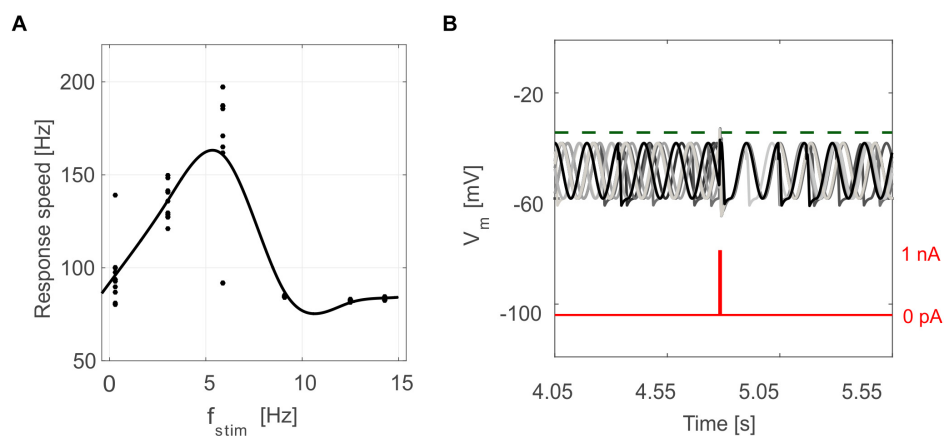


FIGURE 6 | Oscillation-driven properties in E-GLIF simulations. **(A)** Resonance in GR E-GLIF following spike train stimulation at increasing frequencies. The resonance curve was obtained fitting the data points from 10 simulations (black dots) with a smoothing spline. **(B)** Phase-reset of IO E-GLIF V_m during 10 simulations with the same current pulse, causing a spike and a subsequent reset of oscillation phase, independent from the phase before the stimulus.

coming from extra-cerebellar regions (D'Angelo and Casali, 2013; Gandolfi et al., 2013). Future simulations of the granular layer network with E-GLIF neurons will help to elucidate the different contribution of single cell and circuit properties on network oscillations and resonance. This would extend the results of previous studies where detailed microcircuit models and SNNs with Leaky Integrate-and-Fire units were exploited

(D'Angelo et al., 2013; Casali et al., 2019). In the IO circuit, phase reset of STO has been measured in single neurons (Kazantsev et al., 2004), but synchronous stimulation of an olivary area was shown to amplify this response (Lefler et al., 2013). The IO E-GLIF could reproduce the first response during simulation of *in vitro* protocols. In principle, adding gap junctions to the neuron model would account also for the

phase-reset amplification at network level, thanks to the intrinsic communication within IO nuclei.

To simulate IO neurons, E-GLIF was optimized taking the axonal bursting regime as the target behavior (Maruta et al., 2007; Mathy et al., 2009). This aspect challenges the traditional view of CFs as a low-frequency all-or-none signaling pathway: indeed, bursting and rebound activity in IO is fundamental for information encoding, as rebound excitation amplifies the feedback from DCNp cells and olivary bursts elicit complex spikes at PC level. PC E-GLIF successfully reproduced regular firing and the burst-pause pattern following dendritic current stimulation *in vitro*, which can be associated to simple and complex spikes *in vivo* (Masoli et al., 2015). However, bistability and spiking patterns with longer bursts and pauses could not be obtained in the E-GLIF model without losing intrinsic excitability properties. For simulations in SNNs, this is a sufficient approximation since it allows to generate the typical PC network spiking patterns, as shown in the Section “Results.” However, for a more detailed representation even of axonal responses, a multi-compartment version of the PC E-GLIF could be implemented, where multiple E-GLIF neurons are optimized to reproduce the electroresponsiveness of the main PC compartments.

In cerebellar nuclei neurons, rebound excitation has been widely proven *in vitro* but long debated *in vivo* (Alviña et al., 2008). However, recent experimental findings demonstrate that rebound bursting correlates with motor responses and is fundamental for integrating synaptic inputs from PCs, MFs, and IO neurons that all converge in the cerebellar nuclei (Hoebeek et al., 2010; Manto and Oulad Ben Taib, 2010; Witter et al., 2013; Sarnaik and Raman, 2018). Rebound excitation also contributes to cerebellum-driven learning, as demonstrated for associative learning (Ten Brinke et al., 2017). Single-neuron rebound properties are thus crucial in SNNs aimed at multiscale simulations of sensorimotor tasks.

This scenario shows the capability of the E-GLIF point neuron to reproduce the variety of olivo-cerebellar spiking responses following different input stimuli, through a single optimal set of model parameters. Conversely, the traditional approach for single neuron modeling aims at identifying different regions of the parameter space corresponding to different spiking behaviors (Izhikevich, 2003). This makes E-GLIF a best candidate for simulations of SNNs, where neuron response needs to depend on the received input, rather than on the parameter values, achieving higher neurophysiological realism without increasing computational load.

REFERENCES

- Aizenman, C. D., and Linden, D. J. (1999). Regulation of the rebound depolarization and spontaneous firing patterns of deep nuclear neurons in slices of rat cerebellum. *J. Neurophysiol.* 82, 1697–1709. doi: 10.1152/jn.1999.82.4.1697
- Alviña, K., Walter, J. T., Kohn, A., Ellis-Davies, G., and Khodakhah, K. (2008). Questioning the role of rebound firing in the cerebellum. *Nat. Neurosci.* 11, 1256–1258. doi: 10.1038/nn.2195
- Buchin, A., Rieubland, S., Häusser, M., Gutkin, B. S., and Roth, A. (2016). Inverse stochastic resonance in cerebellar purkinje cells. *PLoS Comput. Biol.* 12:e1005000. doi: 10.1371/journal.pcbi.1005000
- Casali, S., Marenzi, E., Medini, C., Casellato, C., and D’Angelo, E. (2019). Reconstruction and simulation of a scaffold model of the cerebellar network. *Front. Neuroinform.* 13:37. doi: 10.3389/fninf.2019.00037
- Cavallari, S., Panzeri, S., and Mazzoni, A. (2014). Comparison of the dynamics of neural interactions between current-based and conductance-based

CONCLUSION

The E-GLIF single-point neuron models were able to capture the complex non-linear dynamics of olivocerebellar neurons including spontaneous firing, subthreshold oscillations, bursting, phase-reset, and resonance. These ingredients, coupled to algorithms accounting for synaptic integration over dendrites (e.g., Marasco et al., 2012; Rössert et al., 2016), will provide the fundamental ingredients to reconstruct non-linear dynamics in extended spiking cerebellar networks. Future work will include embedding these neuron models into cerebellar SNNs to simulate cerebellum-driven motor paradigms and evaluate the impact of single neuron electroresponsiveness on network dynamics, plasticity and, eventually, motor behavior.

DATA AVAILABILITY

All datasets generated for this study are included in the manuscript and/or the **Supplementary Files**.

AUTHOR CONTRIBUTIONS

AG and CC elaborated the mathematical model and optimization, designed and carried out the simulations for each neuron, performed the data analysis, and wrote the manuscript. ED and AP coordinated the whole work and substantially contributed to the writing of the final manuscript.

FUNDING

This project has been developed within the *CerebNest* HBP Partnering Project and has received funding from the European Union’s Horizon 2020 Framework Programme for Research and Innovation under Grant Agreement No. 785907 (Human Brain Project SGA2).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fncom.2019.00035/full#supplementary-material>

- integrate-and-fire recurrent networks. *Front. Neural Circ.* 8:12. doi: 10.3389/fncir.2014.00012
- Choi, S., Yu, E., Kim, D., Urbano, F. J., Makarenko, V., Shin, H. S., et al. (2010). Subthreshold membrane potential oscillations in inferior olive neurons are dynamically regulated by P/Q- and T-type calcium channels: a study in mutant mice. *J. Physiol.* 588, 3031–3043. doi: 10.1113/jphysiol.2009.184705
- Courtemanche, R., Robinson, J. C., and Aponte, D. I. (2013). Linking oscillations in cerebellar circuits. *Front. Neural Circuits* 7:125. doi: 10.3389/fncir.2013.00125
- D'Angelo, E., Antonietti, A., Casali, S., Casellato, C., Garrido, J. A., Luque, N. R., et al. (2016a). Modeling the cerebellar microcircuit: new strategies for a long-standing issue. *Front. Cell. Neurosci.* 10:176. doi: 10.3389/fncel.2016.00176
- D'Angelo, E., Mapelli, L., Casellato, C., Garrido, J. A., Luque, N., Monaco, J., et al. (2016b). Distributed circuit plasticity: new clues for the cerebellar mechanisms of learning. *Cereb.* 15, 139–151. doi: 10.1007/s12311-015-0711-7
- D'Angelo, E., and Casali, S. (2013). Seeking a unified framework for cerebellar function and dysfunction: from circuit operations to cognition. *Front. Neural Circuits* 6:116. doi: 10.3389/fncir.2012.00116
- D'Angelo, E., De Filippi, G., Rossi, P., and Taglietti, V. (1998). Ionic mechanism of electroresponsiveness in cerebellar granule cells implicates the action of a persistent sodium current. *J. Neurophysiol.* 80, 493–503. doi: 10.1152/jn.1998.80.2.493
- D'Angelo, E., Nieuws, T., Maffei, A., Armano, S., Rossi, P., Taglietti, V., et al. (2001). Theta-frequency bursting and resonance in cerebellar granule cells: experimental evidence and modeling of a slow K-dependent mechanism. *J. Neurosci.* 21, 759–770. doi: 10.1523/jneurosci.21-03-00759.2001
- D'Angelo, E., Solinas, S., Mapelli, J., Gandolfi, D., Mapelli, L., and Prestori, F. (2013). The cerebellar Golgi cell and spatiotemporal organization of granular layer activity. *Front. Neural Circuits* 7:93. doi: 10.3389/fncir.2013.00093
- De Gruijl, J. R., Bazzigaluppi, P., de Jeu, M. T. G., and De Zeeuw, C. I. (2012). Climbing fiber burst size and olivary sub-threshold oscillations in a network setting. *PLoS Comput. Biol.* 8:e1002814. doi: 10.1371/journal.pcbi.1002814
- De Schutter, E., and Steuber, V. (2009). Patterns and pauses in Purkinje cell simple spike trains: experiments, modeling and theory. *Neuroscience* 162, 816–826. doi: 10.1016/j.neuroscience.2009.02.040
- De Zeeuw, C. I., Chorev, E., Devor, A., Manor, Y., Van Der Giessen, R. S., De Jeu, M. T., et al. (2003). Deformation of network connectivity in the inferior olive of connexin 36-deficient mice is compensated by morphological and electrophysiological changes at the single neuron level. *J. Neurosci.* 23, 4700–4711. doi: 10.1523/JNEUROSCI.23-11-04700.2003
- De Zeeuw, C. I., Hoebeek, F. E., Bosman, L. W. J., Schonewille, M., Witter, L., and Koekkoek, S. K. (2011). Spatiotemporal firing patterns in the cerebellum. *Nat. Rev. Neurosci.* 12, 327–344. doi: 10.1038/nrn3011
- Diesmann, M., and Gewaltig, M.-O. (2002). "NEST: An environment for neural systems simulations," in *Forsch. und Wissenschaftliches Rechn. Beiträge Zum Heinz-billing-pr.* 2001, Vol. 58, eds T. Plesser and V. Macho (Göttingen: GWDG-Bericht), 43–70.
- Forti, L., Cesana, E., Mapelli, J., and Angelo, E. D. (2006). Ionic mechanisms of autorhythmic firing in rat cerebellar Golgi cells. *J. Physiol.* 3, 711–729. doi: 10.1113/jphysiol.2006.110858
- Galliano, E., Gao, Z., Schonewille, M., Todorov, B., Simons, E., Pop, A. S., et al. (2013). Silencing the majority of cerebellar granule cells uncovers their essential role in motor learning and consolidation. *Cell Rep.* 3, 1239–1251. doi: 10.1016/j.celrep.2013.03.023
- Gandolfi, D., Lombardo, P., Mapelli, J., Solinas, S., and D'Angelo, E. (2013). Theta-frequency resonance at the cerebellum input stage improves spike timing on the millisecond time-scale. *Front. Neural Circuits* 7:64. doi: 10.3389/fncir.2013.00064
- Geminiani, A., Casellato, C., Locatelli, F., Prestori, F., Pedrocchi, A., and D'Angelo, E. (2018). Complex dynamics in simplified neuronal models: reproducing Golgi cell electroresponsiveness. *Front. Neuroinform.* 12:88. doi: 10.3389/fninf.2018.00088
- Gerstner, W., and Kistler, W. M. (2002). *Spiking Neuron Models*. Cambridge: Cambridge Univ. Press.
- Grasselli, G., He, Q., Wan, V., Adelman, J. P., Ohtsuki, G., and Hansel, C. (2016). Activity-dependent plasticity of spike pauses in cerebellar purkinje cells. *Cell Rep.* 14, 2546–2553. doi: 10.1016/j.celrep.2016.02.054
- Hoebeek, F. E., Witter, L., Ruigrok, T. J. H., and De Zeeuw, C. I. (2010). Differential olivo-cerebellar cortical control of rebound activity in the cerebellar nuclei. *Proc. Natl. Acad. Sci. U.S.A.* 107, 8410–8415. doi: 10.1073/PNAS.0907118107
- Hourez, R., Servais, L., Orduz, D., Gall, D., Millard, I., de Kerchove d'Exaerde, A., et al. (2011). Aminopyridines correct early dysfunction and delay neurodegeneration in a mouse model of spinocerebellar ataxia type 1. *J. Neurosci.* 31, 11795–11807. doi: 10.1523/JNEUROSCI.0905-11.2011
- Houston, C. M., Diamanti, E., Diamantaki, M., Kutsarova, E., Cook, A., Sultan, F., et al. (2017). Exploring the significance of morphological diversity for cerebellar granule cell excitability. *Sci. Rep.* 7:46147. doi: 10.1038/srep46147
- Hoxha, E., Boda, E., Montarolo, F., Parolisi, R., and Tempia, F. (2012). Excitability and synaptic alterations in the cerebellum of APP/PS1 mice. *PLoS One* 7:726. doi: 10.1371/journal.pone.0034726
- Ito, M., Yamaguchi, K., Nagao, S., and Yamazaki, T. (2014). Long-term depression as a model of cerebellar plasticity. *Prog. Brain Res.* 210, 1–30. doi: 10.1016/B978-0-444-63356-9.00001-7
- Izhikevich, E. M. (2003). Simple model of spiking neurons. *IEEE Trans. Neural Netw.* 14, 1569–1572. doi: 10.1109/TNN.2003.820440
- Jolivet, R., Rauch, A., Lüscher, H. R., and Gerstner, W. (2006). Predicting spike timing of neocortical pyramidal neurons by simple threshold models. *J. Comput. Neurosci.* 21, 35–49. doi: 10.1007/s10827-006-7074-5
- Jörentell, H., Bengtsson, F., Schonewille, M., and De Zeeuw, C. I. (2010). Cerebellar molecular layer interneurons - computational properties and roles in learning. *Trends Neurosci.* 33, 524–535. doi: 10.1016/j.tins.2010.08.004
- Kazantsev, V. B., Nekorkin, V. I., Makarenko, V. I., and Llinás, R. (2004). Self-referential phase reset based on inferior olive oscillator dynamics. *Proc. Natl. Acad. Sci. U.S.A.* 101, 18183–18188. doi: 10.1073/pnas.0407900101
- Kim, Y. S., Kang, E., Makino, Y., Park, S., Shin, J. H., Song, H., et al. (2013). Characterizing the conductance underlying depolarization-induced slow current in cerebellar Purkinje cells. *J. Neurophysiol.* 109, 1174–1181. doi: 10.1152/jn.01168.2011
- Lachamp, P. M., Liu, Y., and Liu, S. J. (2009). Glutamatergic modulation of cerebellar interneuron activity is mediated by an enhancement of GABA release and requires protein kinase A/RIM1 signaling. *J. Neurosci.* 29, 381–392. doi: 10.1523/JNEUROSCI.2354-08.2009
- Lefler, Y., Torben-nielsen, B., and Yarom, Y. (2013). Oscillatory activity, phase differences, and phase resetting in the inferior olivary nucleus. *Front. Syst. Neurosci.* 7:22. doi: 10.3389/fnsys.2013.00022
- Lennon, W., Hecht-nielsen, R., and Yamazaki, T. (2014). A spiking network model of cerebellar Purkinje cells and molecular layer interneurons exhibiting irregular firing. *Front. Comput. Neurosci.* 8:157. doi: 10.3389/fncom.2014.00157
- Llinás, R. R. (1988). The intrinsic electrophysiological properties of mammalian neurons: insights into central nervous system function. *Science* 242, 1654–1664. doi: 10.1126/science.3059497
- Llinás, R. R. (2014). The olivo-cerebellar system: a key to understanding the functional significance of intrinsic oscillatory brain properties. *Front. Neural Circuits* 7:96. doi: 10.3389/fncir.2013.00096
- Loewenstein, Y., Mahon, S., Chadderton, P., Kitamura, K., Sompolinsky, H., Yarom, Y., et al. (2005). Bistability of cerebellar Purkinje cells modulated by sensory stimulation. *Nat. Neurosci.* 8, 202–211. doi: 10.1038/nn1393
- Long, M. A., Deans, M. R., Paul, D. L., and Connors, B. W. (2002). Rhythmicity without synchrony in the electrically uncoupled inferior olive. *J. Neurosci.* 22, 10898–10905. doi: 10.1523/jneurosci.22-24-10898.2002
- Manto, M., and Oulad Ben Taib, N. (2010). Cerebellar Nuclei: key roles for strategically located structures. *Cerebellum* 9, 17–21. doi: 10.1007/s12311-010-0159-8
- Marasco, A., Limongiello, A., and Migliore, M. (2012). Fast and accurate low-dimensional reduction of biophysically detailed neuron models. *Sci. Rep.* 2, 1–7. doi: 10.1038/srep00928
- Maruta, J., Hensbroek, R. A., and Simpson, J. I. (2007). Intraburst and interburst signaling by climbing fibers. *J. Neurosci.* 27, 11263–11270. doi: 10.1523/JNEUROSCI.2559-07.2007
- Masoli, S., and D'Angelo, E. (2017). Synaptic activation of a detailed purkinje cell model predicts voltage-dependent control of burst-pause responses in active dendrites. *Front. Cell. Neurosci.* 11:278. doi: 10.3389/fncel.2017.00278
- Masoli, S., Rizza, M. F., Sgritta, M., Van Geit, W., Schürmann, F., and D'Angelo, E. (2017). Single neuron optimization as a basis for accurate biophysical modeling:

- the case of cerebellar granule cells. *Front. Cell. Neurosci.* 11:14. doi: 10.3389/fncel.2017.00071
- Masoli, S., Solinas, S., and Angelo, E. D. (2015). Action potential processing in a detailed Purkinje cell model reveals a critical role for axonal compartmentalization. *Front. Cell. Neurosci.* 9:47. doi: 10.3389/fncel.2015.00047
- Mathy, A., Ho, S. S. N., Davie, J. T., Duguid, I. C., Beverley, A. C., and Häusser, M. (2009). Encoding of oscillations by axonal bursts in inferior olive neurons. *Neuron* 62, 388–399. doi: 10.1016/j.neuron.2009.03.023
- McKay, B. E., and Turner, R. W. (2005). Physiological and morphological development of the rat cerebellar Purkinje cell. *J. Physiol.* 567, 829–850. doi: 10.1113/jphysiol.2005.089383
- Miall, R. C., Keating, J. G., Malkmus, M., and Thach, W. T. (1998). Simple spike activity predicts occurrence of complex spikes in cerebellar Purkinje cells. *Nat. Neurosci.* 1, 13–15. doi: 10.1038/212
- Mihalaş, Ş., and Niebur, E. (2009). A generalized linear integrate-and-fire neural model produces diverse spiking behaviors. *Neural Comput.* 21, 704–718. doi: 10.1162/neco.2008.12-07-680
- Molineux, M. L., McRory, J. E., McKay, B. E., Hamid, J., Mehaffey, W. H., Rehak, R., et al. (2006). Specific T-type calcium channel isoforms are associated with distinct burst phenotypes in deep cerebellar nuclear neurons. *Proc. Natl. Acad. Sci.* 103, 5555–5560. doi: 10.1073/pnas.0601261103
- Rokni, D., Tal, Z., Byk, H., and Yarom, Y. (2009). Regularity, variability and bi-stability in the activity of cerebellar Purkinje cells. *Front. Cell. Neurosci.* 3:12. doi: 10.3389/fncel.2009.03.012.2009
- Rössert, C., Pozzorini, C., Chindemi, G., Davison, A. P., Eroev, C., King, J., et al. (2016). Automated point-neuron simplification of data-driven microcircuit models. *arXiv*
- Ruigrok, T. J. H. (2011). Ins and outs of cerebellar modules. *Cerebellum* 10, 464–474. doi: 10.1007/s12311-010-0164-y
- Sarnaik, R., and Raman, I. M. (2018). Control of voluntary and optogenetically perturbed locomotion by spike rate and timing of neurons of the mouse cerebellar nuclei. *eLife* 7:e29546. doi: 10.7554/eLife.29546
- Solinas, S., Forti, L., Cesana, E., Mapelli, J., De Schutter, E., and D'Angelo, E. (2007a). Computational reconstruction of pacemaking and intrinsic electroresponsiveness in cerebellar golgi cells. *Front. Cell. Neurosci.* 1:1. doi: 10.3389/fncel.2007.01.001
- Solinas, S., Forti, L., Cesana, E., Mapelli, J., De Schutter, E., and D'Angelo, E. (2007b). Fast-reset of pacemaking and theta-frequency resonance patterns in cerebellar golgi cells: simulations of their impact in vivo. *Front. Cell. Neurosci.* 1:4. doi: 10.3389/fncel.2007.01.004.2007
- Spanne, A., Geborek, P., Bengtsson, F., and Jörntell, H. (2014). Simulating spinal border cells and cerebellar granule cells under locomotion – a case study of spinocerebellar information processing. *PLoS One* 9:e107793. doi: 10.1371/journal.pone.0107793
- Steuber, V., Schultheiss, N. W., Silver, R. A., De Schutter, E., and Jaeger, D. (2011). Determinants of synaptic integration and heterogeneity in rebound firing explored with data-driven models of deep cerebellar nucleus cells. *J. Comput. Neurosci.* 30, 633–658. doi: 10.1007/s10827-010-0282-z
- Ten Brinke, M. M., Heiney, S. A., Wang, X., Proietti-Onori, M., Boele, H. J., Bakermans, J., et al. (2017). Dynamic modulation of activity in cerebellar nuclei neurons during pavlovian eyeblink conditioning in mice. *eLife* 6, 1–27. doi: 10.7554/eLife.28132
- Tripathy, S. J., Savitskaya, J., Burton, S. D., Urban, N. N., and Gerkin, R. C. (2014). NeuroElectro: a window to the world's neuron electrophysiology data. *Front. Neuroinform.* 8:40. doi: 10.3389/fninf.2014.00040
- Uusisaari, M., and Knöpfel, T. (2011). Functional classification of neurons in the mouse lateral cerebellar nuclei. *Cerebellum* 10, 637–646. doi: 10.1007/s12311-010-0240-3
- Uusisaari, M., Obata, K., Kno, T., and Morphological, T. (2007). Morphological and electrophysiological properties of GABAergic and Non-GABAergic cells in the deep cerebellar nuclei. *J. Neurophysiol.* 97, 901–911. doi: 10.1152/jn.00974.2006
- Van Der Giessen, R. S., Koekkoek, S. K., van Dorp, S., De Gruij, J. R., Cupido, A., Khosrovani, S., et al. (2008). Role of olivary electrical coupling in cerebellar motor learning. *Neuron* 58, 599–612. doi: 10.1016/j.neuron.2008.03.016
- Voogd, J., and Glickstein, M. (1998). The anatomy of the cerebellum. *Trends Cogn. Sci.* 2, 307–313. doi: 10.1016/0166-2236(95)93903-B
- Witter, L., Canto, C. B., Hoogland, T. M., de Gruij, J. R., and De Zeeuw, C. I. (2013). Strength and timing of motor responses mediated by rebound firing in the cerebellar nuclei after Purkinje cell activation. *Front. Neural Circuits* 7:133. doi: 10.3389/fncir.2013.00133
- Zhou, H., Lin, Z., Voges, K., Ju, C., Gao, Z., Bosman, L. W., et al. (2014). Cerebellar modules operate at different frequencies. *eLife* 3, 1–18. doi: 10.7554/eLife.02536

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Geminiani, Casellato, D'Angelo and Pedrocchi. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Corrigendum: Complex Electroresponsive Dynamics in Olivocerebellar Neurons Represented With Extended-Generalized Leaky Integrate and Fire Models

Alice Geminiani^{1*}, Claudia Casellato², Egidio D'Angelo^{2,3} and Alessandra Pedrocchi¹

¹ NEARLab, Department of Electronics, Information and Bioengineering, Politecnico di Milano, Milan, Italy, ² Department of Brain and Behavioral Sciences, University of Pavia, Pavia, Italy, ³ IRCCS Mondino Foundation, Pavia, Italy

OPEN ACCESS

Approved by:
Frontiers Editorial Office,
Frontiers Media SA, Switzerland

***Correspondence:**
Alice Geminiani
alice.geminiani@polimi.it

Received: 20 June 2019

Accepted: 26 June 2019

Published: 19 July 2019

Keywords: neuronal modeling, point neuron, neuron model simplification, neuronal electroresponsiveness, olivocerebellar neurons

A Corrigendum on

Complex Electroresponsive Dynamics in Olivocerebellar Neurons Represented With Extended-Generalized Leaky Integrate and Fire Models

by Geminiani, A., Casellato, C., D'Angelo, E., and Pedrocchi, A. (2019). *Front. Comput. Neurosci.* 13:35. doi: 10.3389/fncom.2019.00035

In the published article, there was an error regarding the affiliations for “Egidio D’Angelo.” As well as having affiliation 2, he should also have IRCCS Mondino Foundation, Pavia Italy. The authors apologize for this error and state that this does not change the scientific conclusions of the article in any way. The original article has been updated.

Citation:

Geminiani A, Casellato C, D'Angelo E and Pedrocchi A (2019) Corrigendum: Complex Electroresponsive Dynamics in Olivocerebellar Neurons Represented With Extended-Generalized Leaky Integrate and Fire Models. *Front. Comput. Neurosci.* 13:48. doi: 10.3389/fncom.2019.00048

Copyright © 2019 Geminiani, Casellato, D'Angelo and Pedrocchi. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



The Embodied Brain of SOVEREIGN2: From Space-Variant Conscious Percepts During Visual Search and Navigation to Learning Invariant Object Categories and Cognitive-Emotional Plans for Acquiring Valued Goals

Stephen Grossberg*

Center for Adaptive Systems, Graduate Program in Cognitive and Neural Systems, Departments of Mathematics & Statistics, Psychological & Brain Sciences, and Biomedical Engineering, Boston University, Boston, MA, United States

OPEN ACCESS

Edited by:

Mario Senden,
Maastricht University, Netherlands

Reviewed by:

Cees van Leeuwen,
KU Leuven, Belgium
Valeri Makarov,
Complutense University of Madrid,
Spain

*Correspondence:

Stephen Grossberg
steve@bu.edu

Received: 19 February 2019

Accepted: 21 May 2019

Published: 25 June 2019

Citation:

Grossberg S (2019) The Embodied Brain of SOVEREIGN2: From Space-Variant Conscious Percepts During Visual Search and Navigation to Learning Invariant Object Categories and Cognitive-Emotional Plans for Acquiring Valued Goals. *Front. Comput. Neurosci.* 13:36. doi: 10.3389/fncom.2019.00036

This article develops a model of how reactive and planned behaviors interact in real time. Controllers for both animals and animats need reactive mechanisms for exploration, and learned plans to efficiently reach goal objects once an environment becomes familiar. The SOVEREIGN model embodied these capabilities, and was tested in a 3D virtual reality environment. Neural models have characterized important adaptive and intelligent processes that were not included in SOVEREIGN. A major research program is summarized herein by which to consistently incorporate them into an enhanced model called SOVEREIGN2. Key new perceptual, cognitive, cognitive-emotional, and navigational processes require feedback networks which regulate resonant brain states that support conscious experiences of seeing, feeling, and knowing. Also included are computationally complementary processes of the mammalian neocortical What and Where processing streams, and homologous mechanisms for spatial navigation and arm movement control. These include: Unpredictably moving targets are tracked using coordinated smooth pursuit and saccadic movements. Estimates of target and present position are computed in the Where stream, and can activate approach movements. Motion cues can elicit orienting movements to bring new targets into view. Cumulative movement estimates are derived from visual and vestibular cues. Arbitrary navigational routes are incrementally learned as a labeled graph of angles turned and distances traveled between turns. Noisy and incomplete visual sensor data are transformed into representations of visual form and motion. Invariant recognition categories are learned in the What stream. Sequences of invariant object categories are stored in a cognitive working memory, whereas sequences of movement positions

and directions are stored in a spatial working memory. Stored sequences trigger learning of cognitive and spatial/motor sequence categories or plans, also called *list chunks*, which control planned decisions and movements toward valued goal objects. Predictively successful list chunk combinations are selectively enhanced or suppressed via reinforcement learning and incentive motivational learning. Expected vs. unexpected event disconfirmations regulate these enhancement and suppressive processes. Adaptively timed learning enables attention and action to match task constraints. Social cognitive joint attention enables imitation learning of skills by learners who observe teachers from different spatial vantage points.

Keywords: invariant object category learning, spatial navigation, visual search, working memory, reinforcement learning, motion perception, attention, adaptive resonance theory

1. PERCEPTION, LEARNING, INVARIANT RECOGNITION AND PLANNING DURING SEARCH AND NAVIGATION CYCLES

This article contributes to an emerging scientific and computational revolution aimed at understanding and designing increasingly autonomous adaptive intelligent algorithms and mobile agents. In particular, it summarizes an emerging neural architecture that is capable of visually searching and navigating an unfamiliar environment while it autonomously learns to recognize, plan, and efficiently navigate toward and acquire valued goal objects. This article accordingly reviews, and outlines how to extend, the SOVEREIGN architecture of Gnadt and Grossberg (2008) (**Figure 1A**). The purpose of that architecture is described in the subtitle of the article: *An autonomous neural system for incrementally learning planned action sequences to navigate towards a rewarded goal*.

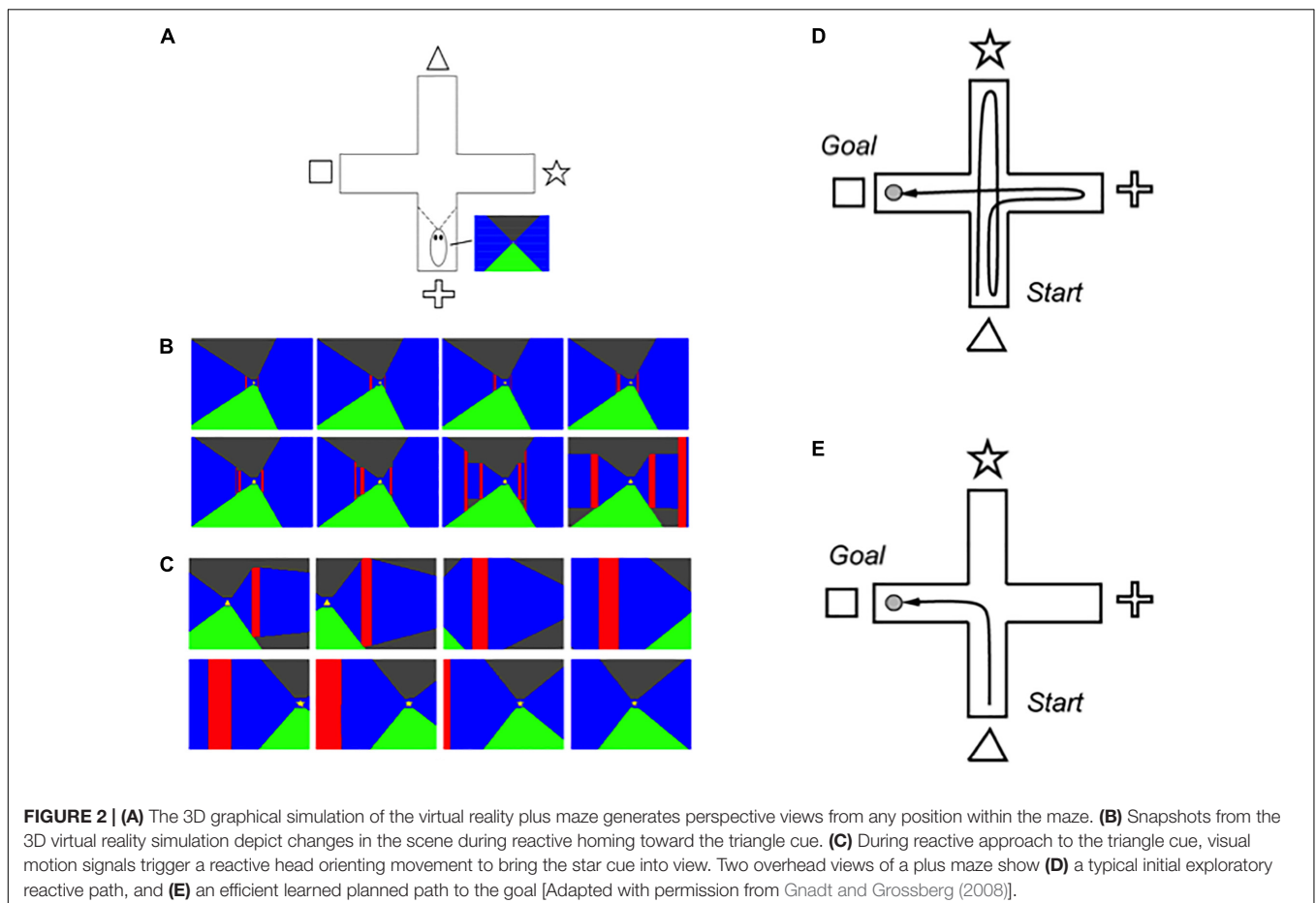
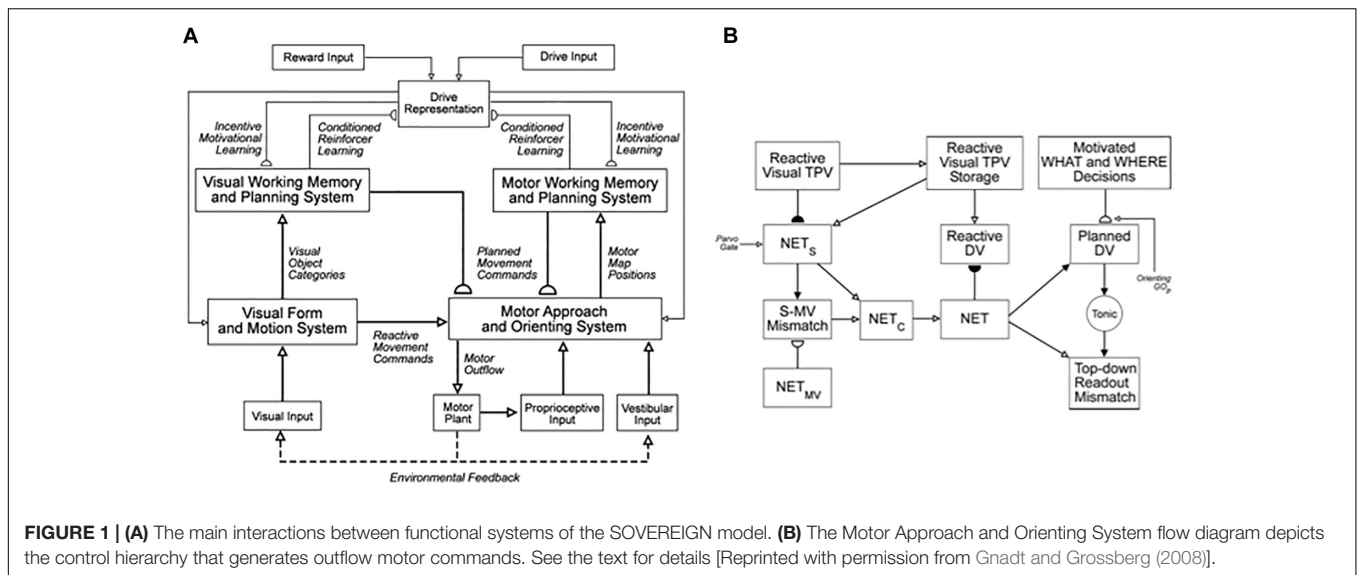
The architecture was called SOVEREIGN because it describes how Self-Organizing, Vision, Expectation, Recognition, Emotion, Intelligent, and Goal-oriented Navigation processes interact during adaptive mobile behaviors. The term *Self-Organizing* emphasizes that SOVEREIGN's learning is carried out autonomously and incrementally in real time, using unconstrained combinations of unsupervised or supervised learning. *Expectation* refers to the fact that key learning processes in SOVEREIGN learn expectations that match incoming data, or predict future outcomes. Good enough matches focus attention upon expected combinations of critical features, while mismatches drive memory searches to learn better representations of an environment. *Recognition* acknowledges that SOVEREIGN learns object categories, or "chunks," whereby to recognize objects and events. *Emotion* denotes that SOVEREIGN carries out reinforcement learning whereby unfamiliar objects can learn to become conditioned reinforcers, as well as sources of incentive motivation that can maintain attention upon valued goals, while actions to acquire those goals are carried out. Reinforcement learning also supports the learning of value categories that can recognize valued combinations of homeostatic drive inputs. *Intelligent* means that SOVEREIGN includes processes whereby sequences, or lists, of objects and positions may be temporarily stored in cognitive and spatial working memories as they are experienced in real

time. Stored sequences trigger learning of sequence categories or plans, also called *list chunks*, that recognize particular sequential contexts and learn to predict the most likely future outcomes as they are modulated by reinforcement learning and incentive motivational learning. *Goal-oriented navigation* means that SOVEREIGN includes circuits for controlling exploratory and planned movements while navigating unfamiliar and familiar environments.

1.1. Learning Routes as a Labeled Graph of Angles Turned and Distances Traveled

SOVEREIGN used these capabilities to simulate how an animal, or animat, can autonomously learn to reach valued goal objects through planned sequences of navigational movements within a virtual reality environment. Learning was simulated in a cross maze (**Figure 2A**) that was seen by the animat as a virtual reality 3D rendering of the maze as it navigated it through time. At the end of each corridor in the maze, a different visual cue was displayed (triangle, star, cross, and square). Sequences of virtual reality views on two navigational routes, shown in color for vividness, are summarized in **Figures 2B,C**, where the floor is green, the walls are blue, the ceiling in black, and the interior corners where pairs of maze corridors meet are in red. **Figure 2B** illustrates how the views change as the animat navigates straight down one corridor, and **Figure 2C** illustrates how the views change as the animal makes a turn from facing one corridor to facing a perpendicular one.

SOVEREIGN incrementally learned how to navigate to a rewarded goal object in this cross maze, which is the perhaps the simplest environment that requires all of the SOVEREIGN designs to explore an unfamiliar visual environment (**Figure 2D**) while learning efficient routes whereby to acquire a valued goal, rather than less efficient or valued routes (**Figure 2E**). Several different types of neural circuits, systems, and learning are needed to achieve this competence. They will be described in the subsequent sections. The same mechanisms generalize to much more complex visual environments, especially because, as will be described below, all the perceptual, cognitive, and affective learning mechanisms scale to more complex environments and dynamically self-stabilize their memories using learned expectation and attention



mechanisms, while the spatial and motor mechanisms are platform independent.

One key SOVEREIGN accomplishment is worthy of mention now because it illustrates how SOVEREIGN goes beyond reactive navigation to autonomously learn the most efficient

routes whereby to acquire a valued goal, while rejecting less efficient routes that were taken early in the exploratory process. SOVEREIGN explains how arbitrary navigational trajectories can be incrementally learned as sequences of turns and linear movements until the next turn. In other words, the model

explains how route-based navigation can learn a *labeled graph* of *angles* turned and *distances* that are traveled between turns. The angular and linear velocity signals that are experienced at such times are used in the model to learn the angles that a navigator turns, and the distances that are traveled in a straight path before the next turn.

The prediction that a labeled graph is learned during route navigation has recently received strong experimental support in Warren et al. (2017) who show how, when humans navigate in a virtual reality environment, such a labeled graph controls their navigational choices during route finding, novel detours, and shortcuts.

1.2. From SOVEREIGN to SOVEREIGN2: New Processes and Capabilities

SOVEREIGN did not include various brain processes and psychological functions of humans that are needed to realize a more sophisticated level of autonomous adaptive intelligence. This article summarizes some of the neural models that have been developed to explain these functions, and that can be consistently incorporated into an enhanced architecture called SOVEREIGN2. These processes have been rigorously modeled and parametrically simulated over a 40-year period, culminating in recent syntheses such as Grossberg (2013, 2017, 2018). They are reviewed heuristically here to bring together in one place the basic design principles, mechanisms, and architectures that they embody. Rigorous embodiment of all of these competences in SOVEREIGN2 will require a sustained research program. The current article provides a roadmap for that task.

The most important new perceptual, cognitive, and navigational properties emerge within feedback networks that regulate one or another kind of attention as part of resonant brain states that support conscious experiences of seeing, feeling, and knowing. These resonant states are modeled as part of Adaptive Resonance Theory, or ART. **Table 1a** also lists resonances that arise during auditory processing. Auditory processing will not be considered below, but is described with the others in Grossberg (2017). SOVEREIGN2 will embody such resonant dynamics, including states that in humans support consciousness, because of a deep computational connection that has been modeled between conscious states and the choice of effective task-relevant actions. ART hereby provides explanations of *what* goes on in each of our brains when we consciously see, hear, feel, or know something; *where* it is going on; and *why* evolution may have been driven to discover conscious states of mind.

Additional processes in SOVEREIGN2 include circuits for target tracking with smooth pursuit and saccadic eye or camera movements (see section 3.2); visual form and motion perception in response to noisy and incomplete sensor signals (see section 4.13); incremental unsupervised view-, size-, and position-specific object category learning and hypothesis testing in real time in response to arbitrarily large non-stationary databases that may include unexpected events (see sections 4.2–4.9, 6.2, and 6.3); incremental unsupervised learning of view-, size-, and position-invariant object categories during free scanning of a scene with eye or camera movements (see sections 4.1, 6.1,

TABLE 1 | (a) Types of resonances and the conscious experiences that they embody. **(b)** Complementary What and Where cortical stream properties.

TYPE OF RESONANCE	TYPE OF CONSCIOUSNESS
surface-shroud	see visual object or scene
feature-category	recognize visual object or scene
stream-shroud	hear auditory object or stream
spectral-pitch-and-timbre	recognize auditory object or stream
item-list	recognize speech and language
cognitive-emotional	feel emotion and know its source
(a)	
WHAT	WHERE
Spatially-invariant object learning and recognition	Spatially-variant reaching and movement
Fast learning without catastrophic forgetting	Continually update sensory-motor maps and gains
IT	PPC
MATCHING LEARNING	WHAT
	WHERE
	EXCITATORY
	INHIBITORY
	MATCH
	MISMATCH

Cortical What stream perceptual and cognitive representations can solve the stability-plasticity dilemma, using brain regions like inferotemporal (IT) cortex, where recognition categories are learned. These processes carry out excitatory matching and match-based learning. Cortical Where stream spatial and motor processes do not solve the stability-plasticity dilemma, but rather adapt to changing bodily parameters, using brain regions like posterior parietal cortex (PPC). Whereas the recognition categories in the cortical What stream become increasingly invariant at higher cortical levels with respect to object views, positions, and sizes, the cortical Where stream elaborates spatial representations of object positions and mechanisms whereby to act upon them. Together the two streams can learn to recognize and become conscious of valued objects and scenes, while directing appropriate actions toward them [Reprinted with permission from Grossberg (2017)].

and 6.4); selective storage in working memory of task-relevant object, spatial, or motor event sequences (see sections 4.10, 6.9, 6.10, and 7); unsupervised learning of cognitive and motor plans based upon working memory storage of event sequences in real time, and Where's Waldo search for currently valued goal objects (see sections 6.10 and 7); unsupervised learning of reaching behaviors that automatically supports accurate tool manipulation in space (see section 5.4); unsupervised learning of present position in space using path integration during spatial navigation (see sections 6.11 and 8); platform-independent navigational control using either leg or wheel movements (see section 5.6); unsupervised learning of adaptively timed actions and maintenance of motivated attention while these actions are executed (see sections 6.7 and 6.8); and social cognitive capabilities like joint attention and imitation learning whereby a classroom of robots can learn spatial skills by each observing a teacher from its own unique spatial perspective (see section 5.5).

2. BRAINS ASSEMBLE EQUATIONS AND MICROCIRCUITS INTO MODAL ARCHITECTURES: CONTRAST DEEP LEARNING

ART architectures embody key design principles that are found in advanced brains, and which enable general-purpose autonomous adaptive intelligence to work. These designs have

enabled biological neural networks to offer unified principled explanations of large psychological and neurobiological databases (e.g., see Grossberg, 2013, 2017, 2018) using just a small set of mathematical *laws* or equations—such as the laws for short-term memory or STM, medium-term memory or MTM, and long-term memory or LTM—and a somewhat larger set of characteristic *microcircuits* that embody useful combinations of functional properties—such as properties of cognitive and cognitive-emotional learning and memory, decision-making, prediction, and action. Just as in physics, only a few basic equations are used to explain and predict many facts about mind and brain, when they are embodied in a somewhat larger number of microcircuits that may be thought of as the “atoms” or “molecules” of intelligence. Specializations of these laws and microcircuits are then combined into larger systems that are called *modal architectures*, where the word “modal” stands for different modalities of intelligence, such as vision, speech, cognition, emotion, and action. Modal architectures are less general than a general-purpose von Neumann computer, but far more general than a traditional algorithm from AI.

As I will illustrate throughout this article, these designs embody computational paradigms that are called *complementary computing*, *hierarchical resolution of uncertainty*, and *adaptive resonance*. In addition, the paradigm of *laminar computing* shows how these designs may be realized in the layered circuits of the cerebral cortex and, in so doing, achieve even more powerful computational capabilities. These computational paradigms differ qualitatively from currently popular algorithms in AI and machine learning, notably Deep Learning (Hinton et al., 2012; LeCun et al., 2015) and its variants like Deep Reinforcement Learning (Mnih et al., 2013). Despite their successes in demonstrating various recent applications, these algorithms do not come close to matching the generality, adaptability, and intelligence that is found in models that more closely emulate brain designs. As just one of many problems, Deep Learning algorithms are susceptible to undergoing catastrophic forgetting, or an unexpected collapse of the memory of previously learned information while new information is being learned, a property that is shared by all variants of the classical back propagation algorithm (Grossberg, 1988). This kind of problem becomes increasingly destructive as a Deep Learning algorithm tries to learn from very large databases. The ART-based systems that are summarized below do not experience these problems.

No less problematic is that Deep Learning is just a feedforward adaptive filter. It does not carry out any of the basic kinds of information processing that are typically identified as “intelligent,” but which are carried out within ART and other biological learning algorithms that are embedded within neural network architectures. Deep Learning has none of the architectural features, such as learned top-down expectations, attentional focusing, and mismatch-mediated memory search and hypothesis testing, that are needed for stable learning in a non-stationary world of Big Data.

Perhaps these problems are why Geoffrey Hinton said in an *Axios* interview on September 15, 2017 (LeVine, 2017) that he is “deeply suspicious of back propagation. . . I don’t think it’s how the brain works. We clearly don’t need all the *labeled data*. . . My

view is, *throw it all away and start over*” (italics mine). This essay illustrates that we do not need to start over.

Section 17 in Grossberg (1988) lists 17 different learning and performance properties of Back Propagation and Adaptive Resonance Theory. The third of the 17 differences between Back Propagation and ART is that ART does not need labeled data to learn. ART can learn using arbitrary combinations of unsupervised and supervised learning. ART also does not experience any of the computational problems that compromise Back Propagation and Deep Learning, including catastrophic forgetting.

3. BUILDING UPON THREE BASIC DESIGN THEMES: BALANCING REACTIVE AND PLANNED BEHAVIORS

The original SOVEREIGN architecture contributed models of three basic design themes about how advanced brains work. The first theme concerns how brains learn to balance between reactive and planned behaviors. During initial exploration of a novel environment, many reactive movements may occur in response to unfamiliar and unexpected environmental cues (Leonard and McNaughton, 1990). These movements may seem initially to be random, as an animal orients toward and approaches many stimuli (**Figure 2D**). As the animal becomes familiar with its surroundings, it learns to discriminate between objects likely to yield a reward and those that lead to punishment or to no valued consequences. Such approach-avoidance behavior is typically learned via reinforcement learning during a perception-cognition-emotion-action cycle in which an action and its consequences elicit sensory cues that are associated with them. When objects are out of direct viewing or reaching ranges, reactive exploratory movements may be triggered to bring them closer. Eventually, reactive exploratory behaviors are replaced by more efficient planned sequential trajectories within a familiar environment (**Figure 2E**). One of the main goals of SOVEREIGN was to explain how erratic reactive exploratory behaviors trigger learning to carry out organized planned behaviors, and how both reactive and planned behaviors may remain balanced so that planned behaviors can be carried out where appropriate, without losing the ability to respond quickly to novel reactive challenges.

3.1. Parallel Streams for Computing Visual Form and Motion

One way that SOVEREIGN realizes a flexible balance between reactive and planned behaviors is its organization into parallel streams for computing visual form and motion. In **Figure 3A**, these streams are labeled PARVO and MAGNO, corresponding to contributions at early visual processing stages of parvocellular cells to form processing and magnocellular cells to motion processing (e.g., Maunsell and Newsome, 1987; DeYoe and Van Essen, 1988; Maunsell et al., 1990; Schiller et al., 1990). Roughly speaking, the form stream supports sustained attention upon foveated objects, whereas the motion stream attracts attention and bodily movements in response to sudden changes,

including motions, in the periphery. sections 3.2 and 4.13 will further describe how SOVEREIGN carries out form processing and will outline how SOVEREIGN2 can achieve much more powerful form processing capabilities. **Figure 3B** provides a more detailed summary of the early motion processing that enables SOVEREIGN to track objects moving at variable speeds (Chey et al., 1997; Berzhanskaya et al., 2007). Orienting movements to a source of motion were controlled algorithmically in SOVEREIGN; e.g., see the Head-Orienting Movement Module in **Figure 3A**.

3.2. Log Polar Retinas and Fixating Unpredictably Moving Targets With Eye Movements

Many primate retinas have a localized region of high visual acuity that is called the *fovea*, with resolution decreasing with distance from the fovea (see **Supplementary Figure S4**) to realize a *cortical magnification factor* whereby spatial representations of retinal inputs in the visual cortex get coarser as they move from the foveal region to the periphery (Daniel and Whitteridge, 1961; Fischer, 1973; Tootell et al., 1982; Schwartz, 1984; Polimeni et al., 2006). The cortical magnification factor is approximated by a log-polar function, which allows a huge reduction in the number of cells that are needed to see Schwartz (1984), Wallace et al. (1984), Schwartz et al. (1995). However, because of this retinal organization, eye and head movements are needed to move the fovea to look at objects of interest.

Both smooth pursuit movements and saccadic eye movements are used to keep the fovea looking at objects of interest. During a smooth pursuit movement, as the eyes track a moving target in a given direction, the entire scene moves in the opposite direction on the retina (**Supplementary Figure S1**). Why does not this background motion interfere with tracking by causing an involuntary motion, called nystagmus, in the opposite direction than the target is moving? How does accurate tracking continue, even after the eye catches up with the moving target, so that there is no net speed of the target on the fovea, and thus no retinal slip signals from the foveal region of the eyes to move them toward the target?

Remarkably, both of these questions seem to have the same answer, which includes the fact that the background motion facilitates tracking, rather than interfering with it, in the manner that is summarized in **Supplementary Figures S1, S2**. **Supplementary Figure S1** summarizes the fact that, for fixed target speed, as the target speed on the retina decreases due to increasingly good target tracking, the background speed in the opposite direction on the retina increases. **Supplementary Figure S2** schematizes the smooth pursuit eye movement, or SPEM, model of Pack et al. (2001) of how cells in the dorsal Medial Superior Temporal region (MSTd), which are activated by the background motion, excite cells that are sensitive to the *opposite* direction in the ventral MST (MSTv) region. The MSTv cells are the ones that control the movement commands whereby the eyes pursue the moving target. When the eyes catch up to the target, they can maintain accurate foveation even in the absence of retinal slip signals, because background motion

signals compensate for the reduced retina speed of the target, and can thus be used to accurately move the eyes in the desired direction at the target speed (**Supplementary Figure S1**). This kind of SPEM model can replace the Head-Orienting Movement Module in SOVEREIGN if an animat with orienting eyes or cameras is used.

When a valued target suddenly changes its speed or direction of motion, then smooth pursuit movements may be insufficient. Ballistic saccadic movements can then catch up with the target. Animals such as humans and monkeys can coordinate smooth pursuit and ballistic saccadic eye movements to catch up efficiently. Indeed, the current speed and direction of smooth pursuit when the target suddenly changes its speed or direction may be used to calibrate a ballistic saccade with the best chance to catch up. This kind of predictive coordination is achieved by the SAC-SPEM model of Grossberg et al. (2012). The sheer number of brain regions that work together to accomplish such coordination (**Supplementary Figure S3**) will challenge future mobile robotic designers to embody this tracking competence in the simplest possible way.

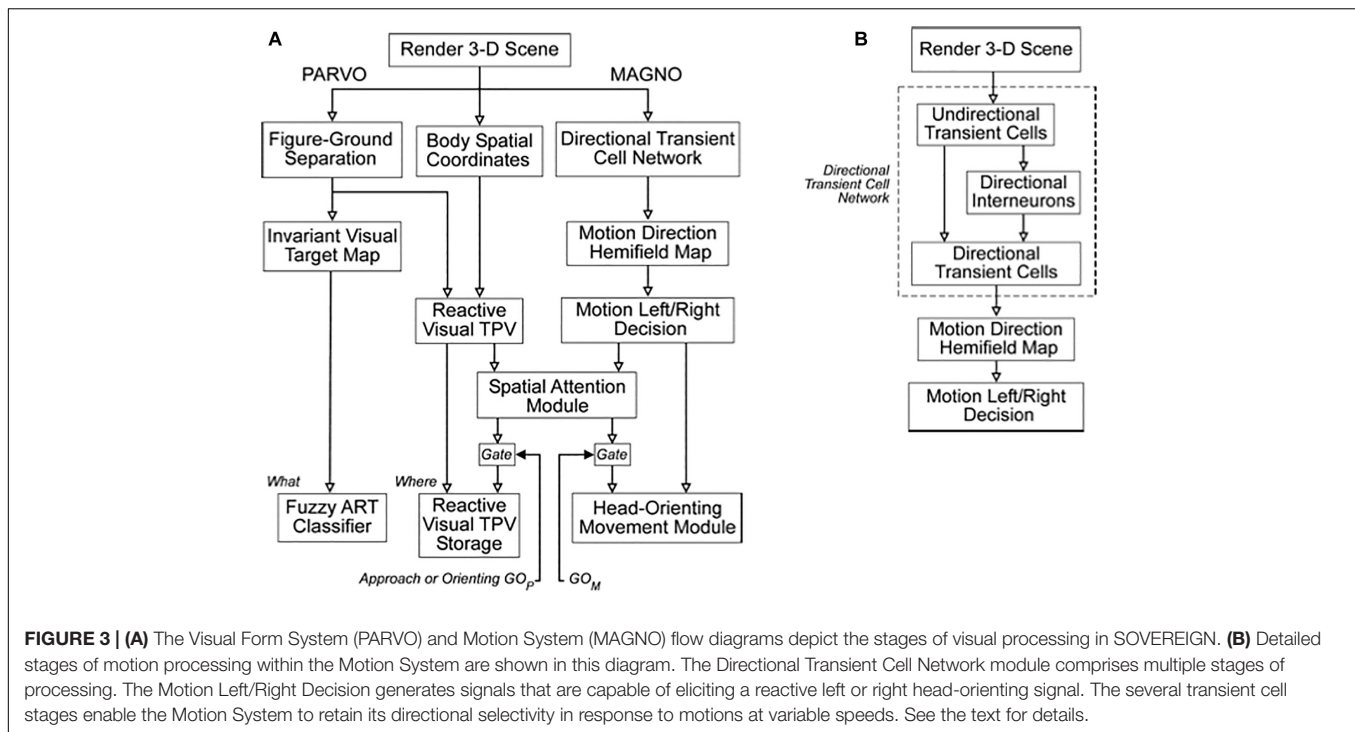
4. BUILDING UPON THREE BASIC DESIGN THEMES: COMPLEMENTARY COMPUTING, HIERARCHICAL RESOLUTION OF UNCERTAINTY, AND ADAPTIVE RESONANCE

The second design theme is that advanced brains are organized into parallel processing streams with computationally complementary properties (Grossberg, 2000, 2017). Complementary computing means that each stream's properties are related to those of a complementary stream much as a key fits into a lock, or two pieces of a puzzle fit together. The mechanisms that enable each stream to compute one set of properties prevent it from computing a complementary set of properties. As a result, each of these streams exhibits complementary strengths and weaknesses. Interactions between these processing streams use multiple processing stages to overcome their complementary deficiencies and generate psychological properties that lead to successful behaviors. This interactive multi-stage process is called *hierarchical resolution of uncertainty*.

Two of these complementary streams are the ventral What cortical stream for object perception and recognition, and the dorsal Where (or Where/How) cortical processing stream for spatial representation and action (Ungerleider and Mishkin, 1982; Mishkin, 1982; Mishkin et al., 1983; Goodale et al., 1991; Goodale and Milner, 1992). Key properties of these cortical processing streams have been shown to be computationally complementary (**Table 1b**).

4.1. Invariant Object Category Learning

One of several basic reasons for this particular kind of complementarity is that the cortical What stream learns object recognition categories that become substantially invariant under changes in an object's view, size, and position at higher



cortical processing stages, such as at the anterior inferotemporal cortex (ITa) and beyond (Tanaka, 1997, 2000; Booth and Rolls, 1998; Fazl et al., 2009; Cao et al., 2011; Chang et al., 2014). These invariant object categories have a compact representation that enables valued objects to be recognized without causing the combinatorial explosion that would have occurred if our brains needed to store every individual exemplar of every object that was ever experienced. However, because they are invariant, these categories cannot, by themselves, locate and act upon a desired object in space. Cortical Where stream spatial and motor representations can locate objects and trigger actions toward them, but cannot recognize them. By interacting together, the What and Where streams can consciously see and recognize valued objects *and* direct appropriate goal-oriented actions toward them.

The original SOVEREIGN model explained simple properties of how such invariant categories are learned as an animal, or animat, explores a novel environment. It used log-polar preprocessing of input images, followed by coarse-coding and algorithmic shift operations, to generate size-invariant and position-invariant input images. These preprocessed images were then input to a Fuzzy ART classifier (Carpenter et al., 1991b) for learning invariant visual 2D view-specific categories whereby SOVEREIGN could recognize an object at variable distances. These view-specific categories were converted into categories that were view-invariant, as well as positionally invariant and size-invariant, by algorithmically associating multiple view-specific categories with a shared view-invariant category (Figure 4A).

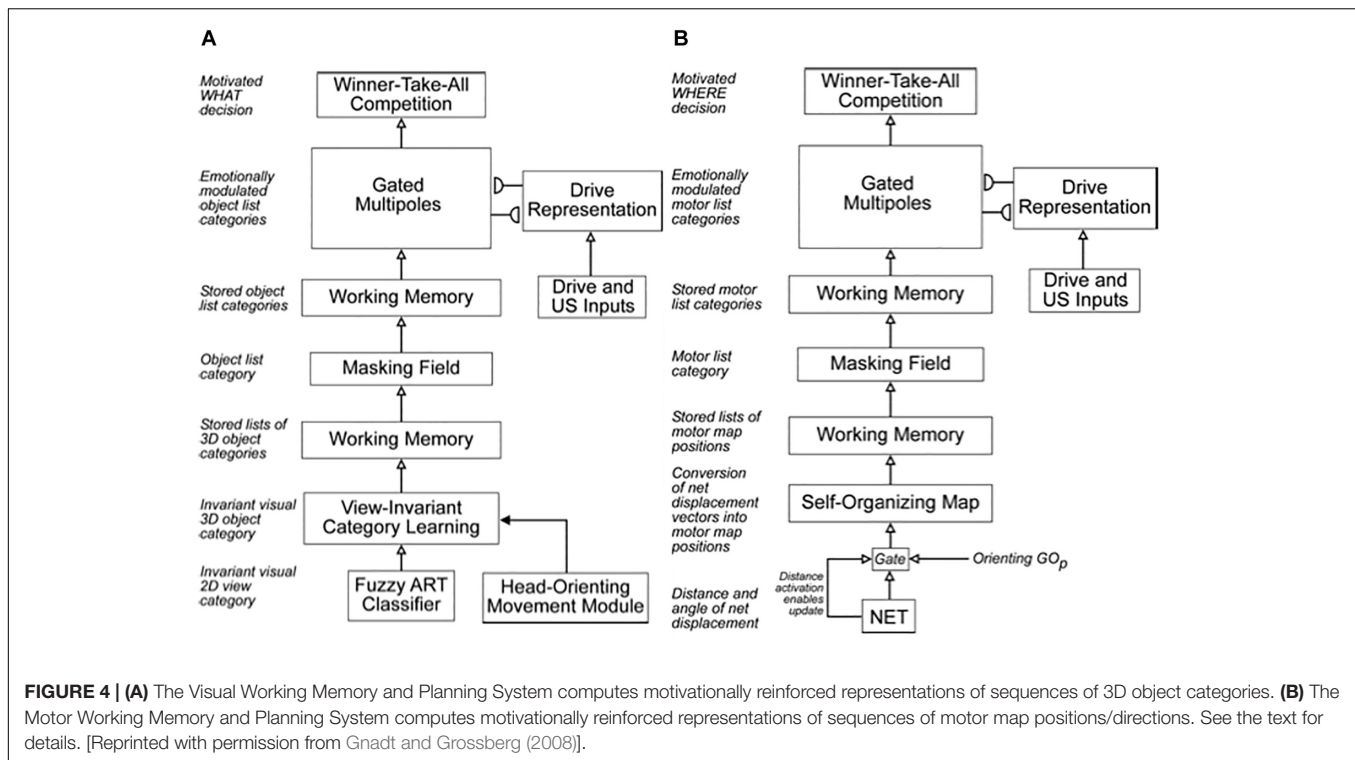
Since SOVEREIGN was published, the 3D ARTSCAN SEARCH model was developed to explain how humans and other primates may accomplish incremental unsupervised learning

of view-, position-, and size-invariant categories, without any algorithmic shortcuts, and how these invariant categories can be used to trigger a cognitively or motivationally driven Where's Waldo search for a desired object in a cluttered scene (Fazl et al., 2009; Grossberg, 2009b; Cao et al., 2011; Foley et al., 2012; Chang et al., 2014; Grossberg et al., 2014). These important *Recognition* and *Where's Waldo* search capabilities, which will be further discussed in sections 6.1 and 6.4, can also be incorporated into SOVEREIGN2 instead of the bottom two category learning processes in Figure 4A.

4.2. Adaptive Resonance Theory: A Universal Design for Autonomous Classification and Prediction

The ART in the Fuzzy ART algorithm abbreviates Adaptive Resonance Theory, which was introduced in 1976 (Grossberg, 1976a,b) and developed into the most advanced cognitive and neural theory of how advanced brains learn to attend, recognize, and predict objects and events in complex changing environments that may be filled with unexpected events. ART currently has an unrivalled explanatory and predictive range about how processes of consciousness, learning, expectation, attention, resonance, and synchrony interact in advanced brains. Along the way, all of the foundational hypotheses of ART have been confirmed by later psychological and neurobiological experiments. See Grossberg (2013, 2017, 2018) for recent reviews and syntheses.

ART's significance is highlighted by the fact that its design principles and mechanisms can be derived from a *thought experiment* whose simple assumptions are familiar to us all as facts that we experience ubiquitously in our daily lives.



These facts embody environmental constraints which, taken together, define a multiple constraint problem that evolution has solved in order to enable humans and other higher animals to be able to autonomously learn to attend, recognize, and predict their unique and changing worlds. Such a competence is essential in autonomous adaptive mobile agents, which is why some ART algorithms were already algorithmically implemented in SOVEREIGN.

4.3. Predictive Brain: Intention, Attention, and Resonance Solve the Stability-Plasticity Dilemma

One of the critical properties of ART that enable it to support open-ended incremental autonomous learning is that resonant states can trigger rapid learning about a changing world while solving the *stability-plasticity dilemma*. This dilemma asks how can our brains learn quickly without being forced to forget previously learned, but still useful, memories just as quickly?

The stability-plasticity dilemma was articulated before the *catastrophic forgetting* problem was stated (French, 1999), and clarifies that it is a problem of *balance* between fast learning and stable memory. Catastrophic forgetting means that an unpredictable part of previously learned memories can rapidly and unpredictably collapse during new learning. This problem becomes particularly acute when learning any kind of Big Data problem, notably during the kind of open-ended incremental learning that an autonomous adaptive robot might need to do as it navigates unfamiliar environments. A catastrophic collapse of previous memories while trying to completely learn about a huge database, not to mention a database that is continually

changing through time, is intolerable in any application that can have serious real world consequences. Popular machine learning algorithms such as Back Propagation and its recent variant, Deep Learning (Hinton et al., 2012; LeCun et al., 2015), do not solve the catastrophic forgetting problem. In brief, Deep Learning is unreliable.

A resonant brain state is a dynamical state during which neuronal firings across a brain network are amplified and synchronized when they interact via reciprocal excitatory feedback signals during an attentive matching process that occurs between bottom-up and top-down pathways. In the case of learning recognition categories, the bottom-up pathways are adaptive filters that tune their adaptive weights, or LTM traces, to more reliably activate the category that best matches the input feature patterns that activate them. The top-down pathways are learned recognition expectations whose LTM traces focus attention upon a prototype of critical features that best predict the active category. As will be explained in greater detail below (see **Figure 7** below), a resonance of this kind is called a *feature-category resonance* in order to distinguish it from the multiple other kinds of resonances that dynamically stabilize learning in different brain systems.

A resonance represents a system-wide consensus that the attended information is worthy of being learned. It is because resonances can trigger fast learning that they are called *adaptive resonances*, and why the theory that explicates them is called Adaptive Resonance Theory. ART's proposed solution of the stability-plasticity dilemma mechanistically links the process of stable learning and memory with the mechanisms of Consciousness, Expectation, Attention, Resonance, and

Synchrony that enable it. Due to their mechanistic linkage, these processes are often abbreviated as the CLEARS processes.

ART hereby predicts that interactions among CLEARS mechanisms solve the stability-plasticity dilemma. That is why humans and other higher animals are *intentional* and *attentional* beings who use learned expectations to pay attention to salient objects and events, why “all conscious states are resonant states,” and how brains can learn both *many-to-one maps* (representations whereby many object views, positions, and sizes learn to activate the same invariant object category), and *one-to-many maps* (learned representations that enable us to expertly know many things about individual objects and events).

As will be explained in greater detail below, the link between Consciousness, Learning, and Resonance is a particularly important one for understanding both characteristically human experiences and how future machine learning algorithms may embody them.

4.4. Object Attention Dynamically Stabilizes Learning Using the ART Matching Rule

ART solves the stability-plasticity dilemma by using learned expectations and attentional focusing to *selectively* process only those data that are predicted to be relevant in any given situation. Because of the CLEARS relationships, such selective attentive processing also solves the stability-plasticity dilemma.

For this to work, the correct laws of object attention need to be used. ART has predicted how object attention is realized in human and other advanced primate brains (e.g., Grossberg, 1980, 2013; Carpenter and Grossberg, 1987a, 1991). In order to dynamically stabilize learning, the learned expectations that focus attention obey a *top-down, modulatory on-center, off-surround* network. This network is said to obey the ART Matching Rule.

In such a network, when a bottom-up input pattern is received at a processing stage, it can activate its target cells, if nothing else is happening. When a top-down expectation pattern is received at this stage, it can provide excitatory modulatory, or priming, signals to cells in its on-center, and driving inhibitory signals to cells in its off-surround, if nothing else is happening. The on-center is *modulatory* because the off-surround network also inhibits the on-center cells, and these excitatory and inhibitory inputs are approximately balanced (“one-against-one”). When a bottom-up input pattern and a top-down expectation are both active, cells that receive both bottom-up excitatory inputs and top-down excitatory priming signals can fire (“two-against-one”), while other cells are inhibited. In this way, only cells can fire whose features are “expected” by the top-down expectation, and an attentional focus starts to form at these cells. As a result only attended feature patterns are learned. The system wherein category learning takes place is thus called an *attentional system*.

The property of the ART Matching Rule that bottom-up sensory activity may be enhanced when matched by top-down signals is in accord with an extensive neurophysiological literature showing the facilitatory effect of attentional feedback (e.g., Sillito et al., 1994; Luck et al., 1997; Roelfsema et al., 1998).

This property contradicts popular models, such as Bayesian Explaining Away models, in which matches with top-down feedback cause only suppression (Mumford, 1992; Rao and Ballard, 1999). A related problem is that suppressive matching circuits cannot solve the stability-plasticity dilemma.

An ART expectation is a *top-down, adaptive, and specific* event that activates its target cells during a *match* within the attentional system. “Adaptive” means that the top-down pathways contain adaptive weights that can learn to encode a prototype of the recognition category that activates it. “Specific” means that each top-down expectation reads out its learned prototype pattern. One psychophysiological marker of such a resonant match is the processing negativity, or PN, event-related potential (Grossberg, 1978, 1984b; Näätänen, 1982; Banquet and Grossberg, 1987).

4.5. ART Is a Self-Organizing Production System: Lifelong Learning of Expertise

The above properties of an expectation are italicized because, as will be seen below, they are *computationally complementary* to those of an *orienting system* that enables ART to autonomously learn about arbitrarily many novel events in a non-stationary environment without experiencing catastrophic forgetting. As will be explained more fully below, if a top-down expectation mismatches an incoming bottom-up input pattern too much, the orienting system is activated and drives a memory search and hypothesis testing for either a better-matching category if the input represents information that is familiar to the network, or a novel category if it is not.

Taken together, the ART attentional and orienting systems constitute a *self-organizing production system* that can learn to become increasingly expert about the world that it experiences throughout the life span of the individual or machine into which it is embedded.

4.6. ART Can Carry Out Open-Ended Stable Learning of Huge Non-stationary Databases

Our ability to achieve learning throughout life can be stated in another way that emphasizes its critical importance in human societies no less than in designing autonomous adaptive robots with real intelligence: Without stable memories of past experiences, we could learn very little about the world, since our present learning would wash away previous memories unless we continually rehearsed them. But if we had to continuously rehearse everything that we learned, then we could learn very little, because there is just so much time in a day to rehearse. Having an active top-down matching mechanism greatly amplifies the amount of information that humans can quickly learn and stably remember about the world. This capability, in turn, sets the stage for developing a sense of self, which requires that we can learn and remember a record of many experiences that are uniquely ours over a period of years.

With appropriately implemented ART algorithms on board, a SOVEREIGN2 robot can continue to learn indefinitely for its entire lifespan.

4.7. Large-Scale Machine Learning Applications in Engineering and Technology

ART enables a general-purpose category learning, recognition, and prediction capability that has already been used in multiple large-scale applications in engineering and technology. When it is embodied completely enough in SOVEREIGN2, then SOVEREIGN2 can also be used to carry out such applications, and can do so with the advantage being able to navigate environments where these applications occur.

Fielded applications include: airplane design (including the Boeing 777); medical database diagnosis and prediction; remote sensing and geospatial mapping and classification; multidimensional data fusion; classification of data from artificial sensors with high noise and dynamic range (synthetic aperture radar, laser radar, multi-spectral infrared, night vision); speaker-normalized speech recognition; sonar classification; music analysis; automatic rule extraction and hierarchical knowledge discovery; machine vision and image understanding; mobile robot controllers; satellite remote sensing image classification; electrocardiogram wave recognition; prediction of protein secondary structure; strength prediction for concrete mixes; tool failure monitoring; chemical analysis from ultraviolet and infrared spectra; design of electromagnetic systems; face recognition; familiarity discrimination; and power transmission line fault diagnosis. Some of these applications are listed at <http://techlab.bu.edu/resources/articles/C5/>.

4.8. Mathematically Provable ART Learning Properties Support Large-Scale Applications

It is because the good learning properties of ART have been mathematically proved and tested with comparative computer simulation benchmarks that ART has been used with confidence in these applications (e.g., Carpenter and Grossberg, 1987a,b, 1990; Carpenter et al., 1989, 1991a,b, 1992, 1998).

These theorems prove how ART can rapidly learn, from arbitrary combinations of unsupervised and supervised trials, to categorize complex, and arbitrarily large, non-stationary databases, dynamically stabilize their learned memories, directly access the globally best matching categories with no search during recognition, and use these categories to predict the most likely outcomes in a given situation.

In particular, ART provably solved the catastrophic forgetting problem that other approaches to machine learning have failed to solve.

4.9. ART Solves the Explainable AI Problem and Extracts Knowledge Hierarchies From Data

ART offers a solution of another problem that other researchers in machine learning and AI are still seeking. The learned weights of the fuzzy ARTMAP algorithm (Carpenter et al., 1992) translate, at any stage of learning, into fuzzy IF-THEN rules that “explain” why the learned predictions work. Understanding

why particular predictions are made is no less important than their predictive success in applications that have life or death consequences, such as medical database diagnosis and prediction, to which ART has been successfully applied. This problem has not yet been solved in traditional AI, as illustrated by the current DARPA Explainable AI program (XAI¹).

In addition, ART can self-organize hierarchical knowledge structures from masses of incomplete and partially incompatible data taken from multiple observers who do not communicate with each other, and who may use different combinations of object names and sensors to incrementally collect their data at different times, locations, and scales (Carpenter et al., 2005; Carpenter and Ravindran, 2008). If swarms of SOVEREIGN2 robots collect data in this distributed way, then they can share it wirelessly to self-organize such cognitive hierarchies of rules.

4.10. Cognitive and Spatial Working Memories and Plans

Figure 4A also summarizes higher cognitive and cognitive-emotional processes that are modeled in SOVEREIGN. Together with Figure 4B, these contribute to SOVEREIGN's *Intelligent and Goal-oriented navigation* processing whereby cognitive working memories (Figure 4A) and spatial working memories (Figure 4B) provide the information whereby cognitive plans (Figure 4A) and spatial plans (Figure 4B) are learned and used to control actions to acquire valued goals. The cognitive working memory temporarily stores the temporal order of sequences of invariant object categories that represent recently experienced objects. These sequences are themselves categorized during learning of cognitive plans, or *list chunks*, that fire selectively in response to particular stored object sequences. Such a network of list chunks is called a Masking Field (Grossberg, 1978; Cohen and Grossberg, 1986, 1987; Grossberg and Myers, 2000; Grossberg and Kazeroonian, 2011; Kazeroonian and Grossberg, 2014). The corresponding spatial working memory and Masking Field in Figure 4B do the same thing for the stored sequences of navigational movements—notably combinations of turns and straight excursions in space—that SOVEREIGN carries out while exploring the maze. These processes will be discussed further in sections 6.9, 6.10, and 7, notably how they need to be enhanced in SOVEREIGN2 to achieve selective processing and storage of only task-relevant sequences of information.

4.11. Reinforcement Learning and Incentive Motivation to Acquire Valued Goals

These cognitive and spatial processes do not themselves compute indices of predictive success and failure. The processes that accomplish goal-oriented selectivity—including gated multipoles and drive representations—occur next (See Figure 12 below). These reinforcement learning and incentive motivational processes enable SOVEREIGN to select, amplify, and sustain in working memory those previous event sequences that have led to predictive success in the past, and to use these list

¹<https://www.darpa.mil/program/explainable-artificial-intelligence>

categories to predict the actions most likely to achieve valued goals in the future. These processes will be further discussed in sections 6.5–6.7.

4.12. Prefrontal Regulation of Cognitive and Cognitive-Emotional Dynamics

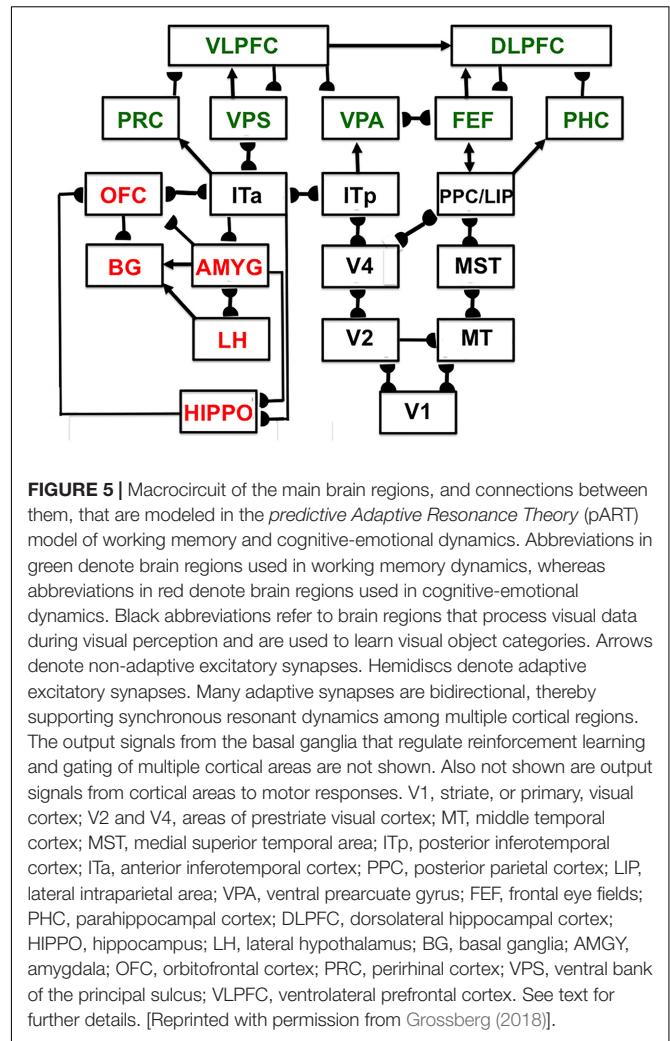
Since SOVEREIGN appeared, the predictive Adaptive Resonance Theory, or pART, model (Grossberg, 2018) has proposed how several parts of the prefrontal cortex (PFC) learn to interact with multiple brain regions to carry out cognitive and spatial working memory, planning, and cognitive-emotional processes. The seven prefrontal cortical regions marked in green in **Figure 5** illustrate this complexity. As one of its several explanatory accomplishments, pART clarifies how a top-down cognitive prime from the PFC can bias object attention in the What cortical stream to anticipate expected objects and events, while it also focuses spatial attention in the Where cortical stream to trigger actions that acquire currently valued objects (Fuster, 1973; Baldauf and Desimone, 2014; Bichot et al., 2015). Section 7 will summarize several of these enhanced capabilities of pART. As these enhanced capabilities of pART are incorporated into SOVEREIGN2, it will be able to carry out more sophisticated cognitive, cognitive-emotional, and Where's Waldo search capabilities than can the SOVEREIGN or the 3D ARTSCAN SEARCH models.

The pART model embodies several different kinds of brain resonances. In particular, the Fuzzy ART classifier in **Figure 4A** is an algorithmic realization of the kind of *feature-category resonance* that links cortical areas V4 and ITp in **Figure 5**. Such a resonance focuses attention upon salient combinations of features while it triggers learning in the bottom-up adaptive filters and top-down learned expectations that bind the attended feature patterns to the object categories that are used to recognize them. Adaptive Resonance Theory, or ART, explicates several different kinds of brain resonances and their different functional roles, as will be further discussed in sections 4.15 and 4.16.

4.13. From Incomplete Early Sensory Representations to Conscious Awareness and Effective Action

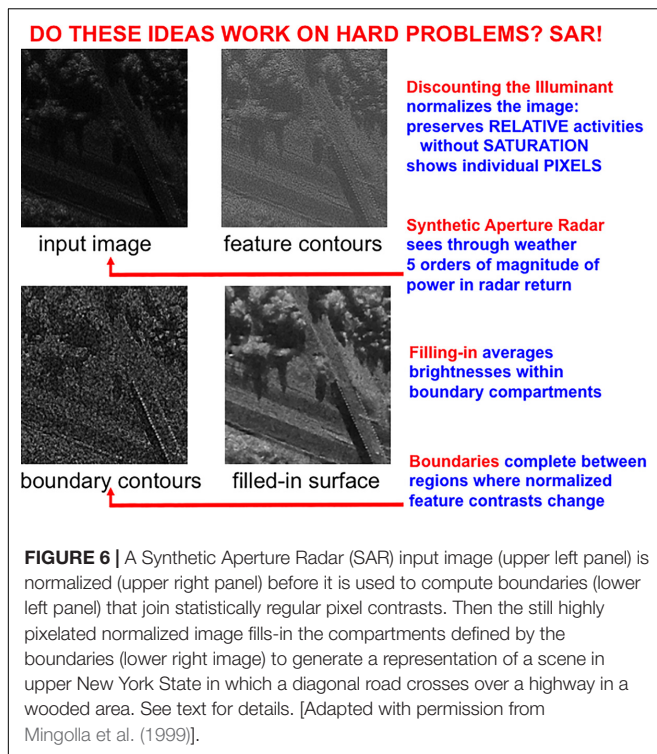
Hierarchical resolution of uncertainty occurs even at the earliest cortical processing levels. One of the most important consequences of hierarchical resolution of uncertainty arises from the fact that the perceptual representations that are computed at early processing stages may not be able to control effective actions. These processing stages did not have to be included in SOVEREIGN because it directly processed simplified virtual reality images (**Figure 2**). SOVEREIGN thus did not have to deal with problems that are raised when images are processed by noisy detectors that are made from biological or physical components.

For example, visual images that are registered on the retina of a human eye are noisy and incomplete due to the existence of the blind spot and retinal veins, which prevent visual features from being registered on the retina at their positions (**Supplementary Figure S4**). **Supplementary Figure S5** illustrates this problem



with the simple example of a line that is occluded by the blind spot and some retinal veins. The parts of the line that are occluded need to be completed at higher processing stages before actions like looking and reaching can be directed to these positions. Processes of boundary completion and surface filling-in are needed to generate a sufficiently complete, context-sensitive, and stable visual surface representation upon which subsequent actions can be based (Grossberg, 1994, 1997, 2013, 2017).

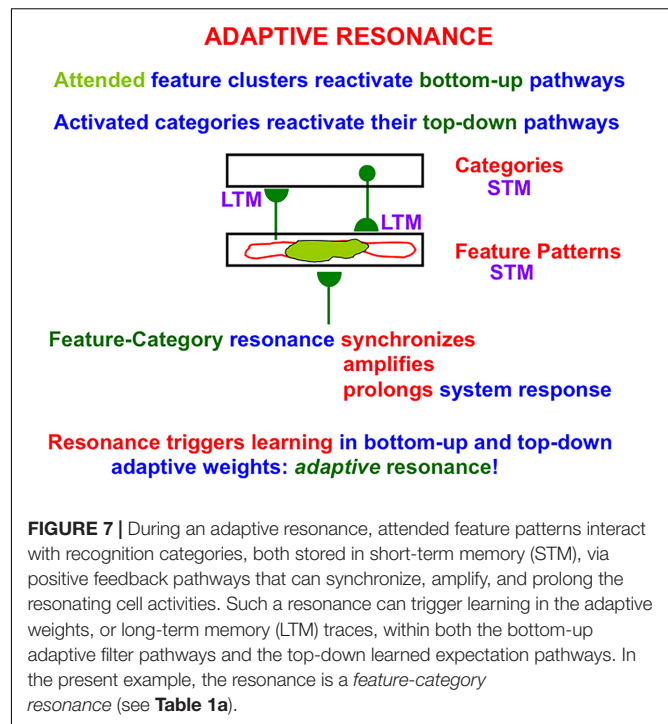
The front end of SOVEREIGN2 can be consistently extended to include these boundary completion and surface filling-in processes, instead of the Render 3-D Scene and Figure-Ground Separation processes in **Figure 3A**. SOVEREIGN2 can then function even using sensory detectors that may be pixelated or degraded in various ways due to use. Such detectors include artificial sensors such as Synthetic Aperture Radar, Laser Radar, and Multispectral Infrared. Synthetic Aperture Radar, or SAR, can be used to process images that can see through the weather. **Figure 6** shows a computer simulation of how a SAR image can be processed by boundary completion and surface filling-in processes that compensate for sensor failures.



Boundary completion and surface filling-in processes illustrate one of the best known examples of complementary computing (Grossberg, 1984a, 1994, 1997; Grossberg and Mingolla, 1985): Boundaries are completed *inwardly* between pairs or greater numbers of inducers in an *oriented* fashion. Boundary completion is also triggered after the processing stage where cortical complex cells pool signals from simple cells that are sensitive to opposite contrast polarities, thus becoming *insensitive* to direction of contrast. Because they pool over opposite contrast polarities—including achromatic black–white contrasts, and chromatic red–green and blue–yellow contrasts—boundaries cannot represent conscious visual qualia. That is, *all boundaries are invisible*. Surface filling-in of brightness and color spread *outwardly* in an *unoriented* fashion until they hit a boundary, or attenuate due to their spatial spread. Surface filling-in is also *sensitive* to direction of contrast. *All conscious percepts of visual qualia are surface percepts*. These three pairs of properties (inward vs. outward, oriented vs. unoriented, and insensitive vs. sensitive to direction of contrast) are manifestly complementary.

4.14. Why Did Evolution Discover Consciousness? Conscious States Control Adaptive Actions

The above review of some of the early processing stages in the visual system provides a foundation for understanding how ART provides a rigorous computational proposal both for *what* happens in each brain and *how* and *where* it happens as it learns to consciously see, hear, feel, or know something, as well as for *why* evolution was driven to discover conscious



states in the first place (Grossberg, 2017). In particular, as noted above, in order to resolve the computational uncertainties caused by complementary computing, the brain needs to use multiple processing stages that include interactions between pairs of complementary cortical processing streams to realize a *hierarchical resolution of uncertainty*.

Because the light that falls on our retinas may be occluded by the blind spot, multiple retinal veins, and all the other retinal layers through which light passes before it reaches the light-sensitive photoreceptors (**Supplementary Figures S4, S5**), these retinal images are highly noisy and incomplete. Using them to control actions like looking and reaching could lead to incorrect, and potentially disastrous, actions.

In order to compute the functional units of visual perception, namely 3D boundaries and surfaces, three pairs of computationally complementary uncertainties need to be resolved using a hierarchical resolution of uncertainty. If this is indeed the case, then why do not the earlier processing stages undermine behavior by causing incorrect, and potentially disastrous, actions to be taken? In the case of visual perception, the proposed answer is that *brain resonance, and with it conscious awareness of visual qualia, is triggered at the cortical processing stage that represents 3D surface representations, after they are complete, context-sensitive, and stable enough to control visually based actions like attentive looking and reaching*. The conscious state is an “extra degree of freedom” that selectively “lights up” this surface representation and enables our brains to selectively use it to control adaptive actions.

ART hereby links the evolution of consciousness to the ability of advanced brains to learn how to control adaptive actions. In the case of visual perception, this surface representation is

predicted to occur in prestriate visual cortical area V4, where a *surface-shroud resonance* that supports conscious seeing is predicted to be triggered between V4 and the posterior parietal cortex, or PPC (Figure 5), before it propagates both top-down to V2 and V1 and bottom-up to the PFC. The PPC is in the dorsal Where cortical stream. An *attentional shroud* is spatial attention that fits itself to the shape of an attended object surface (Tyler and Kontsevich, 1995). An active surface-shroud resonance maintains spatial attention on the surface throughout the duration of the resonance. When spatial attention shifts, the resonance collapses and another object can be attended.

While a surface-shroud resonance is still active, it regulates saccadic eye movement sequences that foveate salient features on the attended object surface. These properties mechanistically explain the distinction between two different functional roles of PPC: its control of top-down *attention* from PPC to V4 and its control of the *intention* to move, a distinction that has been reported in both psychophysical and neurophysiological experiments (e.g., Andersen et al., 1985; Gnadt and Andersen, 1988; Snyder et al., 1997, 1998). How spatial attention regulates the learning of invariant object categories during free scanning of a scene using its intentional choice of scanning eye movements that foveate sequences of salient surface features will be summarized in section 6.4.

The proposed link between consciousness and action is relevant to the design of future autonomous adaptive robots, and provides a new computational perspective for discussing whether machine consciousness is possible, and how it may be necessary to control a robot's choice of context-appropriate actions.

4.15. Synchronized Resonances for Seeing and Knowing: Visual Neglect and Agnosia

Many psychological and neurobiological data have been explained using ART resonances. For example, surface-shroud resonances for conscious seeing and feature-category resonances for conscious knowing of visual events can synchronize via shared visual representations in the prestriate cortical areas V2 and V4 when a person sees and knows about a familiar object (Figure 5). A lesion of the parietal cortex in one hemisphere can prevent a surface-shroud resonance from forming, thereby leading to the clinical syndrome of *visual neglect*, whereby an individual may draw only one half of the world, dress only one half of the body, and make erroneous reaches. A lesion of the inferotemporal cortex can prevent a feature-category resonance from forming, thereby leading to the clinical syndrome of *visual agnosia*, whereby a human can accurately reach for an object without knowing anything about it. See Grossberg (2017) for mechanistic explanations.

4.16. Classification of Adaptive Resonances for Seeing, Hearing, Feeling, Knowing, and Acting

In addition to the surface-shroud resonances that supports conscious seeing and the feature-category resonances that

support conscious knowing, ART explains what resonances support hearing and feeling, and how resonances supporting knowing are synchronously linked to them. All of these resonances support different kinds of learning that solve the stability-plasticity dilemma; e.g., visual and auditory learning, reinforcement learning, cognitive recognition learning, and cognitive speech and language learning.

In summary, *surface-shroud resonances* support conscious percepts of visual qualia. *Feature-category resonances* support conscious learning and recognition of visual objects and scenes. Both kinds of resonances may synchronize during conscious seeing and recognition, so that we know what a familiar object is as we see it. *Stream-shroud resonances* support conscious percepts of auditory qualia. *Spectral-pitch-and-timbre resonances* support conscious learning and recognition of sources in auditory streams. Stream-shroud and spectral-pitch-and-timbre resonances may synchronize during conscious hearing and recognition of auditory streams. *Item-list resonances* support conscious learning and recognition of speech and language. They may synchronize with stream-shroud and spectral-pitch-and-timbre resonances during conscious hearing of speech and language, and build upon the selection of auditory sources by spectral-pitch-and-timbre resonances in order to recognize the acoustical signals that are grouped together within these streams. *Cognitive-emotional resonances* support conscious percepts of feelings, as well as learning and recognition of the objects or events that cause these feelings. Cognitive-emotional resonances can synchronize with resonances that support conscious qualia and knowledge about them.

These resonances embody parametric properties of individual conscious experiences that enable effective actions to be chosen without interference from earlier processing stages. For example, surface-shroud resonances help to control looking and reaching; stream-shroud resonances help to control auditory communication, speech, and language; and cognitive-emotional resonances help to acquire valued goal objects. In autonomous adaptive systems that solve the stability-plasticity dilemma using ART dynamics, formal mechanistic homologs of such different states of resonant consciousness may be needed to choose the different kinds of actions that they control. More information will be summarized below about cognitive-emotional resonances in sections 6.5–6.7.

5. BUILDING UPON THREE BASIC DESIGN THEMES: HOMOLOGOUS CIRCUITS FOR REACHING AND NAVIGATING

A third design theme that is realized by the SOVEREIGN model is that advanced brains use homologous circuits to compute arm movements during reaching behaviors, and body movements during spatial navigation. In particular, both navigational movements and arm movements are controlled

by circuits which share a similar mismatch learning law—called a Vector Associative Map, or VAM (Gaudiano and Grossberg, 1991, 1992; see section 5.3)—that enables learned calibration of *difference vectors* in the manner described below. This proposed homology clarifies how navigational and arm movements can be coordinated when a body navigates toward a goal object before grasping it. SOVEREIGN used difference vectors to model navigational movements. It did not, however, include a controller for arm movements that could grasp a valued object when it came within range. The text below indicates how unsupervised incremental learning in SOVEREIGN2 realizes such a capability and can, moreover, do so using a tool (see section 5.4).

5.1. Arm Movement Control Using Difference Vectors and Volitional GO Signals

Neural models of arm movement trajectory control, such as the Vector Integration to Endpoint, or VITE, model (Bullock and Grossberg, 1988) (Figure 8, left panel) and their refinements (e.g., Bullock et al., 1993) (Figure 8, right panel) propose how cortical arm movement control circuits compute a representation of where the arm wants to move (i.e., the *target position vector* T) and subtracts from it an outflow representation of where the arm is now (i.e., the *present position vector* P). The resulting *difference vector* D between target position T and present position P represents the direction and distance that the arm needs to move to reach its goal position. Basal ganglia (BG) volitional signals of various kinds, such as a GO signal G , transform the difference vector D into a motor trajectory that can move with variable speed by multiplying D with G , before this product is integrated by P . Because P integrates the product DG , DG represents the commanded outflow movement speed. Then P moves at a speed that increases with G , other things being equal. As P approaches T , D approaches zero, along with the outflow speed DG , so the movement terminates at the desired target position.

5.2. Computing Present Position for Spatial Navigation From Vestibular Signals: Place Cells

Because the arm is attached to the body, the present position of the arm can be computed using outflow, or corollary discharge, commands P that are derived directly from the movement commands to the arm itself (Figure 8, left panel). In contrast, when a body moves with respect to the world, no such immediately available present position command is available. The ability to compute a difference vector between a target position and the present position of the body—in order to determine the direction and distance that the body needs to navigate to acquire the target—requires more elaborate brain machinery. At the time SOVEREIGN was published, computation of such a Present Position Vector, called NET in SOVEREIGN, used an algorithm to estimate the information that vestibular signals compute *in vivo*.

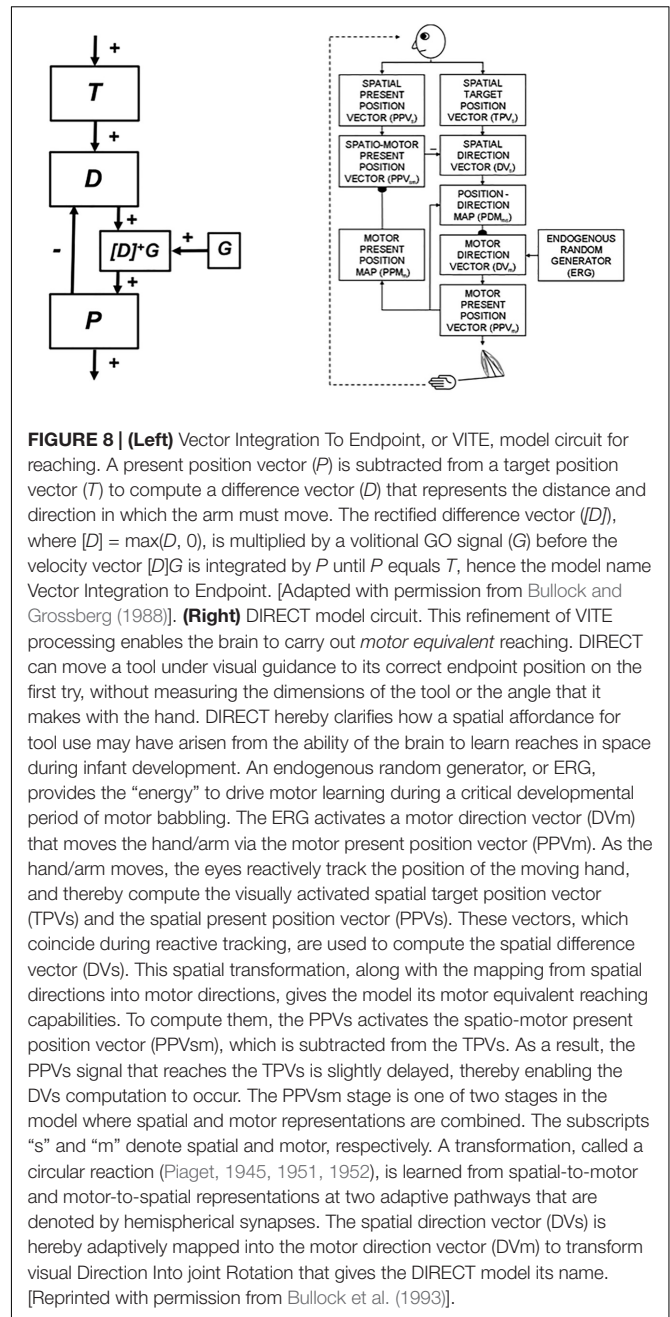


FIGURE 8 | (Left) Vector Integration To Endpoint, or VITE, model circuit for reaching. A present position vector (P) is subtracted from a target position vector (T) to compute a difference vector (D) that represents the distance and direction in which the arm must move. The rectified difference vector ($[D]$), where $[D] = \max(D, 0)$, is multiplied by a volitional GO signal (G) before the velocity vector $[D]G$ is integrated by P until P equals T , hence the model name Vector Integration to Endpoint. [Adapted with permission from Bullock and Grossberg (1988)]. **(Right)** DIRECT model circuit. This refinement of VITE processing enables the brain to carry out *motor equivalent* reaching. DIRECT can move a tool under visual guidance to its correct endpoint position on the first try, without measuring the dimensions of the tool or the angle that it makes with the hand. DIRECT hereby clarifies how a spatial affordance for tool use may have arisen from the ability of the brain to learn reaches in space during infant development. An endogenous random generator, or ERG, provides the “energy” to drive motor learning during a critical developmental period of motor babbling. The ERG activates a motor direction vector (DV_m) that moves the hand/arm via the motor present position vector (PPV_m). As the hand/arm moves, the eyes reactively track the position of the moving hand, and thereby compute the visually activated spatial target position vector (TPVs) and the spatial present position vector (PPVs). These vectors, which coincide during reactive tracking, are used to compute the spatial difference vector (DVs). This spatial transformation, along with the mapping from spatial directions into motor directions, gives the model its motor equivalent reaching capabilities. To compute them, the PPVs activates the spatio-motor present position vector (PPV_{sm}), which is subtracted from the TPVs. As a result, the PPVs signal that reaches the TPVs is slightly delayed, thereby enabling the DVs computation to occur. The PPV_{sm} stage is one of two stages in the model where spatial and motor representations are combined. The subscripts “s” and “m” denote spatial and motor, respectively. A transformation, called a circular reaction (Piaget, 1945, 1951, 1952), is learned from spatial-to-motor and motor-to-spatial representations at two adaptive pathways that are denoted by hemispherical synapses. The spatial direction vector (DVs) is hereby adaptively mapped into the motor direction vector (DV_m) to transform visual Direction Into joint Rotation that gives the DIRECT model its name. [Reprinted with permission from Bullock et al. (1993)].

SOVEREIGN breaks down spatial navigation into sequences of straight excursions in fixed directions, after which a head/body turn changes the direction before another straight excursion occurs. *In vivo*, vestibular signals provide angular velocity and linear velocity signals that can be integrated to compute these head/body angles and straight movement distances. The SOVEREIGN algorithm adds the head/body turn angles, as well as the body approach distances for each straight excursion, to compute NET. Then, as Figure 1B summarizes, NET is subtracted from the Reactive Visual TPV Storage to compute a Reactive DV, which controls the next straight movement in space. Each head/body turn resets NET

to allow the next NET estimate to be computed. Using such computations, SOVEREIGN was able to learn how to navigate toward valued goals in structured environments like the maze in **Figure 2**.

In sufficiently advanced terrestrial animals, from rats to humans, an animal's position in space is computed from a combination of both visual and *path integration* information. The visual information is derived from 3D perceptual representations that are completed by processes such as boundary completion and surface filling-in. The path integration information is derived from vestibular angular velocity and linear velocity signals that are activated by an animal's navigational movements. This vestibular information is transformed by entorhinal grid cells and hippocampal place cells into representations of the animal's present position in space (O'Keefe and Nadel, 1978; Hafting et al., 2005). The GridPlaceMap model simulated how these cells learn their spatial representations as the animal navigates realistic trajectories (e.g., Grossberg and Pilly, 2014). Key properties of the GridPlaceMap model and some of the grid cell and place cell data that it can explain are summarized in section 8.

When SOVEREIGN2 replaces the algorithmic computations of NET in **Figure 1B** by circuits that learn grid and place cells, it can then autonomously learn spatial NET estimates as the animat navigates novel environments that may be far more complicated than the plus maze in **Figure 2**. When such a self-organized NET estimate is used to compute a difference vector between the present and target positions, a volitional GO signal can move the animat toward the desired target, just as in the case of an arm movement.

5.3. From VITE to VAM: How a Circular Reaction Drives Mismatch Learning to Calibrate VITE

In order for VITE dynamics to work properly, its difference vectors need to be properly calibrated. In particular, when T and P represent the same position in space, D must equal zero. However, T and P are computed in two different networks of cells. It is too much to expect that the activities of these two networks, and the gains of the pathways that carry their signals to D , become perfectly matched without the benefit of some kind of learning. The Vector Associative Map, or VAM model explains how this kind of learning occurs (Gaudiano and Grossberg, 1991, 1992). In brief, the VAM model corrects this problem using a form of mismatch learning that adaptively changes the gains in the T -to- D pathways until they match those in the P -to- D pathways, so that when $T = P$, $D = 0$.

The VAM model does this using what has been called a *circular reaction* since the pioneering work of Jean Piaget on infant development (Piaget, 1945, 1951, 1952). All infants normally go through a *babbling phase*, and it is during such a babbling phase that a circular reaction can be learned. In particular, during a visual circular reaction, babies endogenously babble, or spontaneously generate, hand/arm movements to multiple positions around their bodies. As their hands move in front of them, their eyes automatically, or reactively, look

at their moving hands. While the baby's eyes are looking at its moving hands, the baby learns an associative map from its hand positions to the corresponding eye positions, *and* from eye positions to hand positions. Learning of the map between eye and hand in both directions constitutes the "circular" reaction.

After map learning occurs, when a baby, child, or adult looks at a target position with its eyes, this eye position can use the learned associative map to generate a movement command to reach the corresponding position in space. In order for the command to be read out, a volitional GO signal from the BG—notably from the substantia nigra pars reticulata, or SNr—opens the corresponding movement gate (Prescott, 2008). Such a gate-opening signal realizes "the will to act." Then the hand/arm can reach to the foveated position in space under volitional control. Because our bodies continue to grow for many years as we develop from babies into children, teenagers, and adults, these maps continue updating themselves throughout our lives.

In a VAM, endogenous babbling is accomplished by an Endogenous Random Generator, or ERG+, that sends random signals to P that cause the arm to automatically babble a movement in its workspace. This movement is thus not under volitional control. When P gets activated, in addition to causing the arm to move, it sends signals that input an inhibitory copy of itself to D .

The ERG has an opponent organization. It is the ERG ON, or ERG+, component that energizes the babbled arm moment. When ERG+ momentarily shuts off, ERG OFF, or ERG−, is disinhibited and opens a gate that lets P get copied at T , where it is stored. At this moment, both T and P represent the same position in space. If the model were correctly calibrated, the excitatory T -to- D and inhibitory P -to- D signals that input to D in response to the same positions at T and P would cancel, causing D to equal zero. If D is not zero under these circumstances, then the signals are not properly calibrated. The VAM model uses such non-zero D vectors as *mismatch teaching signals* that adaptively calibrate the T -to- D signals. As perfect calibration is achieved, D approaches zero, at which time mismatch learning self-terminates.

Another refinement of VITE showed how arm movements can compensate for variable loads and obstacles, and interpreted the hand/arm trajectory formation stages in terms of identified cells in motor and parietal cortex, whose temporal dynamics during reaching behaviors were quantitatively simulated (Bullock et al., 1998; Cisek et al., 1998).

5.4. Motor-Equivalent Reaching With Clamped Joints and Tools: The DIRECT Model

Yet another VITE model refinement, called the DIRECT model (**Figure 8**, right panel), builds upon VAM calibration to propose how *motor-equivalent reaching* is learned (Bullock et al., 1993). Motor-equivalent reaching explains how, during movement planning, either arm, or even the nose, could be moved to a target position, depending on which movement system receives a GO signal.

The DIRECT model also begins to learn by using a circular reaction that is energized by an ERG (Figure 8, right panel). Motor-equivalent reaching emphasizes that reaching is not just a matter of combining visual and motor information to transform a target position on the retina into a target position in body coordinates. Instead, these visual and motor signals are first combined to learn a representation of the *space* around the actor which can then be downloaded to move any of several motor effectors.

Remarkably, after the DIRECT model uses its circular reaction to learn its spatial representations and transformations, its motor-equivalence properties enable it to accurately move an arm, even when its joints are clamped, to any position in its workspace on the first try. DIRECT can also *manipulate a tool in space*. The conceptual importance of this result cannot be overemphasized: Without measuring tool length or angle with respect to the hand, the model can move the tool's endpoint to touch the target's position correctly under visual guidance *on its first try*, in a single reach without later corrective movements, and without additional learning. In other words, the *spatial affordance* for tool use, a critical foundation of human societies, follows from the brain's ability to learn a circular reaction for motor-equivalent reaching in space. Adding these reaching capabilities to SOVEREIGN2 will enable it to use tools to manipulate target objects after it navigates to them.

5.5. Social Cognition: Joint Attention and Imitation Learning Using CRIB Circular Reactions

The DIRECT model shows how the spatial affordance for tool use could arise as a result of the circular reactions that enable reaching behaviors to develop. With DIRECT on board, a child, monkey, or robot could then volitionally reach objects with its own hand, or even using a tool like a stick. If a monkey happened to pick up a stick in this way, put it into an ant hill, and pulled it out with some ants on it, it could learn this skill to eat ants in the future whenever it wanted to do so. However, another monkey looking at this skill could not learn it from the first one without further brain machinery, because the two monkeys experience this event from two different spatial vantage points. This additional brain machinery is needed for social cognitive skills to be learned, including the learning of joint attention and imitation learning. These are competences upon which all human societies have built.

Grossberg and Vladusich (2010) develops the Circular Reactions for Imitative Behavior, or CRIB, model to explain how imitation learning utilizes *inter*-personal circular reactions that take place between teacher and learner, notably how a learner can follow a teacher's gaze to fixate a valued goal object, and distinguishes them from the classical *intra*-personal circular reactions of Piaget that take place within a single learner, such as the one that enables reaching behaviors to be learned. After a learner can volitionally reach objects on its own, it can also learn, using an *inter*-personal circular reaction, to reach an object at which a teacher is looking,

such as a stick with which to retrieve ants from an anthill. By building upon *intra*-personal circular reactions that are capable of learning motor-equivalent reaches, the CRIB model hereby clarifies how a pupil can learn from a teacher to manipulate a tool in space.

In order to achieve joint attention and imitation learning, the learner needs to be able to bridge the gap between the teacher's coordinates and its own. In the neurobiological literature, this capability is often attributed to *mirror neurons* that fire either if an individual is carrying out an action or just watching someone else perform the same action (Rizzolatti and Craighero, 2004; Rizzolatti, 2005). This attribution does not, however, mechanistically explain how the properties of mirror neurons arise. The CRIB model proposes that the "glue" that binds these two coordinate systems, or perspectives, together is a surface-shroud resonance. How this works is modeled in Grossberg and Vladusich (2010). It is also known that a breakdown of joint attention can cause severe social difficulties in individuals with autism. How these and other breakdowns in learning cause symptoms of autism are modeled by the iSTART model (Grossberg and Seidman, 2006).

If CRIB-like social cognition capabilities are incorporated into a "classroom" of SOVEREIGN2 robots, they can then all learn sensory-motor skills from a teacher who they see from different vantage points.

5.6. Platform Independent Movement Control

If SOVEREIGN2 is used to control an embodied mobile robot, then an important design choice is whether to use legs or wheels with which to navigate. Difference vector (DV) control of direction and distance that is gated by a GO signal can be used in either case.

To help guide the development of a legged robot, neural network models have shown how leg movements can be performed with different gaits, such as walk or run in bipeds, and walk, trot, pace, and gallop in quadrupeds, as the GO signal size increases (Pribe et al., 1997).

An example of DV-GO control in a wheeled mobile robot was developed by Zalama et al. (1995) and Chang and Gaudiano (1998) and tested on robots such as the Khepera and Pioneer 1 mobile robots to demonstrate VAM learning of how to approach rewards and avoid obstacles in a cluttered environment, with no prior knowledge of the geometry of the robot or of the quality, number, or configuration of the robot's sensors. Learning in one environment generalized to other environments because it is based on the robot's egocentric frame of reference. The robot also adapted on line to miscalibrations produced by wheel slippage, changes in wheel sizes, and changes in the distance between the wheels.

In summary, both navigational movements in the world and movements of limbs with respect to the body use a difference vector computational strategy.

Sections 6–8 provide a deeper and broader conceptual and mechanistic insight into the themes that the earlier sections have introduced.

6. RESONANT DYNAMICS FOR PERCEPTION, COGNITION, AFFECT, AND PLANNING

6.1. Invariant Object Category Learning Uses Feature-Category Resonances and Surface-Shroud Resonances

Many of the enhanced capabilities of SOVEREIGN2 will use resonant processes. In particular, in order for SOVEREIGN2 to learn view-, position-, and size-invariant object categories as it scans a scene with eye, or camera, movements, two different types of resonances need to be coordinated: feature-category resonances and surface-shroud resonances. *In vivo*, view-, position-, and size-specific visual percepts in the striate and prestriate visual cortices V1, V2, and V4 are transformed into view-, position-, and size-specific object recognition categories in the posterior inferotemporal cortex (ITp) via *feature-category resonances* (Table 1a and Figure 7) within the What cortical stream.

Within SOVEREIGN, the specific categories in ITp were learned using the unsupervised Fuzzy ART model (Figure 4A). Fuzzy ART can also be used for this purpose in SOVEREIGN2, with visual inputs now coming from 3D boundary and surface representations. Recognition learning may be supervised by replacing Fuzzy ART with Fuzzy ARTMAP (Carpenter et al., 1992) or any similar dynamical or algorithmic supervised version of ART. As will be summarized below, however, truly autonomous *invariant* object category learning that avoids the algorithmic tricks of SOVEREIGN will require more sophisticated network interactions.

Despite its simplicity, Fuzzy ART is an algorithmic realization of dynamical properties of ART that embody both a feature-category resonance (Figure 6) and a classical example of complementary computing. Complementary computing enables feature-category resonances to continuously learn to recognize novel objects using interactions between an *attentional system* in which category learning occurs, and an *orienting system* that drives memory searches and hypothesis testing for novel categories in response to large enough mismatches between bottom-up and top-down input patterns (Figure 9) (Grossberg, 1976b, 1980, 2017).

6.2. Complementary Computing: ART Hypothesis Testing and Learning of Predictive Categories

The need for an orienting system can be seen by answering the question: If learning can occur only if there is a sufficiently good match between bottom-up input patterns and top-down expectations, then how is anything truly novel ever learned? Here is where complementary properties of attentional matching and orienting search are crucial: A sufficiently bad mismatch between an active top-down expectation and a bottom-up input, say because the input is unfamiliar, can drive a memory search and hypothesis testing. Such a mismatch within the attentional system activates the complementary orienting system, which

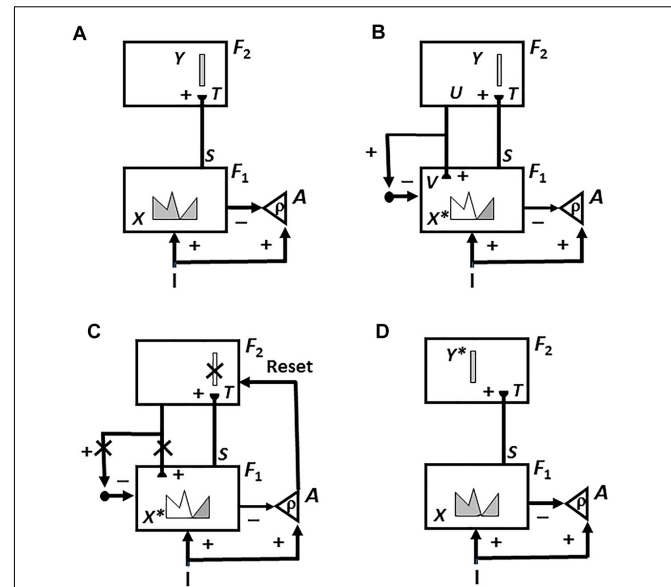


FIGURE 9 | ART cycle of match-induced resonant learning and mismatch-induced reset and search. **(A)** The input pattern I is instated across feature detectors at level F_1 as an activity pattern X , as it also inputs to the orienting system A with a gain ρ called *vigilance*. Activity pattern X sends inhibitory signals to A and a bottom-up excitatory input pattern S to the category level F_2 . Balanced excitatory inputs from I and inhibitory inputs from X keeps A quiet. S inputs are multiplied by learned adaptive weights to define the input pattern T to F_2 . Inputs T are contrast-enhanced and normalized within F_2 by recurrent lateral inhibitory signals that obey the membrane equations of neurophysiology, also called shunting interactions. A small number of cells within F_2 that receive the largest inputs are chosen by this competition. These cells represent the category Y that codes the feature pattern at F_1 . A winner-take-all category is shown. **(B)** Category Y generates top-down signals U that are multiplied by adaptive weights to form a prototype, or critical feature pattern, V . V represents the expectation that Y has learned of the feature pattern to expect at F_1 . If V mismatches I at F_1 , then a new STM activity pattern X^* (the hatched pattern), is chosen at cells where the patterns match well enough; that is, X^* is active at I features that are confirmed by V . Mismatched features (white area) are inhibited. When X changes to X^* , total inhibition decreases from F_1 to A . **(C)** If inhibition decreases sufficiently, A triggers non-specific arousal to F_2 , thereby instantiating that “novel events are arousing.” Vigilance ρ determines how bad a match will be tolerated before non-specific arousal is triggered. Arousal initiates a memory search for a better-matching category in the following way: First, arousal resets F_2 by inhibiting Y . **(D)** After Y is inhibited, X is reinstated and Y stays inhibited as X activates a different category Y^* at F_2 . Search continues until a better matching, or novel, category is selected. When search ends, a resonance develops that supports learning of the attended data in the adaptive weights within both the bottom-up and top-down pathways. After learning, inputs I can activate the globally best-matching categories directly through the adaptive filter without activating the orienting system. [Adapted with permission from Carpenter and Grossberg (1993)].

is sensitive to unexpected and unfamiliar events. The ART attentional system includes the inferotemporal and prefrontal cortices, whereas the orienting system includes the non-specific thalamus and hippocampal system. See Carpenter and Grossberg (1993) and Grossberg and Versace (2008) for supportive neurobiological data.

The fact that ART learns only if a sufficiently good match occurs also imposes constraints upon how top-down adaptive

weights are initially chosen to enable category learning to get started: In any ART system, the top-down adaptive weights that represent learned expectations need to be broadly distributed and large before learning occurs, so that they can match whatever input pattern first initiates learning of a new category. Indeed, when a new category is first activated, it is not known at the category level what pattern of features caused the category to be activated. *Whatever* feature pattern was active needs to be matched by the top-down expectation on the first learning trial, so that resonance and weight learning can begin. Hence the need for the initial values of top-down weights to be broadly distributed and sufficiently large to match *any* feature pattern.

Given that top-down weights are initially broadly distributed, the learning of top-down expectations is a process of *pruning* weights on subsequent learning trials, and uses mismatch-based reset events to discover categories capable of representing the environment. The large initial adaptive weights in top-down expectations helps to explain otherwise mysterious neurobiological data, such as why there is an Inverted-U through time in the power of beta oscillations when an animal first navigates a new maze (Berke et al., 2008; Grossberg, 2009a).

6.3. Complementary PN and N200 Event Related Potentials During Attention and Memory Search

In contrast to the *top-down*, *adaptive*, *specific*, and *match* properties that occur during an attentive match, an orienting system mismatch is *bottom-up*, *non-adaptive*, *non-specific*, and *mismatch*: A mismatch occurs when *bottom-up* activation of the orienting system cannot be adequately inhibited by the bottom-up inhibition from the matched pattern (**Figure 9B**). The signals to and from the orienting system are *non-adaptive*, or not subject to learning. Mismatch-activated output from the orienting system *non-specifically* arouses all the category cells because the orienting system cannot determine which categories read out the expectation that led to mismatch (**Figure 9C**). *Any* category may be responsible, and may thus need to be reset by arousal (**Figure 9D**). Finally, the orienting system is activated by a sufficiently big *mismatch*.

These are properties of the N200 event-related potential, or ERP (Näätänen et al., 1982; Sams et al., 1985). More generally, during an ART memory search, sequences of the predicted mismatch, arousal, and reset events occur that exhibit properties of the sequentially occurring P120, N200, and P300 ERPs, respectively (Banquet and Grossberg, 1987).

In summary, four sets of properties of the attentional system are complementary to those of the orienting system (top-down vs. bottom-up, adaptive vs. non-adaptive, specific vs. non-specific, match vs. mismatch), with the PN and N200 ERPs illustrating these complementary properties. The orienting system can detect that an error has occurred, but does not know what category prediction caused it. The attentional system knows what categories are active, but not if these categories adequately represent current inputs. By interacting, these systems can determine what the error is and discover and learn a new category to correct it. Complementary computing hereby accomplishes

incremental learning and autonomous error correction of a large non-stationary database, without incurring the risk of catastrophic forgetting.

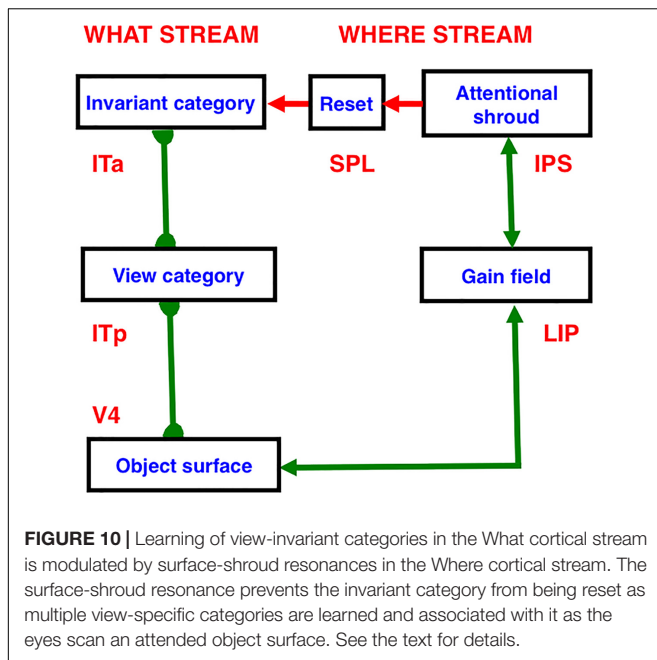
6.4. Autonomous Solution of the Invariant Pattern Recognition Problem During Active Vision

In our brains, as ITp categories are learned using feature-category resonances, they create the substrate for learning view-, position-, and size-invariant object recognition categories within the ventral What cortical processing stream, notably in the anterior inferotemporal cortex, or ITa. The 3D ARTSCAN Search model has been incrementally developed to explain in detail how our brains learn to solve the invariant pattern recognition problem during active vision, a problem that is just as important for human survival as it is for designing machine learning algorithms that can autonomously learn in the real world (Fazl et al., 2009; Cao et al., 2011; Grossberg et al., 2011, 2014; Foley et al., 2012; Chang et al., 2014; Grossberg, 2017). When it is implemented in SOVEREIGN2, the 3D ARTSCAN Search architecture can be used to provide previously unavailable machine learning, recognition, and prediction abilities in autonomous adaptive mobile systems, notably self-training robots.

To carry out effective invariant category learning, the model needed to solve a basic View-to-Object Binding Problem, which concerns how our brains automatically know, without external supervision or prior learning, which views of a novel scene belong to the same object—and thus can be associated with the same invariant category—and which do not—so should not be associated. As a result, the model can learn invariant object categories in response to arbitrary combinations of unsupervised and supervised learning trials as the eyes freely scan a complex scene.

As ITp categories are learned using feature-category resonances (**Figure 7**), they are associated with cells in the anterior inferotemporal (ITa) cortex that learn to become view-, position-, and size-invariant object recognition categories. **Figure 10** illustrates how the View-to-Object Binding Problem is solved during invariant object category learning in ITa within the What cortical stream, with the help of modulation by the PPC in the Where cortical stream, including the inferior parietal sulcus (IPS), the lateral intraparietal area (LIP), and the medial superior parietal lobule (SPL). Surface-shroud resonances that are triggered between V4 and IPS play a critical role in modulating this invariant category learning modulation process, while they also support conscious visibility of the attended object surface.

An active surface-shroud resonance embodies the brain state that maintains spatial attention upon the object that is being learned about. While the object is attended, its shroud also inhibits category reset cells in SPL (**Figure 10**). While the surface-shroud resonance maintains attention on an object surface, it also regulates eye movements that successively foveate the most salient features on the attended surface (not shown in **Figure 10**), but not other objects in the scene, thereby solving the View-to-Object Binding Problem. Each foveation can lead to the learning of a different specific ITp category. The first such ITp category to



be learned chooses cells in ITa with which it will be associated via typical ART dynamics (Figure 10). As successive ITp categories are learned, they can all be associated with the same ITa cells because they cannot be inhibited by SPL. These ITa calls hereby learn to become an invariant object category by being associated with multiple specific ITp categories.

When spatial attention shifts from the object, its shroud collapses, thereby disinhibiting the reset cells in SPL. A transient burst of inhibition from these SPL cells resets the active invariant object category in ITa (Chiu and Yantis, 2009; Fazl et al., 2009). As the invariant object category collapses and the eyes attend another object's surface, new specific ITp and invariant ITa object categories can be learned to represent other objects in a scene. The cycle can then repeat itself. The model can hereby autonomously learn invariant object categories in response to arbitrary combinations of unsupervised and supervised learning trials as its eyes or cameras are directed to scan a complex scene.

After invariant categories are learned, the system can also solve the Where's Waldo Problem; that is, it can search a scene for a desired goal object within it. Such a search requires What-to-Where stream interactions.

6.5. Conditioned Reinforcer and Motivational Learning Use Cognitive-Emotional Resonance

Invariant object categories in ITa (sensory cortex in Figure 11A) learn to activate value categories via conditioned reinforcer pathways, whereas value categories learn to activate object-value categories in the orbitofrontal cortex (OFC) via incentive motivational pathways. Both kinds of learning occur during a *cognitive-emotional resonance* that is triggered when a conditioned stimulus, such as a buzzer sound, activates its invariant object category while an unconditioned stimulus, or

primary reward such as presentation of food to a hungry animal, activates its value category.

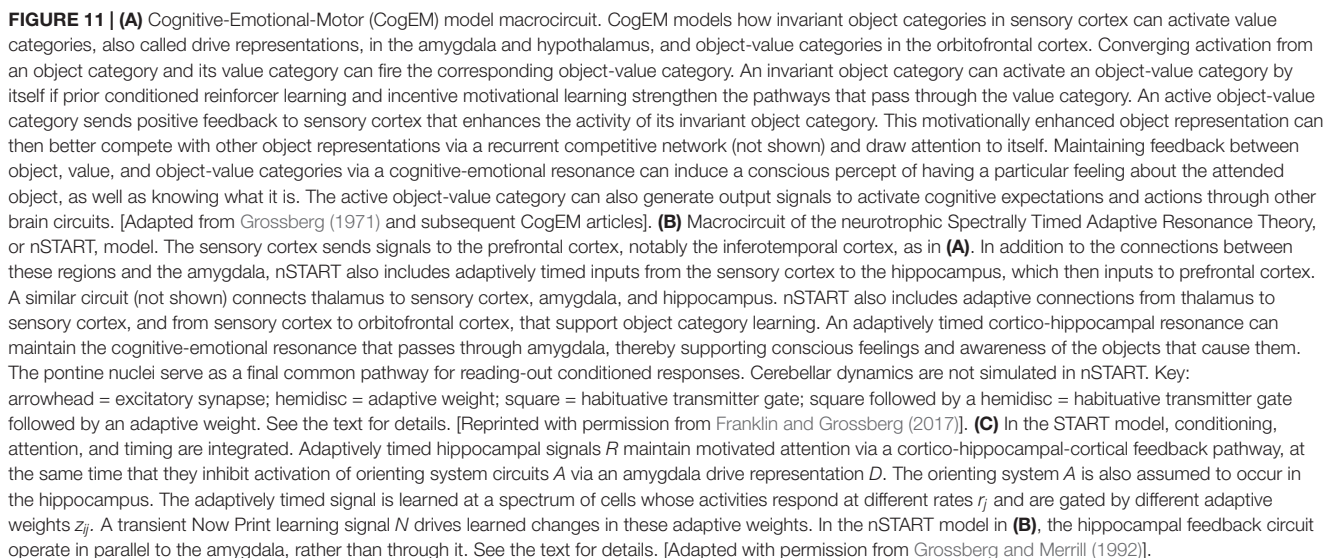
A cognitive-emotional resonance begins when object-value categories fire in response to converging inputs from sensory cortex and a value category. Then top-down feedback from the object-value category to its invariant object category closes a feedback loop between sensory cortex, amygdala (AMYG), and OFC that supports the cognitive-emotional resonance. This kind of resonance focuses motivated attention upon valued objects, while triggering context-appropriate actions toward them.

The model in Figure 11A that accomplishes conditioned reinforcer learning, incentive motivational learning, and release of motor actions toward valued goal objects is called the Cognitive-Emotional-Motor, or CogEM, model. CogEM has been getting incrementally developed since it was introduced in 1971 (e.g., Grossberg, 1971, 1982, 1984b; Grossberg and Gutowski, 1987; Dranias et al., 2008). The drive representations of the CogEM model include opponent processing channels called *gated dipoles* (Grossberg, 1972a,b, 1984b) that organize affective processing into opponent channels such as fear vs. relief, and hunger vs. frustration, which help to regulate behaviors like approach vs. avoidance, and exploration vs. consummation (cf. exploration vs. exploitation). Each gated dipole controls the balance between one pair of opponent affective representations. Variations of the gated dipole design occur in multiple brain processes, including the representation of opponent colors such as red vs. green, opponent directions such as up vs. down, and opponent muscles such as agonists vs. antagonists. Gated dipoles are thus a general design that helps to *reset* brain dynamics in response to sudden changes in environmental contingencies, and to restore brain dynamics to an unbiased state.

6.6. Antagonistic Rebounds Enable Opponent Extinction and Learning From Disconfirmations

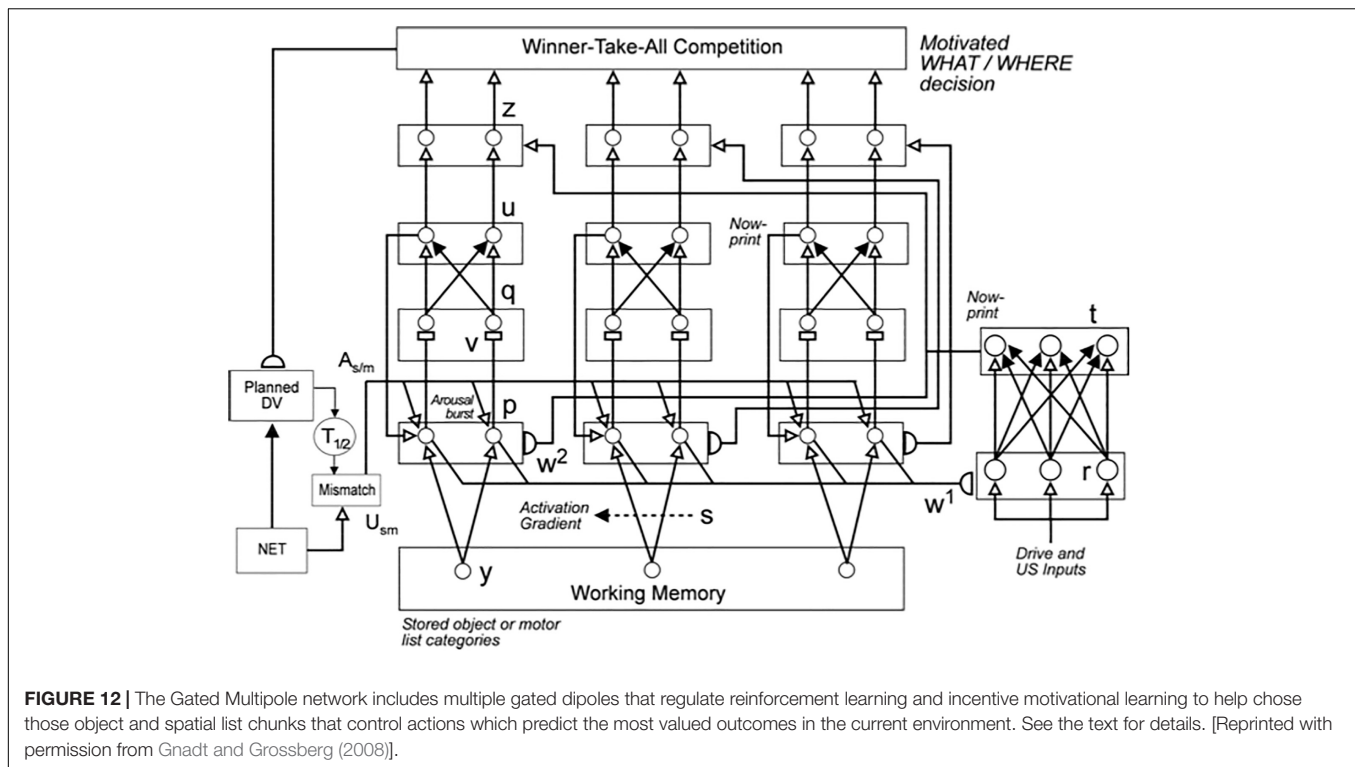
Gated dipole reset takes the form of an *antagonistic rebound* during which activation in its ON channel is replaced by a transient activation, or rebound, in its OFF channel. An antagonistic rebound can be triggered in response to a sudden decrease in the phasic input that was activating the ON channel, or to an unexpected event that causes a sudden increase in the arousal that activates both the ON and OFF channels (Grossberg, 1984b; Grossberg and Schmajuk, 1987). In this way, changing environmental contingencies, including the disconfirmation of expected events, can have reinforcing properties that can modulate which learned plans will be chosen to triggered goal-oriented actions in a particular environmental context.

When adaptive weights learn from both ON channel activations and OFF channel rebounds in response to disconfirmations of previous learning, then approximately equal learned inputs to both the ON and OFF channels can occur and lead to competitive suppression of output signals. The emotional and motivational support for such behaviors is then eliminated; the behavior has been *extinguished*. Recurrent gated



SOVEREIGN models an array of gated dipoles, called gated multipoles (**Figures 4, 12**), in which multiple opponent affective states compete with each other to decide which one of them has the momentarily best combination of sensory and motivational inputs to control behavioral choices as environmental conditions change. Gated multiples within CogEM circuits will also occur in SOVEREIGN2.

Learning often requires that learned associations form between sensory cues and reinforcers that are separated in time, with the sensory cues shutting off hundreds of milliseconds or even seconds before the reinforcer turns on. The CogEM model cannot learn in such situations because the AMYG cannot bridge temporal gaps of such a long duration. *In vivo*, the hippocampus (HIPPO) enables conditioning to bridge temporal gaps using a type of adaptively timed learning (**Figure 11B**) that is called *spectral timing* (Grossberg and Schmajuk, 1989;



Grossberg and Merrill, 1992, 1996). Spectrally timed learning can bridge time intervals of hundreds of milliseconds between the offset of a conditioned stimulus (CS) and the onset of a rewarding unconditioned stimulus (US), as occurs during reinforcement learning paradigms like trace conditioning and delayed-nonmatch to sample. It does so using populations of cells that each respond at different times (the “spectrum”), but for much shorter time intervals than the population response as a whole can span.

How do neurons, which typically fire on a millisecond time scale, span hundreds of milliseconds? Fiala et al. (1996) developed a detailed spectral timing model of cerebellar adaptive timing that links biochemistry, neurophysiology, neuroanatomy, and behavior, and predicts how the metabotropic glutamate (mGluR) receptor system may create a spectrum of delays during cerebellar adaptively timed learning. mGluRs are a form of glutamate receptor that is different from the ionotropic glutamate receptors that support widespread excitatory signaling throughout the brain. Unlike ionotropic glutamate receptors, which directly activate ion channels, mGluR receptors activate biochemical cascades. Spectral timing properties are predicted to be an example of such a biochemical cascade, with intracellular calcium regulating the different response rates of the cells within such a spectrum. This prediction has been supported by several subsequent experiments (e.g., Finch and Augustine, 1998; Takechi et al., 1998; Ichise et al., 2000; Miyata et al., 2000).

In addition to the mGluR spectral timing circuits that have modeled adaptively timed actions using the cerebellum (Fiala et al., 1996), similar mGluR circuits have modeled maintenance of adaptively timed incentive motivation that supports such actions using the HIPPO (Grossberg and Schmajuk, 1989; Grossberg

and Merrill, 1992, 1996), and adaptively timed reinforcement learning in response to unexpected rewards and punishments using the BG (Brown et al., 1999, 2004). Indeed, variants of spectral timing seem to be an ancient evolutionary discovery that includes non-neural systems. Simpler versions of such calcium-modulated spectra also occur in non-neural tissues such as HeLa cancer cells (Bootman and Berridge, 1996), the *puffs* in *Xenopus* oocytes (Yao et al., 1995), and the *sparks* in cardiac myocytes (Cannell et al., 1995; López-López et al., 1995).

In particular, the Spectrally Timed Adaptive Resonance Theory, or START, model has explained and simulated how spectrally timed learning may occur in dentate-hippocampal circuits (Figure 11C) (Grossberg and Schmajuk, 1989; Grossberg and Merrill, 1992, 1996). Data about both normal and abnormal learned timing have been explained by this model, including explanations of timing failures in individuals with autism and Fragile X syndrome (Grossberg and Seidman, 2006; Grossberg and Kishnan, 2018).

The neurotrophic START, or nSTART, model (Figure 11B) developed hippocampal spectral timing properties a different direction by proposing how spectral timing supports memory consolidation of previously learned associations using a combination of endogenous hippocampal bursting and modulation by brain-derived neurotrophic factor, or BDNF, during the consolidation period, which often occurs during periods of sleep (Franklin and Grossberg, 2017). If the HIPPO is ablated shortly after learning, then memory consolidation cannot take place, and medial temporal amnesia can be caused. More generally, the nSTART model explains and simulates why lesions of thalamus, AMYG, HIPPO, and OFC have different effects

on memory consolidation, depending on the phase of learning when they occur.

Both START and nSTART explain how a *cortico-hippocampal resonance* sustains cognitive-emotional resonances using its adaptively timed learning long enough for brains to become conscious of feelings and the events that caused them. The pART model circuit in **Figure 5** includes spectrally timed interactions between anterior inferotemporal cortex (ITa), HIPPO, and OFC, which then closes the adaptively timed feedback loop with ITa.

Adaptively timed behaviors are essential for success in an autonomous adaptive mobile system, including learning to properly time goal-oriented actions and to maintain motivated attention upon desired goal objects long enough to do so. A model HIPPO and cerebellum can be joined to the CogEM multipole model to enable SOVEREIGN2 to learn and control both of these kinds of adaptively timed behaviors.

6.8. Expected vs. Unexpected Disconfirmations Regulate Consummation vs. Exploration

Combining ART and START circuits into a larger architecture enables a brain to adaptively cope with situations wherein cues that have led to expected consequences in the past no longer do so. In particular, it enables humans to wait for delayed rewards, yet also prevents perseveration of behaviors to acquire a goal that is no longer forthcoming, with possibly disastrous consequences, such as starvation if food is no longer available. This competence is achieved by distinguishing *expected disconfirmations*—also called *expected non-occurrences*—of reward from *unexpected disconfirmations*—or *unexpected non-occurrences*—of reward.

In particular, why do not animals treat expected non-occurrences of reward as predictive failures? Why do they not always become frustrated by the immediate non-occurrence of a reliable reward that is typically delayed in time, and trigger exploratory behavior to find it elsewhere, leading to relentless exploration for immediate gratification? And if animals do wait, but the reward does not appear at the expected time, how does the animal adaptively respond to the unexpected non-occurrence of the reward—that is, to the occurrence of *nothing*? In normal animals, expected disconfirmations do not prevent acquisition of a delayed reward, even though unexpected disconfirmations can trigger reset of working memory, attention shifts, frustrative rebounds that can extinguish unsuccessful gated dipole associations, and the release of exploratory behaviors to discover better sources of the desired goal object.

In either case, if the reward happens to occur earlier than expected, the animal could still perceive it via a cognitive-emotional resonance and release a consummatory response. Thus, the registration of ART-like sensory matches is not inhibited during either expected or unexpected non-occurrences (**Figure 9**). However, during an expected disconfirmation, the *effects of mismatches* upon activation the ART orienting system, which cause a reduction of ART inhibition there (**Figure 9B**), are compensated by the addition of adaptively timed input from the HIPPO (**Figure 11C**). Activation of the orienting system is hereby prevented during an expected disconfirmation, and with

it reset of working memory, attention shifts, frustrative rebounds, and the release of exploratory behaviors. In contrast, during an unexpected non-occurrence, the orienting system is disinhibited by the ART mismatch because the spectral timing circuit is not active then, so reset of working memory, attention shifts, frustrative rebounds, and the release of exploratory behaviors can occur with which to correct the predictive error.

A spectral timing response begins immediately after its triggering stimulus, and builds throughout the interstimulus interval, or ISI, between the CS and US (Grossberg and Schmajuk, 1989; Grossberg and Merrill, 1992, 1996). It can thus maintain inhibition of the orienting system until the expected time of occurrence of the rewarding stimulus (**Figure 11C**). Adaptively timed excitation can also maintain motivated attention upon the correct orbitofrontal representation throughout this time interval (**Figure 11C**). By peaking at the expected time of the reward, motivated attention can most probably elicit a learned response when the reward is expected.

6.9. Working Memories and Learning of List Chunk Plans Using Item-List Resonances

During cognitive and cognitive-emotional learning and action cycles, as an animal or animat navigates through its environment, sequences of object categories may be temporarily stored in an *object* working memory (**Figure 4A**) that occurs in human and other primate brains in the ventrolateral prefrontal cortex (VLPFC), at the same time that sequences of the positions/directions where they are found in a scene are temporarily stored in a *spatial* working memory (**Figure 4B**) in the dorsolateral prefrontal cortex (DLPFC; see **Figure 5**).

As they are stored in working memory, object category sequences trigger learning of object plans, or *object list chunks*, while stored position/direction sequences trigger learning of spatial plans, or *spatial list chunks*, that selectively respond to the particular sequences that are stored in their working memory. A network that can learn list chunks of variable length is called a Masking Field (**Figure 4**) (Cohen and Grossberg, 1986, 1987; Grossberg and Kazeroounian, 2011; Kazeroounian and Grossberg, 2014). As illustrated in **Figure 13**, a Masking Field contains cells of variable size in which larger cells respond selectively to longer working memory lists. Masking Fields can learn these properties using simple laws of activity-dependent cell growth during their development, which leads to a multiple-scale network of *self-similar* cells whose cell body sizes and connection strengths covary (Cohen and Grossberg, 1987; Kazeroounian and Grossberg, 2014).

The learning of list chunks by a Masking Field in SOVEREIGN used only bottom-up adaptive filter pathways (**Figure 4**). *In vivo*, list chunk learning is dynamically stabilized by *item-list resonances* in the corresponding parts of the PFC (**Table 1a**). **Figure 13** illustrates the fact that the top-down learned expectation pathways that interact with bottom-up adaptive filter pathways to trigger and sustain an item-list resonance can also regulate choice of the most predictive list chunk in each environment and prime the sequences of working memory

**BOTH RECURRENT INHIBITION AND TOP-DOWN EXCITATORY PRIMING
ARE SELF-SIMILAR IN A MASKING FIELD**

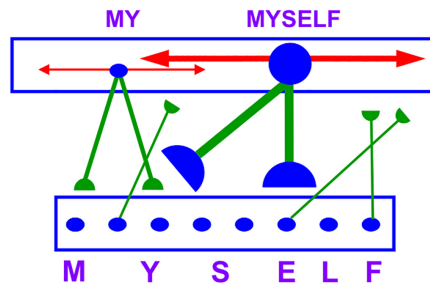


FIGURE 13 | An Item-Order-Rank working memory (lower level) for the short-term sequential storage of item chunks (e.g., M, Y, S, E, L, F) can activate a multiple-scale Masking Field list chunking network (upper level) through a bottom-up adaptive filter. The larger cell sizes and interaction strengths of the list chunks that categorize longer lists (e.g., MYSELF vs. MY) enable the Masking Field to choose the list chunk that currently receives the largest total input, and thus best predicts the sequence that is currently stored in the Item-Order-Rank working memory. The chosen list chunk can then read-out the most likely prediction of what will happen next in that temporal context. Green connections are excitatory. Red connections are inhibitory. Arrowheads at the ends of Masking Field inhibitory recurrent pathways denote connections that undergo no learning. Hemidisks denote connections that can undergo learning, both in the bottom-up filter connections and the top-down expectation connections. Recurrent on-center off-surround connections in the Item-and-Order working memory are not shown, for simplicity. Recurrent self-excitatory connections in the Masking Field are also not shown, again for simplicity. [Reprinted with permission from Grossberg (2018)].

items that support that choice. Such item-list resonances in SOVEREIGN2 can greatly increase the stability of this kind of learning under multiple kinds of perturbations.

6.10. Masking Fields Learn List Chunks From Resonating Item-Order-Rank Working Memories

These particular working memories and list chunking networks are used because they embody fundamental design principles that are needed for autonomous adaptive storage and learning of event sequences. In particular, feedback interactions between both types of circuits solve a Temporal Chunking Problem, which concerns how a new word, motor skill, or navigational route gets learned when it is composed of familiar subsequences, without undermining previous learning of the subsequences. In the case of language, for example, suppose that the new word is composed of syllables that are themselves already familiar words. The problem is: Why is not the brain forced to process the new word as a sequence of smaller familiar words? How does a not-yet-established word representation overcome the salience of already well-learned phoneme, syllable, or word representations to enable learning of the novel word to occur? How does this occur, moreover, under unsupervised learning conditions?

For example, suppose that the words MY, ELF, and SELF have already been learned, and have their own list chunks. When the novel word MYSELF is presented for the first time, all of its familiar subwords also get presented as part of this longer

sequence. What mechanisms prevent the familiarity of MY, ELF, and SELF, which are trying to activate their own list chunks, from forcing the novel longer list MYSELF from being processed as a sequence of these smaller familiar chunks, rather than eventually as a newly learned unitized whole? If this did happen, then longer words could never be learned. Nor could longer navigational routes that include familiar subroutes, or more complex motor skills that include familiar gestures. Our brains would experience a *reductio ad absurdum*. It is because the multiple scales of a Masking Field are self-similar that the larger scale that is activated by MYSELF can inhibit the smaller scales that are activated by MY, ELF, and SELF, even before the list chunk for MYSELF is tuned by category learning. The multiple self-similar spatial scales of Masking Fields hereby enable them to learn how to categorize lists of variable lengths.

Even if a novel longer list like MYSELF could overcome competition from its familiar subwords, what would prevent its new learning from forcing catastrophic forgetting of the list chunks of its familiar subwords? A solution of this problem is said to obey the LTM Invariance Principle. Item-Order-Rank working memories solve the LTM Invariance Principle (Grossberg, 1978, 2017; Bradski et al., 1992, 1994; Grossberg and Myers, 2000; Grossberg and Pearson, 2008; Grossberg and Kazeroonian, 2011; Silver et al., 2011; Kazeroonian and Grossberg, 2014). They store the temporal *order* of sequences of events occurring in time into an evolving spatial gradient of activities over content-addressable *item* representations that can represent items that are repeated multiple times; that is, have different *ranks* (e.g., ABACAD). Thus, Item-Order-Rank working memories can store sequences of events with repeats while satisfying the LTM Invariance Principle. They do so by preserving the *relative* activities of stored items as new items in a sequence are stored, even while the total activity of all stored items can change greatly through time.

Because all working memories need to satisfy the LTM Invariance Principle, *all* working memories, whether linguistic, motor, or spatial, were predicted to be realized by a similar kind of circuit. This circuit was shown to be a specialized version of a type of circuit that is ubiquitous in the brain; namely, a recurrent shunting on-center off-surround network, thereby clarifying how such a seemingly sophisticated design as a working memory could be discovered during evolution. Masking Fields are also recurrent shunting on-center off-surround networks, and thus are also working memories, albeit working memories that also represent list chunks.

Feedback interactions between an Item-Order-Rank working memory and a Masking Field solve the Temporal Chunking Problem, and can do so under unsupervised learning conditions. These feedback interactions trigger an *item-list resonance* that dynamically stabilizes the bottom-up list chunk learning and the learning of the top-down expectations that enable list chunks to activate sequences of events in working memory for skilled performance. Item-list resonances hereby illustrate how ART dynamics solve the stability-plasticity dilemma in the temporal domain, and include predictions about the oscillatory dynamics, including gamma and beta oscillations, that occur during these resonances in primate brains.

All of the predicted properties of Item-Order-Rank working memories have been supported by subsequent psychological data (e.g., Jones et al., 1995; Page and Norris, 1998; Farrell and Lewandowsky, 2004; Agam et al., 2005, 2007) and neurobiological data (e.g., Averbeck et al., 2002, 2003a,b; Bastos et al., 2018; Lundqvist et al., 2018).

In SOVEREIGN2, with item-list feedback signals implemented, each learned list chunk, or plan, can be selectively activated by motivationally salient sequences of previously experienced objects and positions/directions, and can then read out context-sensitive predictions of the objects and positions/directions that should be acquired next, thereby generalizing the SOVEREIGN interactions in **Figure 4**. This learning and performance cycle can continue through time in an unsupervised way using only the world itself as a teacher, but may also be supervised by a human teacher at arbitrary times. As noted in sections 6.5 and 6.6, CogEM includes supervision by rewards, punishments, and unexpected outcomes to drive its reinforcement learning.

6.11. Entorhinal-Hippocampal Resonances That Support Spatial Navigation Are Not Conscious

Yet another kind of resonance may be incorporated into SOVEREIGN2. This is the *entorhinal-hippocampal resonance* that supports learning and stable memory of entorhinal grid cells and hippocampal place cells during spatial navigation that were mentioned in section 5.2. This kind of resonance will be discussed in section 8. It illustrates the claim that, although “all conscious states are resonant states,” the converse statement is not true. In order for a resonant state to become conscious, it is necessary for it to include either representations of external sensory cues, such as visual or auditory cues, or internal sensory cues, such as emotional cues.

7. PREFRONTAL COORDINATION OF WORKING MEMORY, PLANNING, AND COGNITIVE-EMOTIONAL DYNAMICS

The kind of adaptive mobile intelligence that is exhibited by humans and other primates required a major expansion of the PFC to enable its working memory and planning networks to flexibly interact with multiple other brain systems, notably cognitive-emotional systems. The predictive ART, or pART, model (**Figure 5**) (Grossberg, 2018) has clarified how these properties arise through interactions of orbitofrontal cortex (OFC), VLPFC, and DLPFC with the inferotemporal cortex (ITp and ITa), perirhinal cortex (PRC), parahippocampal cortex (PHC), ventral bank of the principal sulcus (VPS), ventral prearcuate gyrus (VPA), frontal eye fields (FEF), hippocampus (HIPPO), amygdala (AMYG), basal ganglia (BG), hypothalamus (LH), PPC, lateral intraparietal cortex (LIP), and visual cortical areas V1, V2, V3A, V4, MT, and MST.

pART model explanations more fully embody and extend many of the processes that were included in SOVEREIGN,

including how the value of visual objects and events is computed, which objects and events cause desired consequences and which may be ignored as predictively irrelevant, and how to plan and act to realize these consequences. To achieve this properties, pART includes reinforcement learning and incentive motivational learning; object and spatial working memory dynamics; and category learning, including the learning of object categories, value categories, object-value categories, and sequence categories, or list chunks. pART also explains properties that go beyond SOVEREIGN and other neural models, such as how to selectively filter expected vs. unexpected events to determine which events get stored in working memory, and how such filtering controls movements toward, and conscious perception of, expected events.

Incorporating this level of sophistication in SOVEREIGN2 will require a coordinated research program. Here primarily the new competences will be reviewed of how events can be selectively filtered before being stored in working memory, and how that ability alters the understanding of how a top-down cognitive prime from the PFC can bias object attention in the What cortical stream to anticipate expected objects and events, while it also focuses spatial attention in the Where cortical stream to trigger actions that acquire currently valued objects (Fuster, 1973; Baldauf and Desimone, 2014; Bichot et al., 2015).

7.1. Minimal Anatomy for Foveating Valued Objects in a Scene: Where's Waldo?

As explained in greater detail in the pART model (Grossberg, 2018), after Where-to-What stream interactions help to learn invariant object categories, What-to-Where stream interactions regulate how to foveate valued target objects in a scene. Previous models like ARTSCAN Search and ARTSCENE Search proposed a *minimal anatomy* that could carry out this function, while also simulating challenging reaction time (RT) data about visual search for target objects (Huang and Grossberg, 2010; Chang et al., 2014). Such a minimal anatomy models how an invariant object representation in the What stream can activate a positional representation in the Where stream that can be used to foveate a valued target object in a scene. However, it did not try to solve the problem of how the brain can selectively filter desired targets from a stream that also contains distractors, so that it only attends, stores, and foveates matched targets. This additional computational property is explained by the pART model (**Figure 5**). However, given the ability of the minimal anatomy to quantitatively simulate challenging RT data in many visual search experiments, it may have evolved before the prefrontal mechanisms of selective working memory storage did, and may operate in parallel with them. It may be worth testing if these simpler circuits are still functional when prefrontal mechanisms are lesioned.

In the minimal anatomy of ARTSCENE Search, winning VLPFC activities send a top-down attentional prime to ITa using a circuit that obeys the ART Matching Rule. In order to transform the primed ITa cells into firing cells, an additional input must converge on ITa. This kind of signal is regulated by the BG

(cf. BG in **Figure 5**). A volitional gate-opening signal from the BG—notably from the substantia nigra pars reticulata, or SNr—lets the primed ITa cells fire. The activated ITa cells then prime the positionally sensitive categories in ITp with which they were associated when ITa was being learned using resonant bottom-up and top-down interactions (**Figure 5**). If one of the primed ITp categories also receives a bottom-up input from an object at its position, then it can fire and activate positional representations in eye movement control regions like LIP and FEF. These positional representations can then move the eyes to the position in space that they represent.

7.2. Cortical What Working Memory Filtering and Activation of Where Target Positions

Multiple experiments show that selective working memory storage in the PFC does occur. The pART model offers an explanation of how this is predicted to work (**Figure 5**). For example, PFC working memory cells do not fire during such tasks that do not require storage of visual information (Fuster, 1973; Kojima and Goldman-Rakic, 1984). Moreover, given the presentation of identical stimuli, neural selectivity in PFC depends on subsequent task demands (Warden and Miller, 2010). Imaging data show that success on working memory tasks covaries with an individual's ability to selectively identify and store task-related stimuli from a larger sequence of stimuli (Awh and Vogel, 2008; McNab and Klingberg, 2008). Subliminal distracters can damage performance in attention tasks, but making distracters supra-threshold can improve performance deficits by facilitating the ability to filter them out (Tsushima et al., 2008). During a memory saccade task in which a salient distractor is flashed at a variable time and position during the memory delay, responses to the salient distractor are more strongly suppressed and correlated with performance in DLPFC than in LIP (Suzuki and Gottlieb, 2013).

In addition to this kind of task-sensitive filtering of individual items *before* they reach the working memory, a mechanistically distinct processes enables all the items that get through the filter to be stably stored *after* they reach the working memory; namely, keeping an SNr gate open to enable the recurrent excitatory connections within PFC to maintain working memory storage. Closing this SNr gate can rapidly reset, or delete, the entire stored sequence from working memory when there is an attention shift to do a different task.

7.3. Interacting Feature-Based Attention, Saccadic Choice, and Selective Working Memory Storage

The property of selective working memory storage clarifies the functional role of neurophysiological data about the role of VPA as “a source for feature-based attention” (Bichot et al., 2015, p. 832), notably why VPA cells selectively match desired combinations of object features, resonate with a target that matches these features, and activate an FEF positional representation that commands a saccade to the target. These properties were discovered when fixating monkeys were

presented with a central cue object that defined a search target, followed by a delay during which the monkeys held a representation of the target in memory. Then an array of eight stimuli appeared which included the search target and seven distractors. The monkeys were rewarded for foveating and maintaining fixation on the target for 800 ms. While the monkeys performed, Bichot et al. (2015) simultaneously recorded from IT, VPA, and FEF in two monkeys, and VPS, VPA, and FEF in two other monkeys.

pART proposes the following mechanistic and functional explanation of how these cells interact together to enable matched objects to be selectively processed and stored by PFC (**Figure 5**): Both ITp (TEO) and ITa (TE) topographically project to PFC (Barbas and Pandya, 1989; Webster et al., 1994; Tanaka, 1996). The ITp projection is to VPA, whose cells, just like the ones in ITp (Tanaka, 1996), exhibit significant sensitivity to extrafoveal positions (Bichot et al., 2015). The ITa projections are to PRC and VPS, which in turn projects to VLPFC. In the data of Bichot et al. (2015), VPS had the largest spatial tuning curves of any cells in their data, consistent with ITa invariance properties.

Active VLPFC top-down signals project to both VPS and VPA, and learn modulatory top-down expectations when VPS and VPA cells are also active. In pART, these expectations obey the ART Matching Rule that is realized by a top-down, modulatory on-center, off-surround network.

VPA cells that receive a previously learned VLPFC-to-VPA prime are enhanced when an extrafoveal object matches its target features, and are suppressed when the object mismatches them, properties that are consistent with the ART Matching Rule. This enhanced VPA activity is sufficient to trigger an output signal to FEF at the corresponding FEF positional representation in FEF. This property is supported by Bichot et al. (2015) data showing VPA activating around 20 ms. before FEF does. FEF can then trigger a saccade to foveate the target. Because objects that mismatch the VPA expectation are inhibited, they are not foveated.

A similar match-mismatch dichotomy regulates the activity of VPS cells when they receive an active VLPFC-to-VPS prime. Their activity is enhanced when an ITa invariant object category matches their receptive field, and are suppressed by a mismatch, again consistent with the ART Matching Rule. When a match occurs, a synchronous VPS-ITa resonance develops that enables the category's temporal order to be stored in VLPFC. This resonance can also propagate top-down through multiple cortical areas (e.g., ITa-ITp-V4-V2-V1 in **Figure 5**) and supports conscious recognition of the object.

8. LEARNING THE PRESENT POSITION IN SPACE OF A NAVIGATOR USING GRID CELLS AND PLACE CELLS

Section 5.2 noted that a representation of an animal's Present Position Vector, or NET, as it navigates in space is derived in SOVEREIGN from an algorithm that computes a head/body turn angle as well the length of the next straight distance that is navigated. That section also noted that, in order for

NET to be computed without algorithmic short cuts, an animal or animat needs to learn a representation of its present position in space as it navigates in different environments. The GridPlaceMap model of spatial navigation (**Figure 14A**) proposes how entorhinal grid cells and hippocampal place cells accomplish this as they are learned in a hierarchy of self-organizing maps. This model forms part of a larger entorhinal-hippocampal system that shows how learning of these maps may be dynamically stabilized by an entorhinal-hippocampal resonance (see section 6.11; **Figure 14B**; Grossberg and Pilly, 2014). This larger system explains why hippocampal place cells may be viewed as learned spatial categories in an entorhinal-hippocampal ART system that enables a stable computation of NET to be autonomously learned in a wide variety of navigated environments.

The GridPlaceMap model and its variants have explained and simulated many behavioral and neurobiological data about spatial navigation and how its circuits learn and remember (e.g., Grossberg and Pilly, 2012, 2014; Mhatre et al., 2012; Pilly and Grossberg, 2012; Grossberg, 2013; Grossberg et al., 2014). A comprehensive review of such data goes beyond the explanatory goals of the current exposition. Some basic facts are nonetheless worth mentioning here:

The model responds to realistic rat navigational trajectories by learning both grid cells with hexagonal grid firing fields of multiple spatial scales, and place cells with one or more firing fields, that match neurophysiological data about their development in juvenile rats. The fact that individual grid cells can fire at positions on a hexagonal lattice when rats navigate in an open field is one of the most remarkable facts in contemporary neuroscience (Hafting et al., 2005). The GridPlaceMap model and its variants show that this property emerges in a grid cell self-organizing map model (**Figure 14**) as a result of basic trigonometric properties of navigation in a two-dimensional space. The fact that hippocampal place cells may be viewed as learned spatial categories in an entorhinal-hippocampal ART system that are dynamically stabilized by top-down attention from hippocampal cortex to entorhinal cortex is supported by neurophysiological data from several labs (Morris and Frey, 1997; Kentros et al., 1998, 2004; Bonnevie et al., 2013).

Other properties of the GridPlaceMap model are also worth summarizing both because they are so parsimonious and data-predictive, and because they will simplify their embodiment in SOVEREIGN 2. For example, the *same* self-organizing map model equations can learn both grid cells and place cells. The different response properties seem to arise entirely due to their different stages of processing in a hierarchy of self-organizing maps (**Figure 14B**). In this hierarchy, hexagonal grid cell response fields are learned in response to stripe cells, which are derived from vestibular angular head velocity and linear velocity signals as realistic spatial trajectories are navigated (**Figure 14**). Place cells with unimodal response fields are learned in response to inputs from the emerging grid cells. Despite their very different response properties, both grid cells and place cells can develop by detecting, learning, and remembering the most frequent and energetic co-occurrences of their inputs. Because each place cell learns to respond to grid cells of several

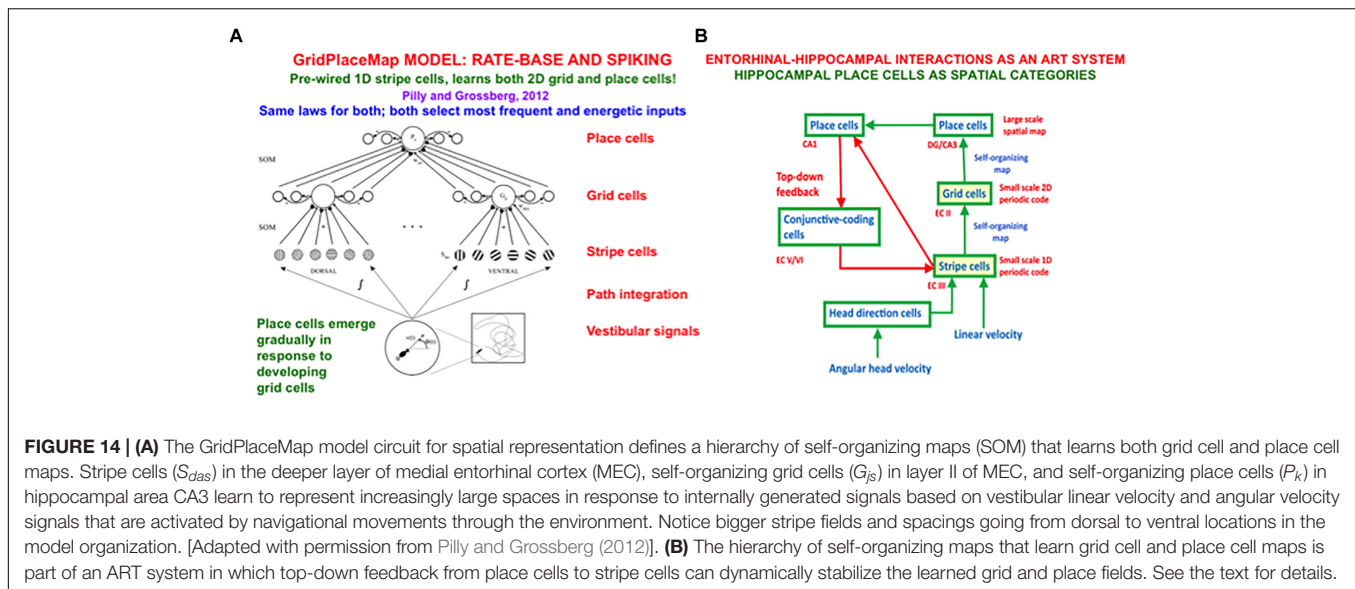
different spatial scales, the spatial scale of the resulting place cell is the *least common multiple* of the grid cell scales that input to it. Thus grid cells that respond on a centimeter scale can support learning of place cells that can represent spaces that are many meters in size.

Figure 14B also includes the known direct pathway from entorhinal cortex (EC111) to the hippocampal CA1 region that bypasses the grid cells. This pathway may learn place cells in CA1 with small spatial scales while, for example, rat pups are still in their nests. An explosion of coordinated grid cell and place cell development occurs as rats emerge from their nests (Langston et al., 2010; Wills et al., 2010), and presumably helps to learn the much larger spatial scales that are needed for adult spatial navigation.

Parsimonious properties also occur at the earliest stages of the GridPlaceMap model. For example, similar *ring attractor* networks are used to convert vestibular *angular velocity* signals into responses of head direction cells, and *linear velocity* signals into responses of stripe cells (**Figure 14B**). Both spatial and temporal learning in the entorhinal-hippocampal system seem to use homologous mechanisms to create a gradient from small to large scales along a dorsoventral axis. The temporal learning is the adaptively timed hippocampal learning that was described in sections 6.7 and 6.8. In particular, during both spatial and temporal learning, cells in different positions along the gradient respond at slower rates from dorsal to ventral. Spatial learning of grid cells and place cells along the dorsoventral axis passes through the *medial* entorhinal cortex to HIPPO, with the largest grid and place cell spatial scales occurring at ventral positions. Spatial learning hereby converts slower cell response rates into larger learned spatial scales. Temporal learning along the dorsoventral axis passes through the *lateral* entorhinal cortex to HIPPO, with the longest time intervals spanned at the most ventral positions in this gradient. Temporal learning uses spectrally timed conditioning with cells in the spectrum responding more slowly at more ventral positions (**Figure 11C**).

This computational homology provides a harmonious explanation of why both spatial and temporal representations occur in the entorhinal-hippocampal system. Many challenging neurophysiological data are explained by this homology between spatial learning in the medial entorhinal-hippocampal system and adaptively timed temporal learning in the lateral entorhinal-hippocampal system (e.g., Hargreaves et al., 2005; Aminoff et al., 2007; Kerr et al., 2007; Eichenbaum and Lipton, 2008; van Strien et al., 2009; Keene et al., 2016). When comparing these spatial and temporal circuits, the GridPlaceMap model is called *spectral spacing* to match the term spectral timing. The computational homology between them is called *neural relativity*.

The top-down hippocampus-to-entorhinal attentional network that stabilizes map learning uses the same ART Matching Rule that stabilizes learning of all ART circuits, including object categories learned via a feature-category resonance (**Figures 7, 9**). In the entorhinal-hippocampal system, this attentive matching process helps to explain neurobiological data about theta, beta, and gamma oscillations, such as, as mentioned above, why there is an Inverted-U through time in the power of beta oscillations when an animal first navigates a



new maze (Berke et al., 2008; Grossberg, 2009a). Also explained are data about how hippocampal, septal, or acetylcholine inactivation may disrupt grid cell learning and performance.

9. CONCLUDING REMARKS

This article summarizes basic design principles, networks, and functional capabilities of the SOVEREIGN architecture (Gnadt and Grossberg, 2008) and outlines a major research program whereby additional brain mechanisms and psychological functions can be consistently added to create a SOVEREIGN2 architecture with much greater capabilities for autonomous adaptive navigation and goal-oriented cognition, emotion, and action in changing environments.

SOVEREIGN was designed to serve as an autonomous neural system for incrementally learning planned action sequences to navigate toward a rewarded goal. SOVEREIGN also illustrates how brains may, at several different organizational levels, regulate the balance between reactive and planned behaviors, and proposes how homologous circuit designs regulate spatial navigation and reaching behaviors. These capabilities were demonstrated by learning efficient routes whereby to navigate to a valued goal in a virtual reality environment.

Some of the designs in SOVEREIGN were realized algorithmically, and can be realized dynamically in SOVEREIGN2. Other processes that are needed to achieve a more comprehensive autonomous adaptive intelligence in an embodied mobile system were not included at all. This article summarizes neural models of important missing capabilities with enough detail to define a research program that can consistently incorporate them into SOVEREIGN2. Missing designs occur across both the What and Where processing streams of SOVEREIGN (e.g., Figure 4).

In order to include these missing designs, SOVEREIGN2 embodies foundational brain design principles such as

complementary computing, hierarchical resolution of uncertainty, and adaptive resonance that enable biological brains to realize their autonomous adaptive intelligence. Some of the missing designs in the What stream occur at early processing stages, such as visual boundary completion and surface filling-in. These processes require hierarchical resolution of uncertainty to be completed. How this occurs sheds light on deep computational reasons for how and why animals like humans and other primates become conscious in order to generate effective actions.

Other missing What stream processes occur at higher processing stages, such as autonomous learning of view-, position-, and size-invariant recognition categories. Such invariant learning requires modulatory interactions from parietal regions of the Where cortical stream to inferotemporal regions of the What cortical stream in order to ensure that only views of a single object get bound together by associative learning in a single invariant object category. The surface-shroud resonances that support invariant category learning also play a role in enabling social cognitive skills such as joint attention and imitation learning to occur between a teacher and a student who experience the world through different spatial perspectives.

Still higher levels of processing have parallel object and spatial processing systems in both the What and Where cortical streams. For example, prefrontal object and spatial working memories need to be able to selectively filter targets from distractors before storing them and their target positions in working memory. The filtering machinery that does this also allows attention to be paid to salient targets, and to use those targets to drive orienting movements toward them.

Cognitive-emotional circuits are needed to enhance predictions and actions that lead to valued outcomes, and to attenuate those that do not. In order to do this effectively, cognitive-emotional learning needs to be able to associate sensory and rewarding cues that are separated in time. Spectral timing circuits in the HIPPO help to support cognitive-emotional learning in inferotemporal-amygdala/

hypothalamus-orbitofrontal circuits. These circuits, in turn, amplify or suppress cognitive and spatial working memory circuits and plans according to whether they generate successful goal-oriented actions or not.

Although looking and reaching behaviors can use target position and present position estimates that can both be directly computed from either external sensory cues or internally generated movement commands, navigational movements need more sophisticated networks to learn a navigator's present position in space. Entorhinal grid cells and hippocampal place cells interact to incrementally learn place cells that can represent spatial scales that are sufficiently large to support navigation in ecologically relevant spaces. These learned spatial categories are dynamically stabilized using the same Adaptive Resonance Theory, or ART, Matching Rule that is found in the resonant dynamics of many of the missing competences from which SOVEREIGN2 can benefit.

These resonances include feature-category resonances, surface-shroud resonances, cognitive-emotional resonances, entorhinal-hippocampal resonances, and item-list resonances. All of these resonances help to dynamically stabilize the learned memories of their respective networks, and thereby enable them to successfully operate in open-ended non-stationary environments without experiencing the learning and forgetting problems, notably catastrophic forgetting, that plagues all algorithms of back propagation type, including the currently popular and useful Deep Learning algorithms, and Bayesian Explaining Away algorithms, among others.

AUTHOR CONTRIBUTIONS

The author confirms being the sole contributor of this work and has approved it for publication.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fncom.2019.00036/full#supplementary-material>

REFERENCES

- Agam, Y., Bullock, D., and Sekuler, R. (2005). Imitating unfamiliar sequences of connected linear motions. *J. Neurophysiol.* 94, 2832–2843. doi: 10.1152/jn.00366.2005
- Agam, Y., Galperin, H., Gold, B. J., and Sekuler, R. (2007). Learning to imitate novel motion sequences. *J. Vis.* 7, 1–17. doi: 10.1167/7.5.1
- Aminoff, E., Gronau, N., and Bar, M. (2007). The parahippocampal cortex mediates spatial and nonspatial associations. *Cereb. Cortex* 17, 1493–1503. doi: 10.1093/cercor/bhl078
- Andersen, R. A., Essick, G. K., and Siegel, R. M. (1985). Encoding of spatial location by posterior parietal neurons. *Science* 230, 456–458. doi: 10.1126/science.4048942
- Averbeck, B. B., Chafee, M. V., Crowe, D. A., and Georgopoulos, A. P. (2002). Parallel processing of serial movements in prefrontal cortex. *Proc. Natl. Acad. Sci. U.S.A.* 99, 13172–13177. doi: 10.1073/pnas.162485599

FIGURE S1 | Smooth pursuit of a target moving with a fixed speed and direction creates retinal slip signals on the retina until the target is foveated, as well as background motion signals in the opposite direction. As the target is acquired, the background motion signals increase, and can maintain predictive pursuit that maintains the target on the fovea. See the text for details. [Reprinted with permission from Pack et al. (2001)].

FIGURE S2 | (a) A leftward eye movement channel. All connections are excitatory. The retinal image is processed by two types of cells in MT. MT cells with inhibitory surrounds (MT^-) connect to MSTv cells, with MT cells preferring greater speeds weighted more heavily. MT cells with excitatory surrounds (MT^+) connect to MSTd cells. MSTv cells have excitatory connections with MSTd cells that prefer opposite directions. MSTv cells drive pursuit eye movements in their preferred direction, and the resulting eye velocity is fed back to MSTv and MSTd cells (thick arrows). Leftward eye rotation causes rightward retinal motion of the background. The MT and MST cells are drawn so as to approximate their relative receptive field sizes. (b) Model MST connectivity. Excitatory connections are shown by solid lines. Inhibitory connections are indicated by dashed lines. Thick line emanating from the pursuit pathway indicate efference copy inputs. The leftward eye movement channel consists of an MSTv cell preferring leftward motion and an MSTd cell preferring rightward motion, and receives an efference copy signaling leftward eye movement. The rightward eye channel is defined analogously. [Reprinted with permission from Pack et al. (2001)].

FIGURE S3 | In this figure, black boxes denote areas belonging to the saccadic eye movement system (SAC), white boxes the smooth pursuit eye movement system (SPEM), and gray boxes, both systems. The abbreviations for the different brain regions are: LIP, lateral intra-parietal area; FPA, frontal pursuit area; MST, middle superior temporal area; MT, middle temporal area; FEF, frontal eye fields; NRTP, nucleus reticularis tegmenti pontis; DLPN, dorso-lateral pontine nuclei; SC, superior colliculus; CBM, cerebellum; MVN/rLVN, medial and rostro-lateral vestibular nuclei; PPRF, a peri-pontine reticular formation; TN, tonic neurons. Although an analysis of how this system works is beyond the scope of this article, the macrocircuit does serve as a reminder that seemingly effortless behavioral competences are often emergent properties of beautifully coordinated brain dynamics among multiple brain regions with different functional roles to play [Reprinted with permission from Grossberg et al. (2012)].

FIGURE S4 | Two views of the eye and retina. The top image shows a drawing of a cross-sectional cut through the eye showing the retinal veins occluding the light coming into the pupil before it reaches the photoreceptors. The photoreceptors send axons to the brain via the optic nerve which, as seen in the bottom image of a top-down view of retina, creates a blind spot that is comparable in size to the fovea [Adapted with permission from Kolb, Fernandez, and Anderson (<http://retina.umh.es/Webvision/sretina.html>)].

FIGURE S5 | This image emphasizes that, even the retinal image of a simple object like a line can be occluded in multiple places by retinal veins and the blind spot, thereby creating multiple positions along the line that do not provide reliable inputs to the brain for directing actions to those positions.

- Averbeck, B. B., Crowe, D. A., Chafee, M. V., and Georgopoulos, A. P. (2003a). Neural activity in prefrontal cortex during copying geometrical shapes. I. single cells encode shape, sequence, and metric parameters. *Exp. Brain Res.* 150, 127–141. doi: 10.1007/s00221-003-1416-6
- Averbeck, B. B., Crowe, D. A., Chafee, M. V., and Georgopoulos, A. P. (2003b). Neural activity in prefrontal cortex during copying geometrical shapes. II. decoding shape segments from neural ensembles. *Exp. Brain Res.* 150, 142–153. doi: 10.1007/s00221-003-1417-5
- Awh, E., and Vogel, E. K. (2008). The bouncer in the brain. *Nat. Neurosci.* 11, 5–6. doi: 10.1038/nn0108-5
- Baldauf, D., and Desimone, R. (2014). Neural mechanisms of object-based attention. *Science* 344, 424–427. doi: 10.1126/science.1247003
- Banquet, J. P., and Grossberg, S. (1987). Probing cognitive processes through the structure of event-related potentials during learning: an experimental and theoretical analysis. *Appl. Opt.* 26, 4931–4946. doi: 10.1364/AO.26.004931

- Barbas, H., and Pandya, D. N. (1989). Architecture and intrinsic connections of the prefrontal cortex in the rhesus monkey. *J. Comp. Neurol.* 286, 353–375. doi: 10.1002/cne.902860306
- Bastos, A. M., Loonis, R., Kornblith, S., Lundqvist, M., and Miller, E. K. (2018). Laminar recordings in frontal cortex suggest distinct layers for maintenance and control of working memory. *Proc. Natl. Acad. Sci.* 115, 1117–1122. doi: 10.1073/pnas.1710323115
- Berke, J. D., Hetrick, V., Breck, J., and Green, R. W. (2008). Transient 23–30-Hz oscillations in mouse hippocampus during exploration of novel environments. *Hippocampus* 18, 519–529. doi: 10.1002/hipo.20435
- Berzhanskaya, J., Grossberg, S., and Mingolla, E. (2007). Laminar cortical dynamics of visual form and motion interactions during coherent object motion perception. *Spat. Vis.* 20, 337–395. doi: 10.1163/156856807780919000
- Bichot, N. P., Heard, M. T., and DeGennaro, E. M. (2015). A source for feature-based attention in the prefrontal cortex. *Neuron* 88, 832–844. doi: 10.1016/j.neuron.2015.10.001
- Bonnevie, T., Dunn, B., Fyhn, M., Hafting, T., Derdikman, D., Kubie, J. L., et al. (2013). Grid cells require excitatory drive from the hippocampus. *Nat. Neurosci.* 16, 309–319. doi: 10.1038/nn.3311
- Booth, M. C., and Rolls, E. T. (1998). View-invariant representations of familiar objects by neurons in the inferior temporal visual cortex. *Cereb. Cortex* 8, 510–523. doi: 10.1093/cercor/8.6.510
- Bootman, M. D., and Berridge, M. J. (1996). Subcellular Ca²⁺ signals underlying waves and graded responses in HeLa cells. *Curr. Biol.* 6, 855–865. doi: 10.1016/s0960-9822(02)00609-7
- Bradski, G., Carpenter, G., and Grossberg, S. (1992). Working memory networks for learning temporal order with application to 3-D visual object recognition. *Neural Comput.* 4, 270–286. doi: 10.1162/neco.1992.4.2.270
- Bradski, G., Carpenter, G. A., and Grossberg, S. (1994). STORE working memory networks for storage and recall of arbitrary temporal sequences. *Biol. Cybernet.* 71, 469–480. doi: 10.1007/s004220050106
- Brown, J., Bullock, D., and Grossberg, S. (1999). How the basal ganglia use parallel excitatory and inhibitory learning pathways to selectively respond to unexpected rewarding cues. *J. Neurosci.* 19, 10502–10511. doi: 10.1523/jneurosci.19-23-10502.1999
- Brown, J. W., Bullock, D., and Grossberg, S. (2004). How laminar frontal cortex and basal ganglia circuits interact to control planned and reactive saccades. *Neural Netw.* 17, 471–510. doi: 10.1016/j.neunet.2003.08.006
- Bullock, D., Cisek, P., and Grossberg, S. (1998). Cortical networks for control of voluntary arm movements under variable force conditions. *Cereb. Cortex* 8, 48–62. doi: 10.1093/cercor/8.1.48
- Bullock, D., and Grossberg, S. (1988). Neural dynamics of planned arm movements: emergent invariants and speed-accuracy properties during trajectory formation. *Psychol. Rev.* 95, 49–90. doi: 10.1037/0033-295x.95.1.49
- Bullock, D., Grossberg, S., and Guenther, F. H. (1993). A self-organizing neural model of motor equivalent reaching and tool use by a multijoint arm. *J. Cogn. Neurosci.* 5, 408–435. doi: 10.1162/jocn.1993.5.4.408
- Cannell, M. B., Cheng, H., and Lederer, W. J. (1995). The control of calcium release in heart muscle. *Science* 268, 1045–1049. doi: 10.1126/science.7754384
- Cao, Y., Grossberg, S., and Markowitz, J. (2011). How does the brain rapidly learn and reorganize view- and positionally-invariant object representations in inferior temporal cortex? *Neural Netw.* 24, 1050–1061. doi: 10.1016/j.neunet.2011.04.004
- Carpenter, G. A., and Grossberg, S. (1987a). A massively parallel architecture for a self-organizing neural pattern recognition machine. *Comput. Vis. Graph. Image Process.* 37, 54–115. doi: 10.1016/s0734-189x(87)80014-2
- Carpenter, G. A., and Grossberg, S. (1987b). ART 2: Stable self-organization of pattern recognition codes for analog input patterns. *Appl. Opt.* 26, 4919–4930. doi: 10.1364/AO.26.004919
- Carpenter, G. A., and Grossberg, S. (1990). ART 3: Hierarchical search using chemical transmitters in self-organizing pattern recognition architectures. *Neural Netw.* 3, 129–152. doi: 10.1016/0893-6080(90)90085-y
- Carpenter, G. A., and Grossberg, S. (1991). *Pattern Recognition by Self-Organizing Neural Networks*. Cambridge, MA: MIT Press.
- Carpenter, G. A., and Grossberg, S. (1993). Normal and amnesic learning, recognition, and memory by a neural model of cortico-hippocampal interactions. *Trends Neurosci.* 16, 131–137. doi: 10.1016/0166-2236(93)90118-6
- Carpenter, G. A., Grossberg, S., and Leshner, G. W. (1998). The what-and-where filter: a spatial mapping neural network for object recognition and image understanding. *Comput. Vis. Image Understand.* 69, 1–22. doi: 10.1006/cvui.1997.0561
- Carpenter, G. A., Grossberg, S., Markuzon, N., Reynolds, J. H., and Rosen, D. B. (1992). Fuzzy ARTMAP: a neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Trans. Neural Netw.* 3, 698–713. doi: 10.1109/72.159059
- Carpenter, G. A., Grossberg, S., and Mehanian, C. (1989). Invariant recognition of cluttered scenes by a self-organizing ART architecture: CORT-X boundary segmentation. *Neural Netw.* 2, 169–181. doi: 10.1016/0893-6080(89)90002-6
- Carpenter, G. A., Grossberg, S., and Reynolds, J. H. (1991a). ARTMAP: supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Netw.* 4, 565–588. doi: 10.1016/0893-6080(91)90012-t
- Carpenter, G. A., Grossberg, S., and Rosen, D. B. (1991b). Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Netw.* 4, 759–771. doi: 10.1016/0893-6080(91)90056-b
- Carpenter, G. A., Martens, S., and Ogas, O. J. (2005). Self-organizing information fusion and hierarchical knowledge discovery: a new framework using ARTMAP neural networks. *Neural Netw.* 18, 287–295. doi: 10.1016/j.neunet.2004.12.003
- Carpenter, G. A., and Ravindran, A. (2008). “Unifying multiple knowledge domains using the ARTMAP information fusion system,” in *Proceedings of the 11th International Conference on Information Fusion*, Cologne. doi: 10.1016/j.neunet.2004.12.003
- Chang, H.-C., Grossberg, S., and Cao, Y. (2014). Where’s Waldo? How perceptual cognitive, and emotional brain processes cooperate during learning to categorize and find desired objects in a cluttered scene. *Front. Integr. Neurosci.* 8:43. doi: 10.3389/fnint.2014.0043
- Chang, C., and Gaudiano, P. (1998). Application of biological learning theories to mobile robot avoidance and approach behaviors. *Adv. Complex. Syst.* 1, 79–114.
- Chey, J., Grossberg, S., and Mingolla, M. (1997). Neural dynamics of motion grouping: from aperture ambiguity to object speed and direction. *J. Opt. Soc. Am.* 14, 2570–2594.
- Chiu, Y.-C., and Yantis, S. (2009). A domain-independent source of cognitive control for task sets: shifting spatial attention and switching categorization rules. *J. Neurosci.* 29, 3930–3938. doi: 10.1523/jneurosci.5737-08.2009
- Cisek, P., Grossberg, S., and Bullock, D. (1998). A cortico-spinal model of reaching and proprioception under multiple task constraints. Boston University. *J. Cogn. Neurosci.* 10, 425–444. doi: 10.1162/089892998562852
- Cohen, M. A., and Grossberg, S. (1986). Neural dynamics of speech and language coding: developmental programs, perceptual grouping, and competition for short-term memory. *Hum. Neurobiol.* 5, 1–22.
- Cohen, M. A., and Grossberg, S. (1987). Masking fields: a massively parallel neural architecture for learning, recognizing, and predicting multiple groupings of patterned data. *Appl. Opt.* 26, 1866–1891. doi: 10.1364/AO.26.001866
- Daniel, P. M., and Whitteridge, D. (1961). The representation of the visual field on the cerebral cortex in monkeys. *J. Physiol.* 159, 203–221. doi: 10.1113/jphysiol.1961.sp006803
- DeYoe, E. A., and Van Essen, D. C. (1988). Concurrent processing streams in monkey visual cortex. *Trends Neurosci.* 11, 219–226. doi: 10.1016/0166-2236(88)90130-0
- Dranias, M., Grossberg, S., and Bullock, D. (2008). Dopaminergic and non-dopaminergic value systems in conditioning and outcome-specific revaluation. *Brain Res.* 1238, 239–287. doi: 10.1016/j.brainres.2008.07.013
- Eichenbaum, H., and Lipton, P. A. (2008). Towards a functional organization of the medial temporal lobe memory system: role of the parahippocampal and medial entorhinal cortical areas. *Hippocampus* 18, 1314–1324. doi: 10.1002/hipo.20500
- Farrell, S., and Lewandowsky, S. (2004). Modeling transposition latencies: constraints for theories of serial order memory. *J. Mem. Lang.* 51, 115–135. doi: 10.1016/s0749-596x(04)00029-4
- Fazl, A., Grossberg, S., and Mingolla, E. (2009). View-invariant object category learning, recognition, and search: how spatial and object attention are coordinated using surface-based attentional shrouds. *Cogn. Psychol.* 58, 1–48. doi: 10.1016/j.cogpsych.2008.05.001

- Fiala, J. C., Grossberg, S., and Bullock, D. (1996). Metabotropic glutamate receptor activation in cerebellar Purkinje cells as substrate for adaptive timing of the classically conditioned eye blink response. *J. Neurosci.* 16, 3760–3774. doi: 10.1523/jneurosci.16-11-03760.1996
- Finch, E. A., and Augustine, G. J. (1998). Local calcium signalling by inositol-1,4,5-triphosphate in Purkinje cell dendrites. *Nature* 396, 753–756. doi: 10.1038/25541
- Fischer, B. (1973). Overlap of receptive field centers and representation of the visual field in the cat's optic tract. *Vis. Res.* 13, 2113–2120. doi: 10.1016/0042-6989(73)90188-0
- Foley, N. C., Grossberg, S., and Mingolla, E. (2012). Neural dynamics of object-based multifocal visual spatial attention and priming: object cueing, useful-field-of-view, and crowding. *Cogn. Psychol.* 65, 77–117. doi: 10.1016/j.cogpsych.2012.02.001
- Franklin, D. J., and Grossberg, S. (2017). A neural model of normal and abnormal learning and memory consolidation: adaptively timed conditioning, hippocampus, amnesia, neurotrophins, and consciousness. *Cogn. Affect. Behav. Neurosci.* 17, 24–76. doi: 10.3758/s13415-016-0463-y
- French, R. M. (1999). Catastrophic forgetting in connectionist networks. *Trends Cogn. Sci.* 3, 128–135. doi: 10.1016/s1364-6613(99)01294-2
- Fuster, J. M. (1973). Unit activity in prefrontal cortex during delayed-response performance: neuronal correlates of transient memory. *J. Neurophysiol.* 36, 61–78. doi: 10.1152/jn.1973.36.1.61
- Gaudiano, P., and Grossberg, S. (1991). Vector associative maps: Unsupervised real-time error-based learning and control of movement trajectories. *Neural Netw.* 4, 147–183. doi: 10.1016/0893-6080(91)90002-m
- Gaudiano, P., and Grossberg, S. (1992). Adaptive vector integration to endpoint: self-organizing neural circuits for control of planned movement trajectories. *Hum. Mov. Sci.* 11, 141–155. doi: 10.1016/0167-9457(92)90056-h
- Gnadt, J. W., and Andersen, R. A. (1988). Memory related motor planning activity in posterior parietal cortex of macaque. *Exp. Brain Res.* 70, 216–220.
- Gnadt, W., and Grossberg, S. (2008). SOVEREIGN: an autonomous neural system for incrementally learning planned action sequences to navigate towards a rewarded goal. *Neural Netw.* 21, 699–758. doi: 10.1016/j.neunet.2007.09.016
- Goodale, M. A., and Milner, A. D. (1992). Separate visual pathways for perception and action. *Trends Neurosci.* 15, 20–25. doi: 10.1016/0166-2236(92)90344-8
- Goodale, M. A., Milner, A. D., Jakobson, L. S., and Carey, D. P. (1991). A neurological dissociation between perceiving objects and grasping them. *Nature* 349, 154–156. doi: 10.1038/349154a0
- Grossberg, S. (1971). On the dynamics of operant conditioning. *J. Theor. Biol.* 33, 225–255.
- Grossberg, S. (1972a). A neural theory of punishment and avoidance, I: qualitative theory. *Math. Biosci.* 15, 39–67. doi: 10.1016/0025-5564(72)90062-4
- Grossberg, S. (1972b). A neural theory of punishment and avoidance, II: quantitative theory. *Math. Biosci.* 15, 253–285. doi: 10.1016/0025-5564(72)90038-7
- Grossberg, S. (1976a). Adaptive pattern classification and universal recoding, I: parallel development and coding of neural feature detectors. *Biol. Cyber.* 23, 121–134.
- Grossberg, S. (1976b). Adaptive pattern classification and universal recoding, II: feedback, expectation, olfaction, and illusions. *Biol. Cyber.* 23, 187–202.
- Grossberg, S. (1978). “A theory of human memory: Self-organization and performance of sensory-motor codes, maps, and plans,” in *Progress in Theoretical Biology*, Vol. 5, eds R. Rosen and F. Snell (New York, NY: Academic Press), 233–374. doi: 10.1016/b978-0-12-543105-7.50013-0
- Grossberg, S. (1980). How does a brain build a cognitive code? *Psychol. Rev.* 87, 1–51. doi: 10.1037//0033-295x.87.1.1
- Grossberg, S. (1982). Processing of expected and unexpected events during conditioning and attention: a psychophysiological theory. *Psychol. Rev.* 89, 529–572. doi: 10.1037//0033-295x.89.5.529
- Grossberg, S. (1984a). “Outline of a theory of brightness, color, and form perception,” in *Trends in Mathematical Psychology*, eds E. Degreef and J. van Buggenhaut (Amsterdam: Elsevier), 59–85.
- Grossberg, S. (1984b). “Some psychophysiological and pharmacological correlates of a developmental, cognitive, and motivational theory,” in *Brain and Information: Event Related Potentials*, eds R. Karrer, J. Cohen, and P. Tueting (New York, NY: Academy of Sciences), 58–142.
- Grossberg, S. (1988). Nonlinear neural networks: principles, mechanisms, and architectures. *Neural Netw.* 1, 17–61. doi: 10.1016/j.neunet.2008.03.015
- Grossberg, S. (1994). 3-D vision and figure-ground separation by visual cortex. *Percept. Psychophys.* 55, 48–120.
- Grossberg, S. (1997). Cortical dynamics of three-dimensional figure-ground perception of two-dimensional figures. *Psychol. Rev.* 104, 618–658. doi: 10.1037//0033-295x.104.3.618
- Grossberg, S. (2000). The complementary brain: unifying brain dynamics and modularity. *Trends Cogn. Sci.* 4, 233–246. doi: 10.1016/s1364-6613(00)01464-9
- Grossberg, S. (2009a). Beta oscillations and hippocampal place cell learning during exploration of novel environments. *Hippocampus* 19, 881–885. doi: 10.1002/hipo.20602
- Grossberg, S. (2009b). Cortical and subcortical predictive dynamics and learning during perception, cognition, emotion, and action. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* 364, 1223–1234. doi: 10.1098/rstb.2008.0307
- Grossberg, S. (2013). Adaptive resonance theory: how a brain learns to consciously attend, learn, and recognize a changing world. *Neural Netw.* 37, 1–47. doi: 10.1016/j.neunet.2012.09.017
- Grossberg, S. (2017). Towards solving the hard problem of consciousness: the varieties of brain resonances and the conscious experiences that they support. *Neural Netw.* 87, 38–95. doi: 10.1016/j.neunet.2016.11.003
- Grossberg, S. (2018). Desirability, availability, credit assignment, category learning, and attention: cognitive-emotional and working memory dynamics of orbitofrontal, ventrolateral, and dorsolateral prefrontal cortices. *Brain Neurosci. Adv.* doi: 10.1177/2398212818772179
- Grossberg, S., and Gutowski, W. E. (1987). Neural dynamics of decision making under risk: affective balance and cognitive-emotional interactions. *Psychol. Rev.* 94, 300–318. doi: 10.1037/0033-295x.94.3.300
- Grossberg, S., and Kazerounian, S. (2011). Laminar cortical dynamics of conscious speech perception: a neural model of phonemic restoration using subsequent context in noise. *J. Acoust. Soc. Am.* 130, 440–460. doi: 10.1121/1.3589258
- Grossberg, S., and Kishnan, D. (2018). Neural dynamics of autistic repetitive behaviors and Fragile X syndrome: Basal ganglia movement gating and mGluR-modulated adaptively timed learning. *Front. Psychol. Psychopathol.* 9:269. doi: 10.3389/fpsyg.2018.00269
- Grossberg, S., Markowitz, J., and Cao, Y. (2011). On the road to invariant recognition: explaining tradeoff and morph properties of cells in inferotemporal cortex using multiple-scale task-sensitive attentive learning. *Neural Netw.* 24, 1036–1049. doi: 10.1016/j.neunet.2011.04.001
- Grossberg, S., and Merrill, J. W. (1992). A neural network model of adaptively timed reinforcement learning and hippocampal dynamics. *Cogn. Brain Res.* 1, 3–38. doi: 10.1016/0926-6410(92)90003-a
- Grossberg, S., and Merrill, J. W. (1996). The hippocampus and cerebellum in adaptively timed learning, recognition, and movement. *J. Cogn. Neurosci.* 8, 257–277. doi: 10.1162/jocn.1996.8.3.257
- Grossberg, S., and Mingolla, E. (1985). Neural dynamics of form perception: boundary completion, illusory figures, and neon color spreading. *Psychol. Rev.* 92, 173–211. doi: 10.1037//0033-295x.92.2.173
- Grossberg, S., and Myers, C. W. (2000). The resonant dynamics of speech perception: interword integration and duration-dependent backward effects. *Psychol. Rev.* 107, 735–767. doi: 10.1037//0033-295x.107.4.735
- Grossberg, S., and Pearson, L. (2008). Laminar cortical dynamics of cognitive and motor working memory, sequence learning and performance: toward a unified theory of how the cerebral cortex works. *Psychol. Rev.* 115, 677–732. doi: 10.1037/a0012618
- Grossberg, S., and Pilly, P. K. (2012). How entorhinal grid cells may learn multiple spatial scales from a dorsoventral gradient of cell response rates in a self-organizing map. *PLoS Comput. Biol.* 8:31002648. doi: 10.1371/journal.pcbi.1002648
- Grossberg, S., and Pilly, P. K. (2014). Coordinated learning of grid cell and place cell spatial and temporal properties: multiple scales, attention, and oscillations. *Philos. Trans. R. Soc. B* 369:20120524. doi: 10.1098/rstb.2012.0524
- Grossberg, S., and Schmajuk, N. A. (1987). Neural dynamics of attentionally-modulated Pavlovian conditioning: conditioned reinforcement, inhibition, and opponent processing. *Psychobiology* 15, 195–240.

- Grossberg, S., and Schmajuk, N. A. (1989). Neural dynamics of adaptive timing and temporal discrimination during associative learning. *Neural Netw.* 2, 79–102. doi: 10.1016/0893-6080(89)90026-9
- Grossberg, S., and Seidman, D. (2006). Neural dynamics of autistic behaviors: cognitive, emotional, and timing substrates. *Psychol. Rev.* 113, 483–525. doi: 10.1037/0033-295x.113.3.483
- Grossberg, S., Srihasam, K., and Bullock, D. (2012). Neural dynamics of saccadic and smooth pursuit eye movement coordination during visual tracking of unpredictably moving targets. *Neural Netw.* 27, 1–20. doi: 10.1016/j.neunet.2011.10.011
- Grossberg, S., Srinivasan, K., and Yazdanbakhsh, A. (2014). Binocular fusion and invariant category learning due to predictive remapping during scanning of a depthful scene with eye movements. *Front. Psychol. Percept. Sci.* 5:1457. doi: 10.3389/fpsyg.2014.01457
- Grossberg, S., and Vladusich, T. (2010). How do children learn to follow gaze, share joint attention, imitate their teachers, and use tools during social interactions? *Neural Netw.* 23, 940–965. doi: 10.1016/j.neunet.2010.07.011
- Grossberg, S., and Versace, M. (2008). Spikes, synchrony, and attentive learning by laminar thalamocortical circuits. *Brain Res.* 1218, 278–312. doi: 10.1016/j.brainres.2008.04.024
- Hafting, T., Fyhn, M., Molden, S., Moser, M. B., and Moser, E. I. (2005). Microstructure of a spatial map in the entorhinal cortex. *Nature* 436, 801–806. doi: 10.1038/nature03721
- Hargreaves, E. L., Rao, G., Lee, I., and Knierim, J. J. (2005). Major dissociation between medial and lateral entorhinal input to dorsal hippocampus. *Science* 308, 1792–1794. doi: 10.1126/science.1110449
- Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A. R., et al. (2012). Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Process. Mag.* 29, 82–97.
- Huang, T.-R., and Grossberg, S. (2010). Cortical dynamics of contextually cued attentive visual learning and search: spatial and object evidence accumulation. *Psychol. Rev.* 117, 1080–1112. doi: 10.1037/a0020664
- Ichise, T., Kano, M., Hashimoto, K., Yangihara, D., Nakao, K., Shigemoto, R., et al. (2000). mGluR1 in cerebellar Purkinje cells essential for long-term depression, synapse elimination, and motor coordination. *Science* 288, 1832–1835. doi: 10.1126/science.288.5472.1832
- Jones, D., Farrand, P., Stuart, G., and Morris, N. (1995). The functional equivalence of verbal and spatial memory in serial short-term memory. *J. Exp. Psychol. Learn. Mem. Cogn.* 21, 1008–1018. doi: 10.1037//0278-7393.21.4.1008
- Kazerounian, S., and Grossberg, S. (2014). Real-time learning of predictive recognition categories that chunk sequences of items stored in working memory. *Front. Psychol. Lang. Sci.* 5:1053. doi: 10.3389/fpsyg.2014.01053
- Keene, C. S., Bladon, J., McKenzie, S., Liu, C. D., O'Keefe, J., and Eichenbaum, H. (2016). Complementary functional organization of neuronal activity patterns in the perirhinal, lateral entorhinal, and medial entorhinal cortices. *J. Neurosci.* 36, 3660–3675. doi: 10.1523/JNEUROSCI.4368-15.2016
- Kentros, C. G., Agnietri, N. T., Streater, S., Hawkins, R. D., and Kandel, E. R. (2004). Increased attention to spatial context increases both place field stability and spatial memory. *Neuron* 42, 283–295. doi: 10.1016/s0896-6273(04)00192-8
- Kentros, C. G., Hargreaves, E., Hawkins, R. D., Kandel, E. R., Shapiro, M., and Muller, R. V. (1998). Abolition of long-term stability of new hippocampal place cell maps by NMDA receptor blockade. *Science* 280, 2121–2126. doi: 10.1126/science.280.5372.2121
- Kerr, K. M., Agster, K. L., Furtak, S. C., and Burwell, R. D. (2007). Functional neuroanatomy of the parahippocampal region: the lateral and medial entorhinal areas. *Hippocampus* 17, 697–708. doi: 10.1002/hipo.20315
- Kojima, S., and Goldman-Rakic, P. S. (1984). Functional analysis of spatially discriminative neurons in prefrontal cortex of rhesus monkey. *Brain Res.* 291, 229–240. doi: 10.1016/0006-8993(84)91255-1
- Langston, R. F., Ainge, J. A., Couey, J. J., Canto, C. B., Bjerknes, T. L., Witter, M. P., et al. (2010). Development of the spatial representation system in the rat. *Science* 328, 1576–1580. doi: 10.1126/science.1188210
- LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521, 436–444. doi: 10.1038/nature14539
- Leonard, B., and McNaughton, B. L. (1990). “Spatial representation in the rat: Conceptual, behavioral, and neurophysiological perspectives,” in *Neurobiology of Comparative Cognition*, eds R. P. Kesner and D. S. Olton (Hillsdale, NJ: Lawrence Erlbaum Associates), 363–422.
- LeVine, S. (2017). *Artificial Intelligence Pioneer Says We Need to Start Over*. Arlington, VA: Axios.
- López-López, J. R., Shacklock, P. S., Balke, C. W., and Wier, W. G. (1995). Local calcium transients triggered by single L-type calcium channel currents in cardiac cells. *Science* 268, 1042–1045. doi: 10.1126/science.7754383
- Luck, S. J., Chelazzi, L., Hillyard, S. A., and Desimone, R. (1997). Neural mechanisms of spatial selective attention in areas V1, V2, and V4 of macaque visual cortex. *J. Neurophysiol.* 77, 24–42. doi: 10.1152/jn.1997.77.1.24
- Lundqvist, M., Herman, P., Warden, M. R., Brincat, S. L., and Miller, E. K. (2018). Gamma and beta bursts during working memory readout suggest roles in its volitional control. *Nat. Commun.* 9:394. doi: 10.1038/s41467-017-02791-8
- Maunsell, J. H., Nealey, T. A., and DePriest, D. D. (1990). Magnocellular and parvocellular contributions to responses in the middle temporal visual area (MT) of the macaque monkey. *J. Neurosci.* 10, 3323–3334. doi: 10.1523/jneurosci.10-10-03323.1990
- Maunsell, J. H. R., and Newsome, W. T. (1987). Visual processing in monkey extrastriate cortex. *Annu. Rev. Neurosci.* 10, 363–401. doi: 10.1146/annurev.neuro.10.1.363
- McNab, F., and Klingberg, T. (2008). Prefrontal cortex and basal ganglia control access to working memory. *Nat. Neurosci.* 11, 103–107. doi: 10.1038/nn2024
- Mhatre, H., Gorchetnikov, A., and Grossberg, S. (2012). Grid cell hexagonal patterns formed by fast self-organized learning within entorhinal cortex. *Hippocampus* 22, 320–334. doi: 10.1002/hipo.20901
- Mingolla, E., Ross, W., and Grossberg, S. (1999). A neural network for enhancing boundaries and surfaces in synthetic aperture radar images. *Neural Netw.* 12, 499–511. doi: 10.1016/s0893-6080(98)00144-0
- Mishkin, M. (1982). A memory system in the monkey. *Philos. Trans. R. Soc. Lond. B* 298, 85–95. doi: 10.1098/rstb.1982.0074
- Mishkin, M., Ungerleider, L. G., and Macko, K. A. (1983). Object vision and spatial vision: two cortical pathways. *Trends Neurosci.* 6, 414–417. doi: 10.1016/0166-2236(83)90190-x
- Miyata, M., Finch, E. A., Khiroug, L., Hashimoto, K., Hayasaka, S., Oda, S. I., et al. (2000). Local calcium release in dendritic spines required for long-term synaptic depression. *Neuron* 28, 233–244. doi: 10.1016/s0896-6273(00)00099-4
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., et al. (2013). Playing atari with deep reinforcement learning. *arXiv*
- Morris, R. G., and Frey, U. (1997). Hippocampal synaptic plasticity: role in spatial learning or the automatic recording of attended experience? *Proc. R. Soc. B.* 1360, 1469–1503.
- Mumford, D. (1992). On the computational architecture of the neocortex. II, the role of corticocortical loops. *Biol. Cyber.* 66, 241–251. doi: 10.1007/bf00198477
- Näätänen, R. (1982). Processing negativity: an evoked-potential reflection. *Psychol. Bull.* 92, 605–640. doi: 10.1037//0033-2909.92.3.605
- Näätänen, R., Simpson, M., and Loveless, N. E. (1982). Stimulus deviance and evoked potentials. *Biol. Psychiatry* 14, 53–98. doi: 10.1016/0301-0511(82)90017-5
- O'Keefe, J., and Nadel, L. (1978). *The Hippocampus as a Cognitive Map*. Oxford: Clarendon Press.
- Pack, C., Grossberg, S., and Mingolla, E. (2001). A neural model of smooth pursuit control and motion perception by cortical area MST. *J. Cogn. Neurosci.* 13, 102–120. doi: 10.1162/089892901564207
- Page, M. P., and Norris, D. (1998). The primacy model: a new model of immediate serial recall. *Psychol. Rev.* 105, 761–781. doi: 10.1037//0033-295x.105.4.761
- Piaget, J. (1945). *La Formation du Symbole Chez L'enfant*. Paris: Delachaux Niestle.
- Piaget, J. (1951). *Play, Dreams and Imitation in Childhood*. London: Routledge and Kegan Paul.
- Piaget, J. (1952). *The Origins of Intelligence in Children*. New York, NY: International Universities Press.
- Pilly, P. K., and Grossberg, S. (2012). How do spatial learning and memory occur in the brain? Coordinated learning of entorhinal grid cells and hippocampal place cells. *J. Cogn. Neurosci.* 24, 1031–1054. doi: 10.1162/jocn_a_00200
- Pilly, P. K., and Grossberg, S. (2013). Spiking neurons in a hierarchical self-organizing map model can learn to develop spatial and temporal properties

- of entorhinal grid cells and hippocampal place cells. *PLoS One* 8:e60599. doi: 10.1371/journal.pone.0060599
- Pilly, P. K., and Grossberg, S. (2014). How does the modular organization of entorhinal grid cells develop? *Front. Hum. Neurosci.* 8:337. doi: 10.3389/fnhum.2014.0037
- Polimeni, J. R., Balasubramanian, M., and Schwartz, E. L. (2006). Multi-area visuotopic map complexes in macaque striate and extra-striate cortex. *Vis. Res.* 46, 3336–3359. doi: 10.1016/j.visres.2006.03.006
- Prescott, T. J. (2008). Action selection. *Scholarpedia* 3:2705.
- Pribe, C., Grossberg, S., and Cohen, M. A. (1997). Neural control of interlimb oscillations, II: biped and quadruped gaits and bifurcations. *Biol. Cyber.* 77, 141–152. doi: 10.1007/s004220050375
- Rao, R. P., and Ballard, D. H. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive field effects. *Nat. Neurosci.* 2, 79–87. doi: 10.1038/4580
- Rizzolatti, G. (2005). The mirror neuron system and its function in humans. *Anat. Embryol. (Berl)* 210, 419–421. doi: 10.1007/s00429-005-0039-z
- Rizzolatti, G., and Craighero, L. (2004). The mirror-neuron system. *Annu. Rev. Neurosci.* 27, 169–192. doi: 10.1146/annurev.neuro.27.070203.144230
- Roelfsema, P. R., Lamme, V. A., and Spekreijse, H. (1998). Object-based attention in the primary visual cortex of the macaque monkey. *Nature* 395, 376–381. doi: 10.1038/26475
- Sams, M., Paavilainen, P., Alho, K., and Näätänen, R. (1985). Auditory frequency discrimination and event-related potentials. *Clin. Neurophysiol.* 62, 437–448. doi: 10.1016/0168-5597(85)90054-1
- Schiller, P. H., Logothetis, N. K., and Charles, E. R. (1990). Functions of the colour-opponent and broad-band channels of the visual system. *Nature* 343, 68–70. doi: 10.1038/343068a0
- Schwartz, E. L. (1984). “Spatial mapping and spatial vision in primate striate and mferotemporal cortex,” in *Sensory Experience, Adaptation and Perception*, eds L. Spillmann and B. R. Woolen (Hillsdale, NJ: Erlbaum), 73–104.
- Schwartz, E. L., Greve, D. N., and Bonmassar, G. (1995). Space-variant active vision: definition, overview, and examples. *Neural Netw.* 8, 1297–1308. doi: 10.1016/0893-6080(95)00092-5
- Sillito, A. M., Jones, H. E., Gerstein, G. L., and West, D. C. (1994). Feature-linked synchronization of thalamic relay cell firing induced by feedback from the visual cortex. *Nature* 369, 479–482. doi: 10.1038/369479a0
- Silver, M. R., Grossberg, S., Bullock, D., Histed, M. H., and Miller, E. K. (2011). A neural model of sequential movement planning and control of eye movements: Item-order-rank working memory and saccade selection by the supplementary eye fields. *Neural Netw.* 26, 29–58. doi: 10.1016/j.neunet.2011.10.004
- Snyder, L. H., Batista, A. P., and Andersen, R. A. (1997). Coding of intention in the posterior parietal cortex. *Nature* 386, 167–170. doi: 10.1038/386167a0
- Snyder, L. H., Batista, A. P., and Andersen, R. A. (1998). Change in motor plan, without a change in the spatial locus of attention, modulates activity in posterior parietal cortex. *J. Neurophysiol.* 79, 2814–2819. doi: 10.1152/jn.1998.79.5.2814
- Suzuki, M., and Gottlieb, J. (2013). Distinct neural mechanisms of distractor suppression in the frontal and parietal lobe. *Nat. Neurosci.* 16, 98–104. doi: 10.1038/nn.3282
- Takechi, H., Eilers, J., and Konnerth, A. (1998). A new class of synaptic response involving calcium release in dendritic spines. *Nature* 396, 757–760. doi: 10.1038/25547
- Tanaka, K. (1996). Inferotemporal cortex and object vision. *Annu. Rev. Neurosci.* 19, 109–139. doi: 10.1146/annurev.neuro.19.1.109
- Tanaka, K. (1997). Mechanisms of visual object recognition: monkey and human studies. *Curr. Opin. Neurobiol.* 7, 523–529. doi: 10.1016/s0959-4388(97)80032-3
- Tanaka, K. (2000). Mechanisms of visual object recognition studied in monkeys. *Spat. Vis.* 13, 147–163. doi: 10.1163/156856800741171
- Tootell, R. B., Silverman, M. S., Switkes, E., and DeValois, R. L. (1982). Deoxyglucose analysis of retinotopic organization in primate striate cortex. *Science* 218, 902–904. doi: 10.1126/science.7134981
- Tsushima, Y., Seitz, A. R., and Watanabe, T. (2008). Task-irrelevant learning occurs only when the irrelevant feature is weak. *Curr. Biol.* 18, R516–R517.
- Tyler, C. W., and Kontsevich, L. L. (1995). Mechanisms of stereoscopic processing: stereoattention and surface perception in depth reconstruction. *Perception* 24, 127–153. doi: 10.1068/p240127
- Ungerleider, L. G., and Mishkin, M. (1982). “Two cortical visual systems: Separation of appearance and location of objects,” in *Analysis of Visual Behavior*, eds D. L. Ingle, M. A. Goodale, and R. J. W. Mansfield (Cambridge, MA: The MIT Press), 549–586.
- van Strien, N. M., Cappaert, N. L., and Witter, M. P. (2009). The anatomy of memory: an interactive overview of the parahippocampal-hippocampal network. *Nat. Rev. Neurosci.* 10, 272–282. doi: 10.1038/nrn2614
- Wallace, R. S., Ong, P. W., Bederson, B. B., and Schwartz, E. L. (1984). Space variant image processing. *Int. J. Comp. Vis.* 13, 71–90.
- Warden, M. R., and Miller, E. K. (2010). Task-dependent changes in short-term memory in the prefrontal cortex. *J. Neurosci.* 30, 15801–15810. doi: 10.1523/JNEUROSCI.1569-10.2010
- Warren, W. H., Rothman, D. B., Schnapp, B. H., and Ericson, J. D. (2017). Wormholes in virtual space: from cognitive maps to cognitive graphs. *Cognition* 166, 152–163. doi: 10.1016/j.cognition.2017.05.020
- Webster, M. J., Bachevalier, J., and Ungerleider, L. G. (1994). Connections of inferior temporal areas TEO and TE with parietal and frontal cortex in macaque monkeys. *Cereb. Cortex* 4, 470–483. doi: 10.1093/cercor/4.5.470
- Wills, T. J., Cacucci, F., Burgess, N., and O’Keefe, J. (2010). Development of the hippocampal cognitive map in preweanling rats. *Science* 328, 1573–1576. doi: 10.1126/science.1188224
- Yao, Y., Choi, J., and Parker, I. (1995). Quantal putts of intracellular Ca²⁺ evoked by inositol trisphosphate in *Xenopus oocytes*. *J. Physiol.* 482, 533–553. doi: 10.1113/jphysiol.1995.sp020538
- Zalama, E., Gaudiano, P., and Coronado, J. L. (1995). A real-time, unsupervised neural network for the low-level control of a mobile robot in a nonstationary environment. *Neural Netw.* 8, 103–123. doi: 10.1016/0893-6080(94)00063-r

Conflict of Interest Statement: The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Grossberg. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



The Energy Homeostasis Principle: Neuronal Energy Regulation Drives Local Network Dynamics Generating Behavior

Rodrigo C. Vergara¹, Sebastián Jaramillo-Riveri², Alejandro Luarte³,
Cristóbal Moënné-Loccoz^{4,5}, Rómulo Fuentes⁴, Andrés Couve³ and Pedro E. Maldonado^{1*}

¹ Neurosystems Laboratory, Faculty of Medicine, Biomedical Neuroscience Institute, Universidad de Chile, Santiago, Chile, ² School of Biological Sciences, Institute of Cell Biology, University of Edinburgh, Edinburgh, United Kingdom, ³ Cellular and Molecular Neurobiology Laboratory, Faculty of Medicine, Biomedical Neuroscience Institute, Universidad de Chile, Santiago, Chile, ⁴ Motor Control Laboratory, Faculty of Medicine, Biomedical Neuroscience Institute, Universidad de Chile, Santiago, Chile, ⁵ Department of Health Sciences, Faculty of Medicine, Pontificia Universidad Católica de Chile, Santiago, Chile

OPEN ACCESS

Edited by:

Mario Senden,
Maastricht University, Netherlands

Reviewed by:

Yuanyuan Mi,
Chongqing University, China
Zhaofei Yu,
Peking University, China

*Correspondence:

Pedro E. Maldonado
pedro@uchile.cl

Received: 07 January 2019

Accepted: 01 July 2019

Published: 23 July 2019

Citation:

Vergara RC, Jaramillo-Riveri S,
Luarte A, Moënné-Loccoz C,
Fuentes R, Couve A and
Maldonado PE (2019) The Energy
Homeostasis Principle: Neuronal
Energy Regulation Drives Local
Network Dynamics Generating
Behavior.
Front. Comput. Neurosci. 13:49.
doi: 10.3389/fncom.2019.00049

A major goal of neuroscience is understanding how neurons arrange themselves into neural networks that result in behavior. Most theoretical and experimental efforts have focused on a top-down approach which seeks to identify neuronal correlates of behaviors. This has been accomplished by effectively mapping specific behaviors to distinct neural patterns, or by creating computational models that produce a desired behavioral outcome. Nonetheless, these approaches have only implicitly considered the fact that neural tissue, like any other physical system, is subjected to several restrictions and boundaries of operations. Here, we proposed a new, bottom-up conceptual paradigm: The Energy Homeostasis Principle, where the balance between energy income, expenditure, and availability are the key parameters in determining the dynamics of neuronal phenomena found from molecular to behavioral levels. Neurons display high energy consumption relative to other cells, with metabolic consumption of the brain representing 20% of the whole-body oxygen uptake, contrasting with this organ representing only 2% of the body weight. Also, neurons have specialized surrounding tissue providing the necessary energy which, in the case of the brain, is provided by astrocytes. Moreover, and unlike other cell types with high energy demands such as muscle cells, neurons have strict aerobic metabolism. These facts indicate that neurons are highly sensitive to energy limitations, with Gibb's free energy dictating the direction of all cellular metabolic processes. From this activity, the largest energy, by far, is expended by action potentials and post-synaptic potentials; therefore, plasticity can be reinterpreted in terms of their energy context. Consequently, neurons, through their synapses, impose energy demands over post-synaptic neurons in a close loop-manner, modulating the dynamics of local circuits. Subsequently, the energy dynamics end up impacting the homeostatic mechanisms of neuronal networks. Furthermore, local energy management also emerges as a neural population property, where most of the energy expenses are triggered by sensory or other modulatory inputs. Local

energy management in neurons may be sufficient to explain the emergence of behavior, enabling the assessment of which properties arise in neural circuits and how. Essentially, the proposal of the Energy Homeostasis Principle is also readily testable for simple neuronal networks.

Keywords: homeostasis, energy, neuronal networks, behavior, emergent properties

INTRODUCTION

Throughout evolution, the development of the nervous system has enabled animals with the capacity to manifest ever-growing complex behavior, which has helped them survive in a changing environment. Understanding how neurons arrange themselves into neural networks that work at producing different behaviors has always been a major goal of neuroscience. Various conceptual frameworks have aimed to explain how behavior emerges from neuronal activity. Arguably, the most relevant is the Neuron Doctrine, proposed by Santiago Ramón y Cajal and further developed by Heinrich Waldeyer-Hartz and Horace Barlow (Barlow, 1972; Bock, 2013). Since then, the same logic has spread into coding paradigms (Lettvin et al., 1959; Fairhall, 2014; Yuste, 2015), especially in information processing frameworks (Fodor, 1983; Friston, 2002; Robbins, 2010; Lorenz et al., 2011), and has been scaled from neurons up to neural networks (Yuste, 2015). A common and key element of these conceptual approaches has been to find neuronal correlates of behaviors, effectively associating specific behaviors with distinct neural patterns. This top-down approach (using behavior as a reference to be mapped into neuronal circuits) has been very successful in providing single-unit or network models that can implement the observed behaviors, yet simultaneously, may make difficult the capture of the emergence of behavior, which is by-large a bottom-up phenomenon. This methodological approach also limits our capacity of predicting the boundaries of the capabilities or the spectrum of behaviors of a given system, because we map or associate only those behaviors that have been well-characterized. More importantly, all theoretical approaches, to our knowledge, have only implicitly addressed the fact that neural tissue, like any other physical system, is subjected to several restrictions and boundaries of operations.

Cells use energy to stay alive and at the same time, maintain some reserves to respond and adapt to dynamic situations, maintaining their *homeostasis*. For neurons, energy availability would be further important, as their energy expenses are high, as compared to other somatic cells (Attwell and Laughlin, 2001; Shulman et al., 2004). Indeed, the metabolic consumption of the brain, which represents 20% of whole-body oxygen consumption, contrasts with the neural tissue representing only 2% of whole body weight (Shulman et al., 2004). Interestingly, the total brain energy consumption increases proportionally with the number of neurons among different species, including humans (Herculano-Houzel, 2011), and the total energy expenditure associated to a neuron during the signaling and resting states is constant in different mammalian species (Hyder et al., 2013). Thus, neurons seem

to present a highly specialized system for managing their energy demands.

Several evidences demonstrate that it is reasonable to assume a constant value for energy availability for neurons over the long term (energetic homeostasis). For instance, cultured neurons exhibit a steady value for free adenosine triphosphate (ATP) in basal conditions, which transiently decrease during the induction of glutamatergic synaptic activity through various energy challenges (Marcaida et al., 1995, 1997; Rangaraju et al., 2014; Lange et al., 2015). This tight energy management suggests a relevant role for neuronal energy homeostasis on neuronal and network functional properties.

Here, we propose a new bottom-up conceptual paradigm for neuronal networks: The Energy Homeostasis Principle. Under this principle, the condition of maintaining neuronal homeostasis triggers synaptic changes in the individual but connected neurons, resulting in the local energy balance scaling up to a network property. This conceptual framework supposes that energy management might be critical in determining plasticity, network functional connectivity, and ultimately behavior.

CELLULAR HOMEOSTASIS AND GIBBS FREE ENERGY

In this article, we propose that behavior may raise as an emergent property rooted in energy requirement of neurons, thus, we would like to start from the level of biochemistry and metabolism. As such, we will begin with the fact that cells are dynamic molecular machines that require the nutrient intake to stay alive. Many biological processes are thermodynamically unfavorable, and through metabolism, cells draw energy from nutrients, and generate metabolic resources necessary to drive their cellular activities (Hofmeyr and Cornish-Bowden, 2000) (for a schematic, see **Figure 1A**). Cellular homeostasis can be defined as a state where the production and consumption of metabolic resources balance each-other, and thus their concentration is constant in time. For our specific context, balancing the intake and consumption of metabolic resources will unavoidably have a global impact on the cellular processes. The network of metabolic processes is large and complex, limiting, to some extent, our capacity to predict cellular behavior using basic principles. Nonetheless, biochemical reactions must be consistent with the laws of thermodynamics.

Thermodynamics can help us understand how a system evolves in time through the comparison of the thermodynamic potential between an initial and final state. For processes at a constant temperature and pressure, the thermodynamic

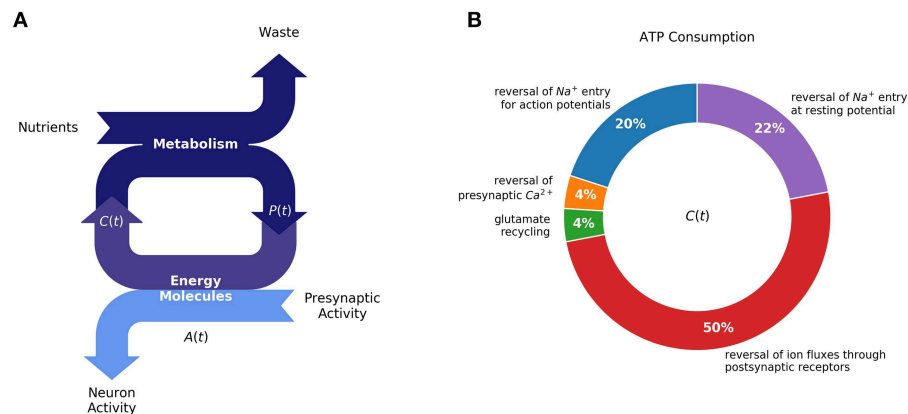


FIGURE 1 | Homeostasis requires a balance between ATP production (metabolism) and ATP consumption (synaptic activity): **(A)** Production of energy molecules by metabolism supports neuron activity, in addition to cell maintenance processes, most notably by ATP $[A(t)]$. Neuronal homeostasis depends on a balance between production and consumption of high-energy molecules. **(B)** Synaptic activity has been estimated to amount for half of total ATP consumption [diagram redrawn from (Harris et al., 2012)]. For cellular homeostasis to be achieved, neurons must regulate their activity and metabolism in response to changing external perturbations. We propose that regulatory mechanisms, responsible for changes synaptic plasticity, reflect the requirement for maintaining a constant level of energy resources available for neurons to function.

potential is given by Gibbs Free Energy (G). This thermodynamic potential will dictate a directional bias of chemical reactions. The Gibbs Free Energy—supporting cellular processes—is provided with a finite amount of metabolic resources. Thus, there is a trade-off between the potential for metabolic work and metabolic expenses, which, we propose, may explain some well-established phenomenology of how cells respond to external perturbations. Additionally, the change in the Gibbs Free Energy (ΔG) and the rate associated with chemical transformations are related (Crooks, 1999). To illustrate the relation between thermodynamics and kinetics, for a reversible reaction $X \rightleftharpoons Y$, the following relation constrains kinetic rates:

$$\frac{\text{rate}(X \rightarrow Y)}{\text{rate}(X \leftarrow Y)} = e^{G(X) - G(Y)} \quad (1)$$

In simple terms, the difference $\Delta G\{X \rightleftharpoons Y\}$ can be thought of as a “directional bias,” indicating how favorable one direction is over the other. In more detail, the Gibbs Free Energy is divided into two components, Enthalpy (H) and Entropy (S):

$$G(X) = H(X) - TS(X) \quad (2)$$

Where, T is the absolute temperature (Silbey et al., 2004). In the context of chemical transformations, Enthalpy is a measure of the energy required to form a given substance, disregarding interactions with other molecules; whereas Entropy can be interpreted as a correction accounting for all possible combinations by which molecules can react (Danos and Oury, 2013). Given the combinatorial nature of entropy, it can also be interpreted as a measure of disorder or information, but certain care must be taken for this interpretation to have physical meaning (Jaynes, 1965). We wish to recognize that the direct application of thermodynamics to biology has many challenges, particularly in describing macro-molecular processes (Cannon,

2014; Cannon and Baker, 2017; Ouldrige, 2018), combining large systems of reactions (e.g., kinetic parameters may be required), and accounting for fluctuations from average behavior (Marland and England, 2018).

A long-standing observation in biology, rooted in thermodynamic laws is that for cells to function, they must couple unfavorable reactions ($\Delta G > 0$) with more favorable ones ($\Delta G < 0$). Common examples of unfavorable processes are the synthesis of macromolecules, and the maintenance of membrane potential; which are coupled with the hydrolysis of ATP, and GTP providing more favorable ΔG (Nicholls, 2013). In turn, ATP, GTP, and monomers for macro-molecules are synthesized from nutrients through metabolism (Nicholls, 2013). For instance, the maximum free energy provided by ATP hydrolysis is related to the concentration of ATP, ADP, and phosphate.

$$\Delta G(ATP \rightarrow ADP + Pi) = \Delta G^\circ + RT(\log[ADP] + \log[Pi] - \log[ATP]) \quad (3)$$

Where, ΔG° is the standard free energy, and \log , the natural logarithm. For generality, we will call hereafter “energy resources” the set of reactants that allow cells to maintain unfavorable reactions in the direction conducive to cellular functioning and survival. We wish to emphasize that balancing the internal production and consumption of metabolic resources by different reactions is critical, given that metabolic resources are finite and shared by many cellular processes. Thus, cells must manage their internal production and consumption of metabolic resources to stay alive and remain functional, which may be of special consequence to cellular activities with high energy demands, such as synaptic activity in neurons. Given that neurons are active most of the time, it is reasonable to expect that current and future disposal of energy resources is privileged, which may be reflected in the regulatory mechanisms responsible for synaptic plastic changes. In the following section,

we will explain how current evidence regarding neuron plasticity appears to support a relatively simple rule: maintain the levels of energy disposal constant, by reducing the consumption of energy resources (e.g., reducing discharge rate, post-synaptic potential), or by increasing high-energy molecule production (e.g., mitochondria and interactions with glia).

ENERGY MANAGEMENT OF BRAIN NEURONS

Neurons are the paramount example of energy expenditure for their function and survival. This situation is reflected in their large metabolic rates and by the comparatively higher sensibility of brain tissues to oxygen and glucose deprivation (Ames, 2000). Reactions controlling the conversion of nutrients into available cytosolic levels of ATP are important to generate the potential metabolic work that is available to a neuron at any given time. During normal conditions, the primary energy substrate in the brain for neurons is blood-derived glucose; however, when at elevated levels in the blood, ketone bodies and lactate can be used as energy sources as well (Magistretti and Allaman, 2018). The glycolytic pathway is the first step to glucose processing, where two pyruvates and two ATPs are generated from one molecule of glucose. In addition, the pyruvate could either be reduced to lactate or enter the Krebs cycle to produce metabolic reducing intermediates that will generate nearly 29 additional ATP molecules per glucose (through oxidative phosphorylation in the mitochondria). Although neurons and astrocytes are capable of glucose uptake and performing both glycolysis and the Krebs cycle, accumulated evidence supports the hypothesis that neurons may “outsource” glycolytic activity to astrocytes under activity conditions (Weber and Barros, 2015). In addition, the central nervous system is provided with small glycogen reserves, which are predominantly present in astrocytes (Brown and Ransom, 2007), but also found in neurons (Saez et al., 2014). In any case, the lactate derived from glycogen break-down may also provide ATP to the neurons under ischemic or sustained electric activity conditions (Brown and Ransom, 2007).

ATP sources change dynamically with neuronal activity and several mechanisms account for this fine-tuning response. First, neuronal mitochondria are capable of raising ATP synthesis in response to increased synaptic stimuli (Jekabsons and Nicholls, 2004; Connolly et al., 2014; Rangaraju et al., 2014; Toloe et al., 2014; Lange et al., 2015). Although the molecular mediators for this activation are not completely elucidated, the increase of the respiratory rate of an isolated mitochondria correlates well with the ADP concentration (Brown, 1992), and neuronal mitochondrial function has been satisfactorily modeled considering the changes in ATP and ADP levels (Berndt et al., 2015). As an alternative mechanism, it has been reported that operating on milder stimulation conditions, the activity of Na⁺-pump rapidly induces ATP synthesis of the mitochondria, in response to neuronal activity independent from changes in the adenosine nucleotides (Baeza-Lehnert et al., 2018). Second, neuronal activity is known to elicit local increases in blood

flow (neurovascular coupling), glucose uptake, and oxygen consumption (Sokoloff, 2008). Coherently, glucose uptake and glycolytic rate of astrocytes are further increased in response to the activity of excitatory neurons, potentially as a consequence of the local rise of glutamate, ammonium (NH₄), nitric oxide (NO), and importantly, K⁺ (Magistretti and Allaman, 2018). As such, an increased glycolytic rate on astrocytes leads to lactate accumulation that is shuttled into neurons which generate ATP through oxidative phosphorylation. Thus, in CNS neurons, different neuronal and non-neuronal ATP sources work “on demand,” depending on the local levels of synaptic activity.

What Is ATP Used for in Neurons?

Neurons are perhaps the largest eukaryotic cell in nature, their surface may be up to 10,000 times larger than an average cell (Horton and Ehlers, 2003). The large size of neurons supposes that structural processes, such as protein and lipid synthesis or the traffic of subcellular organelles, should be sustained by high levels of ATP synthesis. In addition to this fact, energy consumption during signaling is far more important. Indeed, it has been estimated that nearly 75% of the gray-matter energy budget is used during signaling; a number that is coherent with the decrease of energy consumption, observed under anesthesia, and is estimated to be around 20% of the total energy budget (Attwell and Laughlin, 2001; Harris et al., 2012).

Most of the neuron's energy budget during signaling is used to restore ion gradients across the plasma membrane, mediated by the action of different ATP-dependent pumps. For example, assuming an average firing rate of 4 Hz, a presynaptic neuron's ATP is mostly used for restoring the Na⁺ gradient due to action potentials, and to sustain the resting potential (22% and 20% of energy consumption, respectively). Meanwhile, at the post-synaptic neuron, ATP is primarily used to extrude ions participating in post-synaptic currents—about 50% of the energy consumption (Harris et al., 2012). More detailed descriptions of the neuron energy budget is provided in **Figure 1B**.

Neuron's ATP Availability Is Tightly Regulated

All cellular organizations require a minimum amount of ATP for survival. It is well-known that when ATP levels decrease below a certain threshold for different eukaryotic cells, apoptosis or necrosis is induced (Eguchi et al., 1997). Nevertheless, determining the maximum and minimum thresholds of a cell's ATP requirement for not only to survive but to realize a specialized function, is less apparent. In any case, this feature must be necessarily shaped by evolutionary adaptations of cells to their specific tissue environment. It is not completely clear how a neuron's ATP levels, during rest and upon activity, may impact its structure and function. Interestingly, by computational and mathematical modeling, it has been proposed that a compromise among energy consumption and information processing capacity has shaped the fundamental features of neuronal structure and physiology, including neuronal body size, ion channel density, and the size and frequency of synaptic inputs (Sengupta et al., 2013). For example, a larger neuronal body has a better capacity

to discriminate and respond to different synaptic inputs (coding capacity), but at the cost of higher energy consumption. On the other hand, with a fixed size for the soma, the ion channel density required to obtain maximum energy efficiency is at a lower value than the density needed to maximize the coding capacity. Similarly, although small synaptic inputs at low frequencies are energetically more efficient, better coding capacity arises with larger inputs and rates. These energy constraints may have introduced important consequences during cellular evolution, such that neurons with similar shape and function may harbor similar metabolic features, even across different species.

Remarkably, it has been found that energy consumption of neurons, across the brains of varying species, is constant (Herculano-Houzel, 2011). This result supposes a critical restriction for the function of neuronal networks and their coding properties. For example, sparse coding i.e., brain computations that emerge from the increased firing rate of a few neurons during a task, has been proposed as a mechanistic solution to the limited energy availability for brain neurons (Attwell and Laughlin, 2001; Laughlin, 2001; Lennie, 2003; Weber and Barros, 2015). Thus, it is also possible that variables such as the ATP cytosolic concentration may have been finely tuned during evolution to allow for the emergence of fundamental properties, including some forms of synaptic plasticity.

Accumulating evidence supports that neurons, in time, harbor a narrow window of ATP cytosolic concentration availability [$A(t)$]. Despite not having dynamic measurements with absolute values of $A(t)$, different experimental approaches on cultured neurons show that this variable tends to remain constant at resting conditions and after momentary synaptic challenges. Accordingly, 60 min of different sorts of glutamatergic stimulation leads to a nearly 5-fold decrease of $A(t)$ (Marcaida et al., 1995, 1997), but when a brief glutamatergic or electric stimulation is applied, only a transient and reversible decrease on ATP levels occurs and the $A(t)$ is subsequently restored to basal levels (Rangaraju et al., 2014; Lange et al., 2015).

Tight management of $A(t)$ also operates on axonal compartments with important functional consequences. For instance, isolated axons from the optical nerve, under low glucose conditions, demonstrate a pronounced decay of ATP levels during high-frequency stimulation (50–100 Hz) (Trevisiol et al., 2017). Interestingly, compound action potentials (CAPs), generated by those stimulated axons, are reduced to the same extent and in high coincidence as the $A(t)$, suggesting that electric activity depends on $A(t)$ (Trevisiol et al., 2017). In addition, isolated axons exhibit a constant value for $A(t)$, which immediately and steeply decays after the inhibition of glycolysis and oxidative phosphorylation, in concomitance with CAPs. However, when inhibitors are washed out, both $A(t)$ and CAPs return to basal levels, further supporting that the system tends to reach a constant value for $A(t)$. The tendency of the system to set a constant value for $A(t)$ is also manifest in conditions where expenditures are highly reduced. For example, $A(t)$ remains constant on pre-synaptic terminals of cultured hippocampal neurons, despite the inhibition of action potential firing due to incubation with the Na^+ channel blocker Tetrodotoxin (TTX) (Rangaraju et al., 2014). Conversely, the same study showed

that electrical stimulation of 10 Hz by 1 min, concomitantly evokes ATP synthesis on pre-synaptic terminals, restoring $A(t)$ to basal levels (Rangaraju et al., 2014). From now on, we will call the basal value of $A(t)$ as the homeostatic availability of ATP (A_H).

Mechanisms accounting for the intrinsic control of A_H in neurons are less explored than in other cells. In the short term, there is a direct and fast effect of ATP molecules and their hydrolysis products, such as AMP/ADP, over the activity of different metabolic enzymes and ion channels. Indeed, neurons are largely known for being extremely, even disproportionately, sensitive to decreases in ATP sources, leading to a fast and significant inhibition of electrical activity (Ames, 2000). For example, ATP-sensitive K^+ channels open during decreased ATP levels, hyperpolarizing the neuron to reduce endocytosis and the opening of voltage-sensitive Na^+ channels, thus preventing the ATP expenditure associated to both processes (Ben-Ari et al., 1990). On the other hand, it has been elegantly shown that action potential firing on pre-synaptic terminals' gate activity-driven ATP production is also required to allow proper synaptic transmission (Rangaraju et al., 2014). This close dependency of ATP levels to synaptic functioning has suggested that the affinity constant for ATP (e.g., K_m) of different pre-synaptic enzymes, might be close to certain resting ATP levels (Rangaraju et al., 2014). It is tempting to speculate that the fine-tuning of the affinity constant from key enzymes might be a broader phenomenon in neurons. In addition, it is known that calcium entry, which is transiently modified by electrical activity, is capable of orchestrating changes in ATP production. For example, synaptic stimulation with brief NMDA pulses, not only lead to pronounced increases of cytosolic calcium levels, but also of the mitochondrial matrix, whose ATP producing enzymes are known to be stimulated by calcium increases (Tarasov et al., 2012; Lange et al., 2015). Indeed, transient increases of calcium levels are thought to be a sort of metabolic alarm which prepares cells to confront high energy demands by increasing ATP production by the mitochondria (Bhosale et al., 2015).

As a complementary mechanism, changes in the ATP and AMP ratio gate the activity of other metabolic sensors which, in turn, induce a specific signaling cascade for short and long-term adaptations of neuronal functions. For example, all known eukaryotic cells, including neurons, harbor energy sensors, such as AMP-activated protein kinase (AMPK), which tend to restore ATP concentration by decreasing anabolic and/or energy consuming processes, while increasing energy production through catabolism post-energy challenges (Potter et al., 2010; Hardie, 2011; Hardie et al., 2012). AMPK is a highly evolutionary-conserved serine/threonine kinase enzyme that is activated either by diminished cellular energy (high AMP/ATP ratio) and/or through increased calcium (Hardie et al., 2012). Recent evidence shows that in dorsal root ganglion neurons—which express the transient receptor potential ankyrin 1 (TRPA1) channel for thermal and pain transduction—the AMPK activation results in a fast, down-regulation of membrane-associated TRPA1 and its channel activity within minutes, which is consistent with lowering energy expenditure by diminishing post-synaptic currents (Wang et al., 2018).

Furthermore, it has been demonstrated that calcium overload, induced by an excitotoxic NMDA stimulus on cultured cortical neurons, can be reduced by the activation of AMPK, which would save the energy involved in the reversal of a Ca^{++} potential (Anilkumar et al., 2013). Interestingly, the actions of the catalytic subunit of neuronal AMPK also includes the inhibition of axon outgrowth and dendritic arborization during neuronal development, for adapting to metabolic stress mediated by the suppression of Akt and mTOR signaling pathways (Ramamurthy et al., 2014). This result suggests that AMPK may also operate in mediating structural synaptic changes during the activity of mature neurons, contributing to control energy expenditures in the long-term. Furthermore, it has been shown that the maintenance of long-term potentiation (LTP), which is energetically demanding, is dampened when AMPK activity is pharmacologically activated (mimicking a low ATP/AMP ratio), or conversely, LTP could be rescued when an ATP mimetic, ara-A, was added during an energy challenge. Thus, under low energy conditions, neuronal AMPK tends to inhibit changes on ionic gradients and reduce changes on cytoarchitecture, which can upregulate the value of $A(t)$, impacting plastic capacity as well.

Summarizing, each neuron has a certain amount of ATP available to them, which is constantly consumed by their different functions which can mostly be explained using ion gradient changes on axons and dendrites. At the same time, ATP production will compensate the ATP expenditure reaching an A_H that should remain constant until another specific synaptic challenge arrives (Figure 2). In the next section, we will discuss the potential functional consequences of these adaptations in special cases of neuronal plasticity.

Revisiting Neuronal Plasticity Under the Perspective of Energy Constraints

A narrow window of ATP cytosolic concentration across time supports a bottom-up view of neuronal energy constraints, which may explain some well-described plastic adaptations from the literature. Measurements of glucose and oxygen consumption (reflecting energy consumption) have not distinguished between the contribution from glial and neuronal metabolism and the total energy expenditure attributed to one neuron (Hyder et al., 2013). Nonetheless, neurons would keep energy availability during the increment of energy demands, which include action potentials, potential propagation or dendritic depolarization, by dynamically sharing expenses with astrocytes glial cells (Hyder et al., 2006; Barros, 2013). It is worth mentioning that energy management is partly performed by these latter cells (Magistretti, 2006; Magistretti and Allaman, 2015). Indeed, we must consider that ATP neural production is provided by the local pyruvate and glial lactate. Where a theoretical model aimed to explain brain energy availability from rat and human brains, it indirectly suggested that glial and neuron lactate sources may dynamically vary across different species and activity levels, with the condition of maintaining a rather constant energy production (Hyder et al., 2013).

We will follow a very simplified view of ATP metabolism characterized by two collections of processes: Those that produce ATP (e.g., from local pyruvate and glial lactate), and those that consume ATP (e.g., recovery of ion gradients, structural and functional synapse maintenance). We can formalize the effect of these processes on ATP concentration (A) by a simply differential equation:

$$\frac{\partial A}{\partial t} = P(t, A, \dots) - C(t, A, \dots) \quad (4)$$

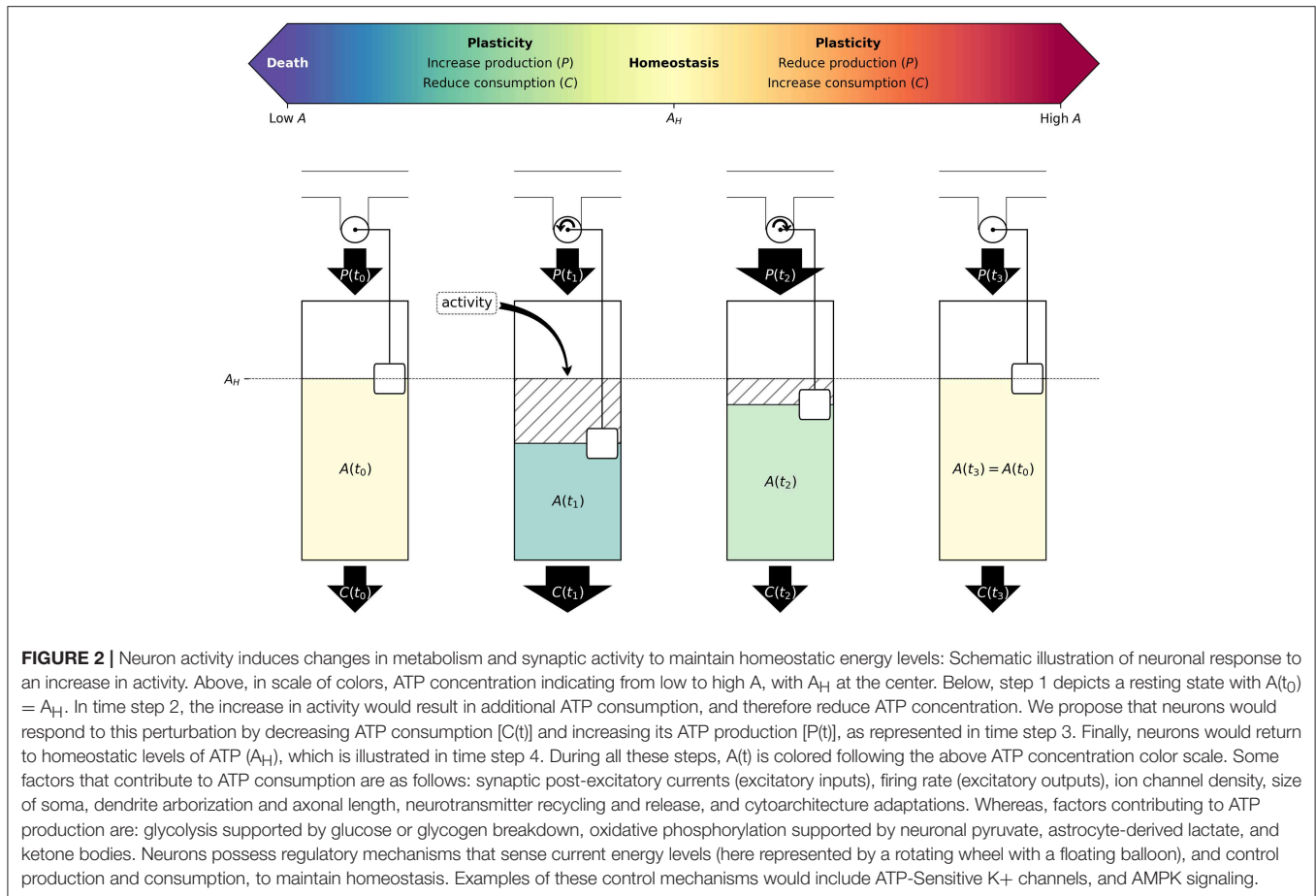
Where, $P(t, A, \dots)$ is a function that represents the sum of all reaction rates that produce ATP (e.g., anaerobic and aerobic metabolism), whereas $C(t, A, \dots)$ is the sum of all reaction rates that consume ATP (e.g., membrane repolarization, structural and functional synapse maintenance). Both production (P) and consumption (C) rates are dynamic (they depend on time), but, more importantly, they depend on the levels of ATP available (A). Homeostasis will be achieved when production and consumption rates are equal, and the concentration of ATP is constant in time. We will represent the homeostatic concentration of ATP by A_H .

We can interpret the observations of relatively constant ATP concentrations in neurons, as reflecting the action of feedbacks that adjust ATP production (P) and consumption (C) rates, compensating deviations of ATP (A), such that neurons return to homeostatic ATP levels (A_H). We can expect that in case cells have an excess of ATP, they would respond by decreasing production or/and increasing consumption; and analogously, in case ATP levels are reduced, cells would respond by increasing production or/and decreasing consumption. We will call this regulation the neuron “energy management,” and summarize it mathematically using these equations:

$$\begin{cases} A > A_H \Rightarrow \frac{\partial P}{\partial t} \leq 0, \frac{\partial C}{\partial t} \geq 0 \\ A < A_H \Rightarrow \frac{\partial P}{\partial t} \geq 0, \frac{\partial C}{\partial t} \leq 0 \end{cases} \quad (5)$$

Meaning that the differences between A with A_H determines whether ATP production (P) and consumption (C) processes increase, decrease, or maintain their rates over time. Note that we also consider the possibility that neurons may respond to energy challenges by adjusting production and consumption, but it must be at least one of those variables.

It is critical to notice that this formalization makes some important simplifications. First, we understand that in addition to ATP, the concentrations of ADP, AMP, and other energy resources do determine homeostasis and influence neuronal changes. ATP is a reasonable departure point, given its prevalence in metabolism and the evidence supporting its role in synaptic plasticity, and therefore, will be the main example of energy resource exploit in our argument. An additional reasonable assumption is that the magnitude of the change in reaction rates should correlate with the magnitude of the distance to homeostasis, which we have omitted from the equations but will become relevant later in our argument for proposing experiments. We expect to expand toward a more detailed formalism in future work. Despite its simplicity, we think that our model can help to understand several previous studies



and propose some experiments aimed at empirically evaluating the relation between energy resource availability and neural plasticity. We expect that simple phenomenological models, such as ours, will encourage both theoretical and experimental efforts, provided they can be readily falsified empirically, and be compared to theoretical derivations from biochemical first principles.

The tendency to set ATP at A_H might be compatible with homeostatic plastic changes that return a neuronal network to a basal firing rate, after prolonged periods of increased or decreased synaptic activity (homeostatic synaptic plasticity). Accordingly, it has been theoretically proposed that the excitability threshold of neurons might be a direct function of ATP (Huang et al., 2007). For example, the K_{ATP} channel-opener diazoxide decreases bursting and regular firing activity of the immature entorhinal cortex neurons (Lemak et al., 2014), which is coherent with a tight association of firing rates with contingent ATP concentration. Also, theoretically, neuronal circuits governed by purely Hebbian-plasticity rules are predicted to converge on instability, or to the opposite—total inactivity (Miller and MacKay, 1994). One possible solution to enable neuronal circuits to remain responsive is to limit the amount of synaptic strength per neuron. At least on excitatory synapses, this problem has shown to be solved by another form of synaptic plasticity

termed “homeostatic synaptic plasticity,” and more specifically, “synaptic scaling” (Turrigiano et al., 1998; Turrigiano, 2012). Synaptic scaling emerges to counteract the effects of long periods of increased or decreased synaptic activity in a multiplicative manner, thus allowing neurons to continuously reset the weight of their synaptic inputs to remain responsive to new environmental and cellular contexts. In the long term, the consequence of this regulation is that the firing rate of cortical cells in culture is sustained to an average set point (Turrigiano, 2012). As far as we know, no attempt has been made to relate or prove the influence of neuronal energy load or $A(t)$ on this phenomenon.

Simple experiments on synaptic scaling could be performed to examine whether the tendency to reach A_H has a predictive value on the synaptic activity of neuronal networks. As shown in the seminal experiments of Turrigiano’s group, when a GABAergic inhibitor bicuculline (Bic) is acutely added to cultured neurons, it produces a significant increase in average firing rate. However, during 48 h of stimulation, firing rates return to control values. On the other hand, neuronal firing rates can be completely abolished soon after adding either tetrodotoxin (TTX) or 6-cyano-7-nitroquinoxaline-2,3-dione (CNQX). Nevertheless, during the 48 h of incubation, activity levels also return to a basal value (Turrigiano et al., 1998). The

observed adaptive changes that operate in the long-term makes this experiment ideal for manipulating energy parameters.

Similar to the experiment of synaptic scaling performed by Turrigiano's group, in our theoretical experiment, cultured neurons would be submitted to 48 h of synaptic activity stimulation with Bic. During the stimulus, $A(t)$ will transiently decrease, inducing plastic changes on the network that will return ATP concentration to A_H , in a given time period (t_1) (**Figure 3A**). In all conditions, $P(t)$ and $C(t)$ change accordingly with $A(t)$, following Equations 4 and 5. However, if, during the stimulus, the neurons were pharmacologically modified to partially decrease ATP production [e.g., by blocking oxidative phosphorylation with sodium azide], expenditures $C(t)$ are expected to be rapidly lowered and the time window required to return to the A_H value will be shortened (**Figure 3B**). Conversely, one could "enlarge" the theoretical value of A_H on cultured neurons by adding an ATP mimetic, such as ara-A. Here we assume that ara-A would cause inhibition of AMPK signaling, and that concentrations employed are low enough not to disturb the ATP synthesis. Thus, we propose that the neurons will take more time to return to A_H (**Figure 3C**). Under these three conditions, the firing rate of neurons should also be adapted to the same level as in the initial state, before stimulation, as well as ATP concentrations should return to the homeostatic value A_H .

FROM MOLECULES TO BEHAVIORAL HOMEOSTASIS

In the previous sections, we have discussed how the energy homeostasis can affect synaptic plasticity in one neuron. Subsequently, this plasticity can impact other neurons that will trigger the same control systems to keep their A_H . Since energy demands are transferred through synapses, and synapses appear or disappear according to energy demands, a network homeostasis comes into play. In this section, we argue that energy constraints scale up a level of organization and how homeostasis in one level is affected by homeostasis in the others.

From Neurons to a Neural Network

The first level is single neuron homeostasis, which is the balance between $C(t)$ and $P(t)$ in single neurons. Importantly, as far as an action potential producing a post-synaptic potential goes, it necessarily imposes an increment in $C(t)$ for the post-synaptic neuron. As such, neurons manage their energy needs which also present an external demand from pre-synaptic neurons, and also imposes an energy demand over the post-synaptic neurons. The fact that a local increase in the $C(t)$ can produce a change in post-synaptic neuron's $C(t)$ supports that energy management is also a neural population property, which we will name network homeostasis. The single neuron homeostasis is closely related to the network homeostasis through a two-way directional interaction, where the network structure imposes constraints on the range of possible homeostatic states that a neuron can achieve, which will, in turn, pose stress on the network through interactions with neighboring neurons. In the

same way, these neurons will respond by modifying their synaptic weights (also known as network connectivity), the number and the location of their synapses, thus changing the functionality of the neural network structure (maybe even micro-anatomically). In any condition that causes an imbalance between $C(t)$ and $P(t)$, the neurons will tend to change. Since neurons activate each other through synapses, this means that the activity of the pre-synaptic neurons will induce metabolic work in the post-synaptic ones. In turn, a post-synaptic neuron will modulate its synaptic weight to couple the input from the pre-synaptic neuron to its own metabolic needs. This process will continue recursively until the neurons balance their $C(t)$ and $P(t)$, in which case the network would have reached homeostasis. Essentially, network homeostasis is driven by the needs of each neuron, as each of them will change in an attempt to reach their own A_H . Note that it is not necessary that every neuron should reach its own A_H , as the connectivity of activity within the network may not allow them to improve further. However, every single neuron must have enough $P(t)$ to devote toward maintenance processes required to stay alive. As such, network homeostasis becomes a neural population property.

Network homeostasis is tightly related to single neuron homeostasis; therefore, neural network homeostasis will be only achieved when several of the neurons that compose it individually can maintain themselves within homeostatic ranges (e.g., achieving A_H). It is known that synaptic and dendrite pruning are a part of healthy development (Huttenlocher, 1990; Riccomagno and Kolodkin, 2015), which we could interpret as adjustments required to couple with the trade-off between maintaining the structure vs. the energy spent in action and post-synaptic potentials. In worse cases where suboptimal conditions are imposed on a single neuron by the neural network homeostasis, we expect to find neuron death. This phenomenon is documented as a part of normal brain development in some species (Huttenlocher, 1990), and also in pathological conditions (Perry et al., 1991; Kostrzewa and Segura-Aguilar, 2003; Pino et al., 2014).

From Neural Networks to Behavior

Behavior can be broadly described as the set of actions performed by an organism, or anything that an organism does that involves movement and response to stimulation. These actions are adaptive when they increase the survival and reproduction probability. In a top-down interpretation of behavior, these actions are the result of the activation of the neuronal circuit that developed evolutionary to fulfill a need. Nonetheless, according to the Energy Homeostasis Principle, at the neural circuitry level, the actions performed by an organism are out of spatial and temporal context, since all the cells experiences are perturbations of the network activity. For a given neuron, the activity dynamics is dependent on the cumulative synaptic currents, regardless of the type of pre-synaptic cells that evoked them, or in the case of sensory receptors, the type of energy that is transduced. Similarly, it makes no difference for a given neuron to have neuron-neuron or neuro-muscular/endocrine synapses. Conversely, we can reinterpret behavior as the observed consequence of the homeostatic activity of an extended neural network (brain) which

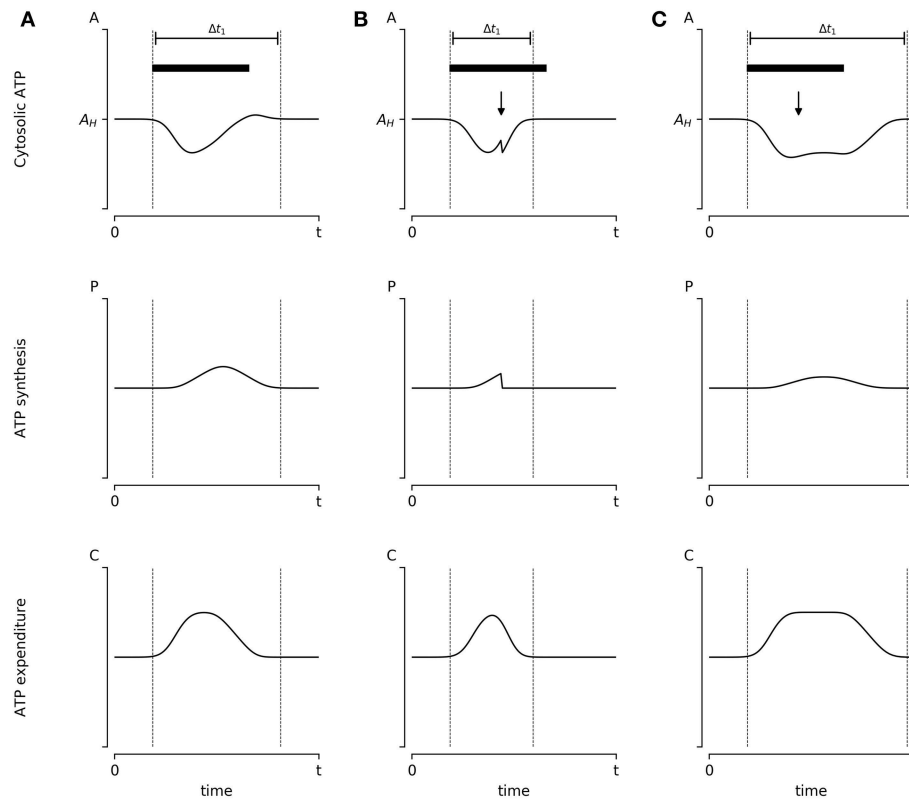


FIGURE 3 | Hypothetical experiment to evaluate the influence of energy availability on synaptic scaling: The influence of energy availability in synaptic plasticity could be evaluated empirically. Here we propose a simple hypothetical scenario and explain what outcomes we predict under each condition. **(A)** Cultured neurons are stimulated with bicuculline (Bic) for 48 h (denoted by a black bar), which presumably induces transient changes in ATP concentration $[A(t)]$. Neurons respond by increasing ATP production $[P(t)]$ and reducing ATP consumption $[C(t)]$ by reducing its firing rate, which leads to the reestablishment of homeostatic ATP concentration within the time window enclosed by dotted lines. **(B)** During stimulation, cultured neurons can be pharmacologically treated to partially inhibit oxidative phosphorylation (i.e., reducing ATP synthesis) denoted by a black arrow. Following the Energy Homeostasis Principle, we propose this will result in a further reduction of ATP concentration, which will induce an accelerated reduction in ATP consumption through the reduction of synapse firing rates. Thus, we propose that in this scenario the time window required to return to homeostasis is shortened. **(C)** Almost an identical protocol to **(B)** is applied to neurons; however, using an ATP mimetic molecule (denoted by the black arrow). We assume that ATP mimetic molecules would delay the reduction of synapse firing rate by allosterically inhibiting AMPK, resulting in an enlarged period of energy consumption. Thus, we propose a wider time window before reaching A_H . All graphics follow Equations 4 and 5, with the additional assumption that the magnitude of the adjustment of $P(t)$ and $C(t)$ are proportional to the distance of ATP levels $A(t)$ to homeostatic levels A_H . Results from these kinds of experiments could advance the understanding (and potentially manipulate) of the mechanisms responsible for neural adaptations, uncovering the relevant role of metabolic elements, such as metabolic sensors and/or nutrient availability.

interacts with the environment. Sensory input and motor outputs can thus be viewed as “environmental synapsis.” Under this framework, what we call behavior may be not necessarily be different from the range of actions neurons engage in any circuit.

However, the interaction with the environment has an important difference that will impact the energy balance in the neuronal network. We can operationalize behavior in a neural system as a set of inputs and outputs that occur in a closed-loop manner. For instance, when we move our eyes, the brain is generating output activity, which in turn modifies the subsequent input to the retina. These dynamics occur for all sensory systems, where motor acts modify sensory inputs (Ahissar and Assa, 2016). In this process, for each brain action, we should expect changes to occur in some sensory inputs. In other words, behavior can be seen as one of the ways in which the brain stimulates itself.

In principle, this closed-loop scheme would enable the brain to completely predict the sensory consequence of the motor actions. This processes of active inference is in line with previous proposals such as Friston’s free energy principle and predictive coding (Friston, 2010; Schroeder et al., 2010). It is crucial to note that Friston’s Free Energy Principle used an informational approach where aspects such as temperature do not refer to the absolute temperature measured in an experimental setting. As such, the Energy Homeostasis Principle does not conceptually overlap with Friston’s proposal; they can be considered as complementary. From a bottom-up view, Friston’s proposal answers the epiphenomena, which can be related to information processing, rather omitting the underlying physiological constraints. However, in any of these proposals, there is an agreement that the brain is capable of predicting sensory input, and that it seems to reduce uncertainty as far

as possible. In the case of Friston's proposal, it refers to the reduction of informational uncertainty, while in the Energy Homeostasis Principle, it refers to the reduction of energy sensorial input uncertainty.

Parsimoniously, the brain cannot fully predict the sensory inputs that occur after every motor act, because changes that are independent of the action of the organism also occur in the environment, and these changes may be critical to its survival. According to the Energy Homeostasis Principle, we should expect that neural networks will operate in a way that will favor the behavioral input activities within homeostatic energy ranges. If a given input is energetically too demanding, we should expect a change in behavior. If a given set of motor activity consistently produces an energy stressor input, it will cause synaptic changes in the brain, as the energy balance processes are spread over the neuronal network.

Sensory input represents the major energy challenge in the brain, while the motor output is the only way a neural network can modify this input. This way, the neuron in the network has the chance to regulate its $C(t)$, given the pressure representing sensory input. Neural networks will restrict the palette of behaviors that can be observed, while behavior will impose energy demands that the neural network will couple with by modifying behavior. For these reasons, behavior can also be considered as a phenomenon which affects the energy homeostasis in a two-way direction. Thus, an at least three-level nested system can be depicted, where each level will have a two-way interaction with each other (see **Figure 4**).

According to the Energy Homeostasis Principle, a key aspect to explaining adaptive behavior must reside within the brain's macro-structure and evolutionary mechanisms. Given a certain palette of sensory specializations, set of effectors, and brain structures, it will impose the range of possible energy disposal attractors that can emerge. For instance, the number of sensory cells, and their sensitivity to stimuli will determine the energy input imposed on neural tissue. The effectors given will determine the space in which a neural network must control that input. The macrostructure of the brain will then impose general restrictions on the neural network homeostasis. For instance, the visual cortex mainly receives visual input, and the communication through brain regions is mainly achieved using major tracts. As such, the series of $C(t)$, imposed by one neuron on another, must follow the paths allowed by the macrostructure. This means, that neural network homeostasis will not only be a function of energy input provided by the sensory cells, and the chances to control it using effectors, but also of macro anatomical restrictions produced by genetic mechanisms (Gilbert, 2000a,b).

Evolutionary pressures must act over all traits—genetic, physiological, and behavioral—of the organism (Darwin, 2003). As such, evolutionary pressures have selected a sensory and effectors pallet, as well as a brain macrostructure. We conjecture that from the set of behaviors that satisfy the energy constrictions, behaviors that statistically improve the chances of surviving will be selected. We propose that the macro-anatomical structures impose a certain system of dynamic energy management among neural networks, which force the emergence of a certain set of energy attractors producing, in turn, a specific set of behaviors.

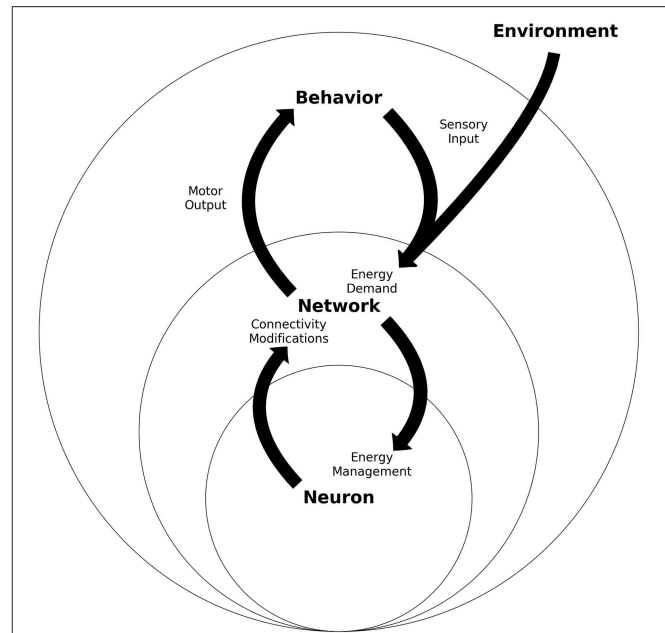


FIGURE 4 | Energy homeostasis: An integrated view of neurons, networks and behavior schematic of the three nested levels of the energy homeostasis system. Each level represents one unit and its proximal operations. Neuron refers to one neuron which must manage its energy consumption, which will trigger neuron-plastic changes. Many of these neurons will build a network, which has connectivity properties and population energy demands. Many networks working together will deploy behavior through motor actions, while also receive the sensory input. All levels present a two-way energy interaction between behavior, neural networks, and neurons. The figure intends to present how sensory input can be considered an energy demand at network and neuron levels, while motor output through behavior gives room to control part of the sensory input.

It is important to consider that in animals that display a large set of behaviors, probably what is selected is the ability to learn. This concretely would mean that the selected is not a specific behavior, rather the flexibility with which an organism must adapt behaviorally during its own life.

Given this bottom-up view, we conclude the existence of behaviors strictly required for survival, and others which might present adaptive advantages given the specific context of the organism. In human primates, for example, there is a vast set of behaviors that are not strictly for the survival of the single individual in any context, yet they exist, such as leisure activities, those related to the production of art in its multiple forms, and even pathological behaviors which might directly impact the individual's health or survival. As far as these non-strictly adaptive behaviors do not impact the organism's life, they might be highly adaptive in certain contexts.

In any case, evolutionary mechanisms will shape the nervous system's macrostructure and behavior so that both are aligned in a way where solving the energy constraints of a single neuron and the neural network will lead to survival. If not, that macrostructure is expected to be lost, as those organisms will die. In fact, there is no need that all these levels work in

alignment per se, rather they must only be aligned to survive. Evolutionary pressures will remove the organisms where the three level systems present goals that don't benefit each other. Since we can only see the animals that present all three level goals aligned, we have historically thought that neurons, neural networks, and organisms share the same goal. We proposed here that evolution shaped organisms, so when a neuron solves their needs, the behavior emerges as an epiphenomenon, which enables the organism to solve its needs, hence surviving.

PERSPECTIVES OF REINTERPRETATION

In this section, we aim to contrast our proposal with evidence and highlight the corollary aspects which can open new avenues of research. Concretely, we will evaluate if we can reassess evidence, considering the Energy Homeostasis Principle. We think that this proposal is parsimonious as in spirit the rule is simple, what makes it complex is the wide range of interactions and properties that can emerge from the neural interaction constraint imposed by this rule. We believe that our proposal captures the essence of the concept of Braitenberg's vehicles (Braitenberg, 1986)¹ and provides a plausible solution to the dynamics elaborated there. Naturally, Energy Homeostasis Principle still has some limitations. It is unclear how we can scale up this principle in networks as large as the brain. The metabolic mechanisms in neurons are quite complex, and still we need more empirical information to tune the mathematical modeling. We decided to use ATP as an energetic proxy, but many other molecules are used by neurons as energetic resources, and may present a dual signaling-resource condition activating control systems. We have mentioned abundant literature that shows an association between energy related variables and neural activity. However, we have not presented direct evidence of how energy constraints shape plasticity and neural network properties. Despite these limitations, the Energy Homeostasis Principle can be tested empirically by associating plasticity markers with energy availability, production, and consumption as mentioned in section "Revisiting neuronal plasticity under the perspective of energy constraints." More importantly, this proposal serves new empirical avenues to study the working of the brain. For instance, plasticity has always been thought to be the changes required to fix a given behavior. However, according to our proposal, plasticity is a process that takes place not only during learning but continuously, as a core component of the constant deployment of behavioral changes. As such, plasticity might not only be a determinant of behavior acquisition but a key aspect of ongoing behavior. In the following subsections, we will briefly discuss different strategies which can be used to extend further from

Equation (5), an example of evidence interpreted considering the Energy Homeostasis Principle, and then discuss other theoretical and empirical avenues which can be reinterpreted based on this paradigm.

Modeling Strategies to Implement Energy Homeostasis Principle

We did not extend our mathematical definitions beyond (Equations 4 and 5) as we aim to set a theoretical ground fertile for different modeling strategies. Equations (4) and (5) describe a quite simple idea that neurons take resources to couple with their energetic demands, and that these two must balance each other in order for the neuron to survive. However, the specific strategies used to operationalize the terms within (Equations 4 and 5) was purposefully left open to avoid constraints into specific modeling paradigms. Equations (1–3) were included to better formalize the problem at a metabolic level. These equations are relevant to build the theoretical argument, however, we would not consider them necessary for modeling, at least in a first approach.

In general terms, Energy Homeostasis Principle requires a dynamic modeling, and a topographic or structural component ideally framed from bottom-up. There is already an example that fits with these requirements (Yuan et al., 2018). In this work, they used the ratio between the energy consumed in synaptic transmission and the total metabolic energy consumed in synaptic transmission and dendritic integration over time. This ratio is used as a third component of Hebbian synaptic plasticity, allowing it to change synaptic weights according to this energetic ratio and pre-synaptic activity. This is a nice example of how to include energetics constraints in neural activity modeling. Under the Energy Homeostasis principle view, the ratio does not make sense in terms of metabolism and neuron needs, because it only addresses energy consumption, without considering the impacts in productions and availability. Therefore, ignoring the restrictions in energy consumption derived from production and availability. This consumption ratio makes sense under a top-down view supported in an information codification logic. Therefore, we suggest to define that ratio according to Equation (4), including consumption, availability, and production following the control mechanisms here presented.

Besides this particular model, graph theory could represent a starting point to define the structure of a dynamic network, in which nodes properties can be updated in a temporal fashion. Graph theory is already used to recall the structural properties of brain networks (Feng et al., 2007; De Vico Fallani et al., 2014), therefore, without a doubt it will be a suitable representation which can be extended to consider the energetic management. Moreover, graph theory contains a vast amount of metrics to characterize networks (Costa et al., 2007), and more importantly, could allow to contrast those metrics against real data (Demirtaş and Deco, 2018; Klinger, 2018).

Strategies such as Free Energy Principle (Friston, 2010), or those that profit of predictive coding conceptions (Spratling, 2008; Schroeder et al., 2010; Huang and Rao, 2011), can also serve as a basis for energy homeostatic modeling. However, we suggest to use energy consumption instead of neural activity as predictor.

¹Braitenberg vehicle is a concept proposed by Braitenberg (1986), referring to a simple vehicle that has two sensors, each connected to an effector, i.e., a wheel, that provides the vehicle the ability to navigate in a given environment. The activation of the sensors can increase or decrease the speed of the respective wheel. In addition to the sensor-effector relationship, the physical configuration of the sensors and the wheels will determine the navigation "behavior" of the vehicle when the stimuli are present in the environment. One major conclusion of this concept is that complex behaviors arise from relatively simple properties (sensors, circuitry, and effectors) of the system in interaction with the environment.

In this case reducing surprise would be analogous to reduce the chances of a neuron to be driven out of energy homeostasis. Energy availability and production ideas are more complex to include. In general terms, and based on the concepts exposed in the previous sections, energy consumption is constrained to energy availability and production. As such, to adapt a predictive coding paradigm requires to include energetic restrictions, which must take into account the rate and amount of energy or activity equivalents that can be managed by the neurons within physiological ranges.

Many other strategies can be used. The above mentioned are often used in neuroscience, however, any modeling strategy that suits the temporal dynamic of energy management, and its topographical bottom-up properties, should be able to capture the essence of the Energy Homeostasis Principle.

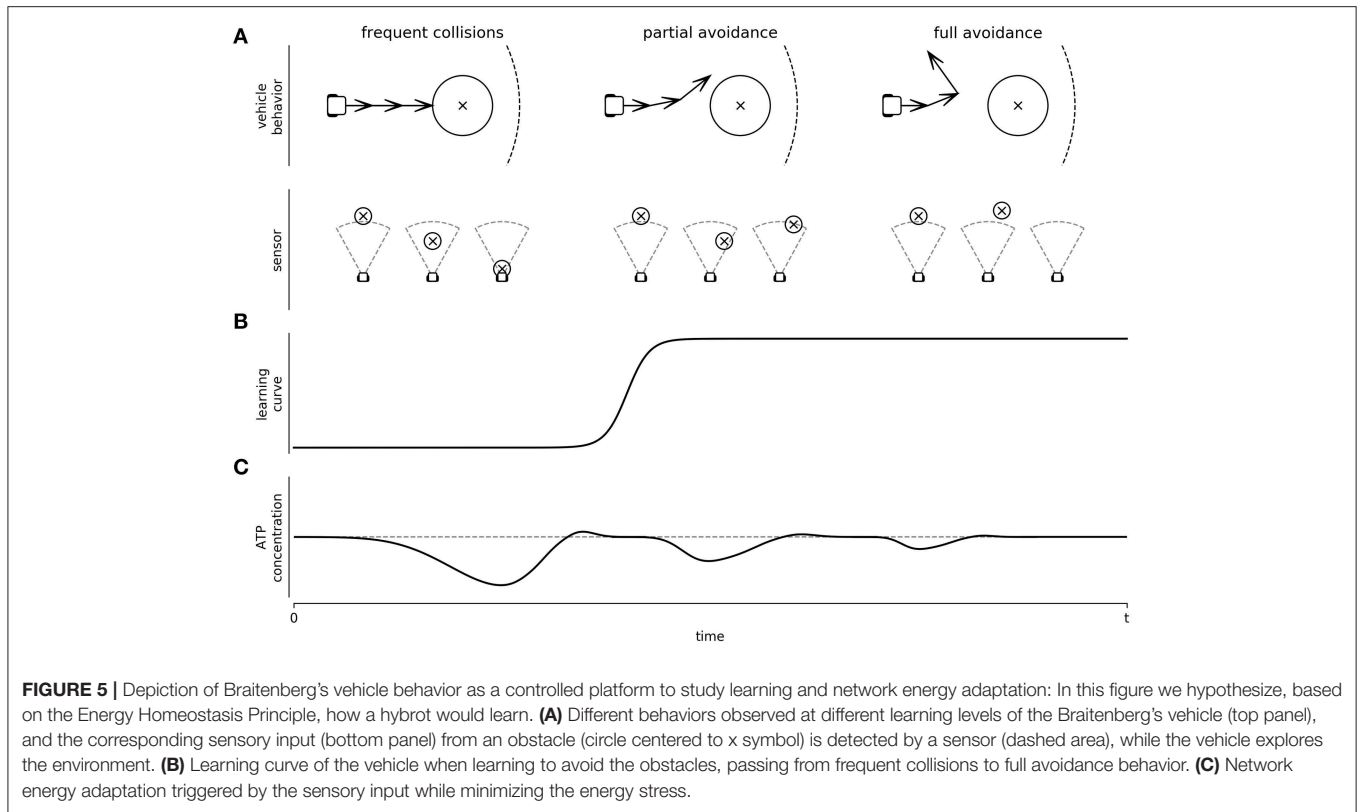
Hybrot: An Analysis Using the Energy Homeostasis Principle

Let us discuss the energy principle proposed here in the context of a simple, *in vivo* network model. Empirically, one critical aspect of relating energy management with behavior is the major challenge of controlling sensory inputs. Most of experimental, *in vivo* animal models are not only sensible to acoustic, visual, physical, and chemical stimuli of the environment, but also to proprioceptive inputs, such as muscle contraction, tendon tension, blood acidification, hormone levels, among others. Strictly speaking, there is no way to properly control the sensory input of an animal *in vivo*, and the behavioral *in vitro* protocol seems to be unreal. Nonetheless, there are some protocols that can be considered as initial efforts of trying to build *in vitro* behavioral protocols. Specifically, some reports demonstrate that if we connect a neuronal culture of dissociated cortical or hippocampal neurons to an external device, coherent behavior can be obtained (Novellino et al., 2007; Tessadori et al., 2013).

Concretely, a system decodes the firing rate of the neurons in the culture and generates an output which is used to control two wheels of a vehicle. The vehicle has distance sensors. The sensor activity is coded to electrical pulses delivered back to the culture. The stimulation frequency is a function of the distance to an obstacle in front of the sensors. If the vehicle gets closer to an obstacle, then the stimulation frequency increases. If the vehicle crashes into an obstacle, a stimulation (20 Hz for 2 s) is delivered, which is previously known to trigger plasticity (Jimbo et al., 1999; Tateno and Jimbo, 1999; Madhavan et al., 2007; Chiappalone et al., 2008; le Feber et al., 2010). Leaving the vehicle in a circular maze with several obstacles under the operation of this protocol will cause it to “learn” to navigate, while avoiding impacts with obstacles (Tessadori et al., 2012). This model constitutes a protocol that enables studying the molecular, electrophysiological, and behavioral properties of neural processing simultaneously; above all, it allows the full control of the sensory input that this network will have.

Is this learning-like phenomenon compatible with the Energy Homeostasis Principle? When a single neuron is submitted to constant stimulation, we expect to have a 1–1 stimulation-action

potential response. However, at a frequency stimulation as low as 10 Hz, the neurons will decay over time until they are unresponsive, or their response is importantly delayed (Gal et al., 2010). If interpreted through the Energy Homeostasis Principle we can hypothesize the following mechanism. First, we can postulate that at a frequency of 10 Hz or higher, stimulations become energetically stressful. As a response, neurons will respond with modifications in their synaptic weights in the short term, and with changes on their cytoarchitecture in the long term. Both processes will result in changes to the network structure. Each time the vehicle crashes, a stressful 20 Hz pulse will be delivered inducing plasticity. Functional restructuration is expected at each impact; leading to a random walk through different neural functional configurations, where each neuron will jump from state to state to minimize energy stress (see **Figure 5**). It is expected that those network configurations that decrease the effects of the sensory input will reduce energy stress due to impacting obstacles. But the best network configuration to the energy stress is indeed to avoid it. Eventually, a network configuration will arise which will prevent the vehicle from crashing. Since no energy stress will be delivered as a sensory input with this configuration, this structure will seemingly stabilize on a configuration of homeostatic energy expenditure (**Figure 5**). We are aware that the above interpretation may oversimplify the actual mechanisms followed by the neurons. Neuronal changes are most likely not completely random and more complex regulations may be taking place. However, we want to point out that they can be sufficient to explain the phenomenology of the observations. As such, energy management, as a local rule, will impact the neural network structure as an emergent property, where, in turn, it will impact behavior. Critically, in this example, we have focused on sensorial input as an increment of neural activity. This might not always be the case (such as under sensorial isolation). Despite that, under this specific scenario, we propose that networks will minimize energy consumption; the goal is to arrive to A_H , not to the minimum possible energy expenditure. Therefore, if the sensorial input would move $A(t)$ below A_H , we would expect network modifications to increase expenses. In any case, the obtained behaviors must be at least compatible with the dynamic constraints imposed by $C(t)$, despite it being too high or low. In this example, behavior emerged to satisfy the energy needs of the neuron by means of $C(t)$. Finally, from all the vehicle movements, only a few, like avoiding the obstacle, might be interpreted as purposeful from an observer's point of view, the remaining ones may be considered a random trajectory. Importantly, this attribute is provided by the observer, as the neurons would only be managing their energetic demands. More research is required to evaluate what is happening with *behavior*, when the obstacles are out of the sensor's range along with the *learning* curve of the vehicle. Nonetheless, the Energy Homeostasis Principle allowed us to propose this hypothesis (**Figure 5C**), and it can be empirically addressed. Naturally, using the same experimental approach, we can evaluate how plasticity is affected by energetic demands induced electrically or by altering neurotransmitter concentrations. We can use the vehicle's behavior, or we can use the Graph Theory index already used to characterize networks



(Costa et al., 2007) to associate neural network properties with energetic demands and metabolic activity.

The Neuron Doctrine and the Energy Homeostasis Principle

Historically, the primary efforts to connect neuron activity with neural network dynamics and behavior was first proposed in 1888 (Barlow, 1972; Bock, 2013), which is referred to as “The Neuron Doctrine,” maintained and developed to this day (Dehaene, 2003; Moser and Moser, 2015; Sheppard, 2016). In general terms, this theoretical proposal tries, in dual form, to solve the information coding and processing problem and has been supported by intracranial recordings, where abundant examples can be found (Letvin et al., 1959; Fairhall, 2014; Moser and Moser, 2015). Specifically, neurons are expected to code for specific properties of the environment, where its activity is associated with the detection of specific stimuli. For instance, neurons in the primary visual cortex of mammals are selectively sensible to oriented bars (Hubel and Wiesel, 1962; Taylor, 1978), while in the Lateral Geniculate there is evidence supporting the existence of circular receptive fields representing portions of visual space (Reid and Shapley, 1992). In these cases, neurons have receptive fields, which can be interpreted as a specific topologic relation of V1 with a certain retinal region; and therefore, with the image. Receptive fields with the same selectivity feature can also be found in the tactile, and auditory primary cortex, evidence of which is often interpreted as environmental stimuli being coded as a map in the brain (Penfield and Boldrey, 1937; Penfield, 1965; Ruth Clemo and Stein, 1982). This classic evidence is also

theoretically line up with the recent hippocampus where neurons (Moser and Moser, 2015).

Critically, most of the evidence supporting the neuron doctrine is associated to the neuron discharge rate. Since this discharge rate is a part of the $C(t)$, it necessarily means that most of the evidence supporting the neuron doctrine supports the Energy Homeostasis Principle as well. For this reason, it is plausible to consider that most of the neuron doctrine evidence is also evidence indicating how energy expenses of one neuron can be directly associated with behavior. Furthermore, high discharge rates, as mentioned above, are expected to trigger plasticity mechanisms. Also, only a low percentage of neurons present high discharge rates (Olshausen and Field, 2005), which should be expected under the Energy Homeostasis Principle scope. Moreover, due to the fact that high discharge rates might trigger changes in functional connectivity (synaptic weights), it should not be surprising that when presenting more complex visual scenes, classic receptive fields are no longer detectable (Fairhall, 2014). We may consider that classic stimulation visual protocols impose an energy input, reflected in the high discharge rate, which needs to be managed. In contrast, visual scenes are regularly experienced, therefore already managed energy, and the firing pattern are considerably lower. As such, we think that the neuron doctrine is not necessarily wrong, rather it has not focused on how the discharge rate is a proxy of energy demands imposed on neurons, which in turn affects their homeostasis. Also, that plasticity might have a functional role in ongoing behavior rather than only stabilizing learned behaviors.

The neural doctrine paradigm has been closely related to information coding paradigms. The coding paradigms follow the same logic as the genetic code; the idea that information is universally coded using the same dictionary or codebook. In the case of genetics, what we call the genetic code, is an arrangement of how the sequence of nucleic acids informs specific amino acid sequences when assembling proteins. In the case of a neural code, the assumption is that environmental stimuli are translated into brain activity, which is then translated into motor output. More specifically, it is possible to map specific neuron activities to specific properties of the environment. For instance, the intensity of the stimulation can be mapped to the discharge rate of the sensory neurons (Gardner and Johnson, 2013). The transduction of the stimuli is usually non-linear and sensitive to differences with previous stimulation rather than the raw value of the stimuli—Weber's law (Gardner and Johnson, 2013). This adaptation law has a direct interpretation in the context of energy expenditure by neurons, as neurons coding raw stimuli would demand a greater energy supply. Weber's law has also been extended to complex cognitive functions, such as quantity estimations (Dehaene, 2003)—where discharge rates are used as the code of quantities for specific neurons—suggesting that energy saving may be a strategy widely used by neurons.

Of course, the discharge rate is far from being the only neural code proposed. Coupled with the complexity of sensory activity, temporal coding was proposed, where the exact temporal relationship between each neuron spike would be the key to understanding how environmental information is translated into brain activity (Connor and Johnson, 1992; Friston, 1997). Temporal coding is implicitly related to energy demands, as the time between action potentials trigger plasticity mechanisms, associated with one of the most expensive items of the neuron physiology—post-synaptic potential and plastic mechanisms (Attwell and Laughlin, 2001). Another strategy was population coding (Georgopoulos et al., 1986; Nicolelis, 2003; Moxon and Foffani, 2015). Population coding uses the activity of a high number of neurons, where the discharge rate, timing, and as many properties can be extracted make it possible for a human or non-human primate to move a robotic arm or similar, with the brain. As more neurons are included, more information is obtained, and we should expect that we will better predict the arm movement. This approximation is good when the aim is to predict behavior but is not useful to understand how behavior emerges from neural activity. If reassessed using Energy Homeostasis Principle, we interpret that population coding works as it is a good assessment of neural network homeostasis, implicitly providing information about plastic changes and neural energy management. Up to some extent, all approaches have to do with when, how much, and which neurons are discharging, which in turn can be interpreted as when and how much energy is expended by individual neurons and the network.

When evaluating evidence related to a whole-brain approach, the neuron doctrine is mostly applied by associating the bold signals of brain regions to specific behaviors. Critically, the fMRI signal is derived, to some extent, by the changes triggered through the glia to couple with the energy demands (Otsu et al., 2015).

Therefore, we can interpret that energy management associated to glial function, is already associated directly with behavior. Moreover, it suggests that energy management can be mapped into networks associated to specific behaviors. Naturally, the specifics in which Energy Homeostasis Principle would impact large networks like brains is still elusive, and it probably would require to incorporate formally the functional properties of the glia.

In general, the fMRI approach strongly resembles the serial symbolic programming paradigms, where a module can be homologized to a programming function, and the network would be the general architecture of the software. The loss of a programming function leads to the loss of a specific functionality of the software. This metaphor was addressed in classic literature (Hovland, 1960; Searle, 1980), suggesting that the brain processes information using a symbolic serial paradigm. As such, most of the neural correlates within the neurocognitive domain are interpreted as information processing, ranging from a strictly symbolic to a correlative information approach. However, using a bottom-up approach and the Energy Homeostasis Principle, those attributions are an observer's bias, as the one described in Braitenberg's vehicles (Braitenberg, 1986). Behavioral functions of a neuron or the neural network would be the epiphenomena of neurons regulating their own homeostasis. In fact, as explained in the previous section, we can describe how the vehicle learns to avoid obstacles without using any informational, symbolic, or teleological explanations. Using this bottom-up approach, it is expected that an informational approach will be useful, as far as the neurons' and the neural network's needs are aligned with the organism's. However, it should be interpreted as an epiphenomenon of neural networks solving their own needs.

Reinterpreting Evidence Toward New Research Avenues

As we have discussed above, energy management, though implicitly considered, is a key feature of the nervous system. This necessarily means that most of our current evidence can be reinterpreted in the light of the Energy Homeostasis Principle. We expect that this reinterpretation will trigger new ideas and strategies to understand the neural phenomena. As an example, we may try to explain the neuronal changes associated with learning processes, based on iconic paradigms such as the long-term potentiation (LTP) and depression (LTD) (Nabavi et al., 2014; Jia and Collingridge, 2017). Both phenomena involve a large amount of energy expense where the ATP could be followed to understand the phenomena of plasticity as one of energy management. This is key, considering that even the Hebbian rules (Kempster et al., 1999), operates differently, according to the neuron type (Abbott and Nelson, 2000), highlighting the difficulties in predicting plasticity according to neural activity. At the same time, the calcium ion plays a critical signaling role within neural physiology, where we should ask if it might be a signal of energy expenditure. It is known that metabolic processes sense the ATP-AMP ratio (Ames, 2000), however, they have not been studied in association to the plasticity phenomena.

Consequentially, we can assess energy management and not solely from a molecular or electrophysiological perspective. For instance, can we consider inhibitory neurons as an adaptive feature to control brain energy expenditure? This is most intriguing if we consider that inhibitory neurons are key to increasing the neural circuits' controlling properties (e.g. negative feedback structures).

Simultaneously, the central nervous system is the only structure of the body which is actively isolated from the vascular system. It has its own system to maintain stable the neuron proximal environment. Moreover, astrocytes coordinate themselves through calcium waves, producing local changes in blood flow and hyperemia (increase on blood irrigation) (Otsu et al., 2015). The brain-blood barrier is not only a filter, but it works functionally to support the energy demands of the neural networks. In fact, synapses are currently suggested as tripartite structures (neuron-neuron and astrocyte) (Wang and Bordey, 2008), where the glutamate-release excitatory synapses are proposed to control neurovascular coupling, and thus, brain energy during conditioning and behavior (Robinson and Jackson, 2016). This would be a clear example of a neural activity involving external support for energy management.

Moreover, there is a vast number of shapes for neural cells. It is currently unknown why some neurons display large, dendritic arborizations and short axons, while others present long axons and rather small dendritic arborizations. Similarly, there are varying basal discharge rates of activity. We think it is worth exploring whether the likelihood of particular morphologies and rate of activities are associated with energy constraints. For instance, can a neuron manage to maintain a long axon and at the same time a huge dendritic arborization where it must maintain a large number of dendritic spines? If we explore the evidence we already have, we are confident that new insights into neuron morphology will appear. Even more, if an unlikely neuron shape or size which is energetically more expensive presents itself; we should expect that those neurons would be more sensible to energy demands and may be more susceptible to neural death (Le Masson et al., 2014). In fact, Paul Bolam proposed that the reason behind Parkinson's is due to the dying out of dopaminergic neurons because of their huge size, which is very expensive in energy terms (Bolam and Pissadaki, 2012; Pissadaki and Bolam, 2013). It is most likely that many of these traits are genetically determined, however, energy constraints might limit the possible morphological variety. Furthermore, that genetic determinants of neuron specializations may be triggered in response to the C(t).

Finally, the Energy Homeostasis Principle paradigm, combined with a bottom-up view, allows us to reinterpret behavior in a much more flexible way. Animals display many behaviors that are not intrinsically adaptive. Leisure activities are an evident example. Why the dog likes to go for the ball or follow a car? Why would we like to learn how to play the piano or to paint? Using a top-down approach would force us to interpret that evolution endorses us with a leisure activity brain module and that all behaviors are somehow beneficial. It seems more parsimonious to think that evolution restricted the system through macrostructure, so that survival-related brain functions will be selected and inherited. Above all, a wide

set of diverse, seemingly useless behaviors can appear, without compromising organism survival or neural needs. Therefore, the only constraint for behavior is that the organisms must stay alive and that the sensory input can be successfully managed, in terms of its energy demand, by the neural networks and the neurons within them. As we already explained before, we think that in the cases of the vehicles controlled by neural cultures, the rules of the stimulation given is critical in understanding how they learn to avoid obstacles. From all the works that reported learning-like properties of *in vitro* dissociated cultures of neurons (Novellino et al., 2007; Mulas and Massobrio, 2010; Tessadori et al., 2012), two main conclusions can be obtained: (1) learning-like properties are not dependent on a priori, highly intricate and sophisticated neural structures, and (2) there is at least one property which does not require a brain evolution argument to explain the emergence of behavior (but probably requires a neural tissue evolution argument). This would be particularly important in relation to behaviors that are not directly tied to survival.

Because of the space limitations, many of these latter considerations are laid out in a basic form. Nonetheless we stress that some of these speculations can be assessed by reviewing the current literature under the Energy Homeostasis Principle rationale. However, the proposal may encourage the development of falsifiable hypotheses, allowing for the testing of these intuitions through empiric work. Therefore, we propose the principle as a novel paradigm from which we can reinterpret neuroscience experimental data, as well inspire the design of experiments which may connect biochemical knowledge to cognitive neuroscience.

AUTHOR CONTRIBUTIONS

RV, SJ-R, and AL developed the initial general argument. SJ-R was the main contributor to the metabolic and biochemistry sections. AL was the main contributor to the single neuron's physiological sections. RV was the main contributor to the neural networks and behavior sections. PM contributed by articulating the different sections to build the manuscript. RV wrote the first draft. CM-L designed and built all the figures, based on all the authors' recommendations. CM-L, AC, and RF contributed with key observations that refined the general argument. All authors revised and edited the manuscript. RV and PM edited the final version.

FUNDING

This work was supported by the post-doctoral FONDECYT grant 3160403, given to RV, the Ph.D. scholarship by The Darwin Trust of Edinburgh, given to SJ-R, the regular FONDECYT grant 1151478, given to RF, the BNI Puente post-doctoral fellowship, given to CM-L, and the BNI Millennium Institute No. P09-015-F, given to AC.

ACKNOWLEDGMENTS

We would like to thank Dr. Felipe Barros for his valuable comments in an earlier version of the manuscript.

REFERENCES

- Abbott, L. F., and Nelson, S. B. (2000). Synaptic plasticity: taming the beast. *Nat. Neurosci.* 3, 1178–1183. doi: 10.1038/81453
- Ahissar, E., and Assa, E. (2016). Perception as a closed-loop convergence process. *Elife* 5, 1–26. doi: 10.7554/eLife.12830
- Ames, A. (2000). CNS energy metabolism as related to function. *Brain Res. Brain Res. Rev.* 34, 42–68. doi: 10.1016/S0165-0173(00)00038-2
- Anilkumar, U., Weisová, P., Düssmann, H., Concannon, C. G., König, H. G., and Pohn, J. H. M. (2013). AMP-activated protein kinase (AMPK)-induced preconditioning in primary cortical neurons involves activation of MCL-1. *J. Neurochem.* 124, 721–734. doi: 10.1111/jnc.12108
- Attwell, D., and Laughlin, S. B. (2001). An energy budget for signalling in the grey matter of the brain. *J. Cereb. Blood Flow Metab.* 21, 1133–1145. doi: 10.1097/00004647-200110000-00001
- Baeza-Lehnert, F., Saab, A. S., Gutiérrez, R., Larenas, V., Díaz, E., Horn, M. (2018). Non-Canonical control of neuronal energy status by the Na⁺ pump. *Cell Metab.* 29, 668–680.e4. doi: 10.1016/j.cmet.2018.11.005
- Barlow, H. B. (1972). Single units and sensation: a neuron doctrine for perceptual psychology? *Perception* 1, 371–394. doi: 10.1068/p010371
- Barros, L. F. (2013). Metabolic signaling by lactate in the brain. *Trends Neurosci.* 36, 396–404. doi: 10.1016/j.tins.2013.04.002
- Ben-Ari, Y., Krnjević, K., and Crépel, V. (1990). Activators of ATP-sensitive K⁺ channels reduce anoxic depolarization in CA3 hippocampal neurons. *Neuroscience* 37, 55–60. doi: 10.1016/0306-4522(90)90191-6
- Berndt, N., Kann, O., and Holzhütter, H. G. (2015). Physiology-based kinetic modeling of neuronal energy metabolism unravels the molecular basis of NAD(P)H fluorescence transients. *J. Cereb. Blood Flow Metab.* 35, 1494–1506. doi: 10.1038/jcbfm.2015.70
- Bhosale, G., Sharpe, J. A., Sundier, S. Y., and Duchon, M. R. (2015). Calcium signaling as a mediator of cell energy demand and a trigger to cell death. *Ann. N. Y. Acad. Sci.* 1350, 107–116. doi: 10.1111/nyas.12885
- Bock, O. (2013). Cajal, Golgi, Nansen, Schäfer and the neuron doctrine. *Endeavour* 37, 228–234. doi: 10.1016/j.endeavour.2013.06.006
- Bolam, J. P., and Pissadaki, E. K. (2012). Living on the edge with too many mouths to feed: why dopamine neurons die. *Mov. Disord.* 27, 1478–1483. doi: 10.1002/mds.25135
- Braitenberg, V. (1986). *Vehicles: Experiments in Synthetic Psychology*, 2nd Edn. Cambridge: MIT Press, 168.
- Brown, A. M., and Ransom, B. R. (2007). Astrocyte glycogen and brain energy metabolism. *Glia* 55, 1263–1271. doi: 10.1002/glia.20557
- Brown, G. C. (1992). Control of respiration and ATP synthesis in mammalian mitochondria and cells. *Biochem. J.* 284, 1–13. doi: 10.1042/bj2840001
- Cannon, W. R. (2014). Concepts, challenges, and successes in modeling thermodynamics of metabolism. *Front. Bioeng. Biotechnol.* 2:53. doi: 10.3389/fbioe.2014.00053
- Cannon, W. R., and Baker, S. E. (2017). Non-steady state mass action dynamics without rate constants: dynamics of coupled reactions using chemical potentials. *Phys. Biol.* 14:055003. doi: 10.1088/1478-3975/aa7d80
- Chiappalone, M., Massobrio, P., and Martinoia, S. (2008). Network plasticity in cortical assemblies. *Eur. J. Neurosci.* 28, 221–237. doi: 10.1111/j.1460-9568.2008.06259.x
- Connolly, N. M. C., Dussmann, H., Anilkumar, U., Huber, H. J., and Pohn, J. H. M. (2014). Single-cell imaging of bioenergetic responses to neuronal excitotoxicity and oxygen and glucose deprivation. *J. Neurosci.* 34, 10192–10205. doi: 10.1523/JNEUROSCI.3127-13.2014
- Connor, C. E., and Johnson, K. O. (1992). Neural coding of tactile texture: comparison of spatial and temporal mechanisms for roughness perception. *J. Neurosci.* 12, 3414–3426. doi: 10.1523/JNEUROSCI.12-09-03414.1992
- Costa, L. D. F., Rodrigues, F. A., Travieso, G., and Villas Boas, P. R. (2007). Characterization of complex networks: a survey of measurements. *Adv. Phys.* 56, 167–242. doi: 10.1080/00018730601170527
- Crooks, G. E. (1999). Entropy production fluctuation theorem and the nonequilibrium work relation for free energy differences. *Phys. Rev. E* 60, 2721–2726. doi: 10.1103/PhysRevE.60.2721
- Danos, V., and Oury, N. (2013). Equilibrium and termination II: the case of petri nets. *Math. Struct. Comput. Sci.* 23, 290–307. doi: 10.1017/S0960129512000126
- Darwin, C. (2003). *On the Origin of Species by Means of Natural Selection*. ed J. Carroll (Peterborough, ON: Broadview Press), 672.
- De Vico Fallani, F., Richiardi, J., Chavez, M., and Achard, S. (2014). Graph analysis of functional brain networks: practical issues in translational neuroscience. *Philos. Trans. R. Soc. B Biol. Sci.* 369:20130521. doi: 10.1098/rstb.2013.0521
- Dehaene, S. (2003). The neural basis of the Weber-Fechner law: a logarithmic mental number line. *Trends Cogn. Sci.* 7, 145–147. doi: 10.1016/S1364-6613(03)00055-X
- Demirtaş, M., and Deco, G. (2018). “Computational models of dysconnectivity in large-scale resting-state networks,” in *Computational Psychiatry: Mathematical Modeling of Mental Illness*, eds A. Anticevic and J. D. Murray (San Diego, CA: Elsevier Academic Press), 87–116. doi: 10.1016/B978-0-12-809825-7.00004-3
- Eguchi, Y., Shimizu, S., and Tsujimoto, Y. (1997). Intracellular ATP levels determine cell death fate by apoptosis or necrosis. *Cancer Res.* 57, 1835–1840.
- Fairhall, A. (2014). The receptive field is dead. *Long live the receptive field? Curr. Opin. Neurobiol.* 25, 9–12. doi: 10.1016/j.conb.2014.02.001
- Feng, J., Jost, J., and Qian, M. (2007). *Networks: From Biology to Theory*, eds J. Feng, J. Jost, and M. Qian (London: Springer). doi: 10.1007/978-1-84628-780-0
- Fodor, J. A. (1983). *The Modularity of Mind: An Essay on Faculty Psychology*. Cambridge, MA: MIT Press, 145. doi: 10.7551/mitpress/4737.001.0001
- Friston, K. (1997). Another neural code? *Neuroimage* 5, 213–220. doi: 10.1006/nimg.1997.0260
- Friston, K. (2002). Beyond phrenology: what can neuroimaging tell us about distributed circuitry? *Annu. Rev. Neurosci.* 25, 221–250. doi: 10.1146/annurev.neuro.25.112701.142846
- Friston, K. (2010). The free-energy principle: a unified brain theory? *Nat. Rev. Neurosci.* 11, 127–138. doi: 10.1038/nrn2787
- Gal, A., Eytan, D., Wallach, A., Sandler, M., Schiller, J., and Marom, S. (2010). Dynamics of excitability over extended timescales in cultured cortical neurons. *J. Neurosci.* 30, 16332–16342. doi: 10.1523/JNEUROSCI.4859-10.2010
- Gardner, E. P., and Johnson, K. O. (2013). “Sensory coding,” in *Principles of Neural Science*, eds E. R. Kandel, J. H. Schwartz, and T. Jessell (New York, NY: McGraw-Hill), 449–474. doi: 10.1126/science.158.3799.399
- Georgopoulos, A. P., Schwartz, A. B., and Kettner, R. E. (1986). Neuronal population coding of movement direction. *Science* 233, 1416–1419. doi: 10.1126/science.3749885
- Gilbert, S. F. (2000a). “Neural crest cells and axonal specificity,” in *Developmental Biology*, ed S. F. Gilbert (Sunderland, MA: Sinauer Associates), 407–441.
- Gilbert, S. F. (2000b). “The central nervous system and the epidermis,” in *Developmental Biology*, ed S. F. Gilbert (Sunderland, MA: Sinauer Associates), 379–410.
- Hardie, D. G. (2011). Sensing of energy and nutrients by AMP-activated protein kinase. *Am. J. Clin. Nutr.* 93, 891S–896S. doi: 10.3945/ajcn.110.001925
- Hardie, D. G., Ross, F. A., and Hawley, S. A. (2012). AMPK: a nutrient and energy sensor that maintains energy homeostasis. *Nat. Rev. Mol. Cell Biol.* 13, 251–262. doi: 10.1038/nrm3311
- Harris, J. J., Jolivet, R., and Attwell, D. (2012). Synaptic energy use and supply. *Neuron* 75, 762–777. doi: 10.1016/j.neuron.2012.08.019
- Herculano-Houzel, S. (2011). Scaling of brain metabolism with a fixed energy budget per neuron: Implications for neuronal activity, plasticity and evolution. *PLoS ONE* 6:e17514. doi: 10.1371/journal.pone.0017514
- Hofmeyr, J.-H. S., and Cornish-Bowden, A. (2000). Regulating the cellular economy of supply and demand. *FEBS Lett.* 476, 47–51. doi: 10.1016/S0014-5793(00)01668-9
- Horton, A. C., and Ehlers, M. D. (2003). Neuronal polarity and trafficking. *Neuron* 40, 277–295. doi: 10.1016/S0896-6273(03)00629-9
- Hovland, C. I. (1960). Computer simulation of thinking. *Am. Psychol.* 15, 687–693. doi: 10.1037/h0044165
- Huang, C.-W., Huang, C.-C., Cheng, J.-T., Tsai, J.-J., and Wu, S.-N. (2007). Glucose and hippocampal neuronal excitability: Role of ATP-sensitive potassium channels. *J. Neurosci. Res.* 85, 1468–1477. doi: 10.1002/jnr.21284
- Huang, Y., and Rao, R. P. N. (2011). Predictive coding. *Wiley Interdiscip. Rev. Cogn. Sci.* 2, 580–593. doi: 10.1002/wcs.142
- Hubel, D. H., and Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.* 160, 106–54. doi: 10.1113/jphysiol.1962.sp006837

- Huttenlocher, P. R. (1990). Morphometric study of human cerebral cortex development. *Neuropsychologia* 28, 517–527. doi: 10.1016/0028-3932(90)90031-I
- Hyder, F., Patel, A. B., Gjedde, A., Rothman, D. L., Behar, K. L., and Shulman, R. G. (2006). Neuronal-glial glucose oxidation and glutamatergic-GABAergic function. *J. Cereb. Blood Flow Metab.* 26, 865–877. doi: 10.1038/sj.jcbfm.9600263
- Hyder, F., Rothman, D. L., and Bennett, M. R. (2013). Cortical energy demands of signaling and non-signaling components in brain are conserved across mammalian species and activity levels. *Proc. Natl. Acad. Sci. U.S.A.* 110, 3549–3554. doi: 10.1073/pnas.1214912110
- Jaynes, E. T. (1965). Gibbs vs Boltzmann entropies. *Am. J. Phys.* 33, 391–398. doi: 10.1119/1.1971557
- Jekabsons, M. B., and Nicholls, D. G. (2004). *In situ* respiration and bioenergetic status of mitochondria in primary cerebellar granule neuronal cultures exposed continuously to glutamate. *J. Biol. Chem.* 279, 32989–3000. doi: 10.1074/jbc.M401540200
- Jia, Z., and Collingridge, G. L. (2017). Learning about synaptic GluA3. *Neuron* 93, 254–256. doi: 10.1016/j.neuron.2017.01.004
- Jimbo, Y., Tateno, T., and Robinson, H. P. C. (1999). Simultaneous induction of pathway-specific potentiation and depression in networks of cortical neurons. *Biophys. J.* 76, 670–678. doi: 10.1016/S0006-3495(99)77234-6
- Kempter, R., Gerstner, W., and Van Hemmen, J. L. (1999). Hebbian learning and spiking neurons. *Phys. Rev. E* 59, 4498–4514. doi: 10.1103/PhysRevE.59.4498
- Klinger, E. G. (2018). *Approximate Bayesian Model Selection for Local Cortical Networks at Cellular Resolution*. Available online at: <https://mediatum.ub.tum.de/doc/1426522/file.pdf> (accessed July 6, 2019).
- Kostrzewa, R. M., and Segura-Aguilar, J. (2003). Novel mechanisms and approaches in the study of neurodegeneration and neuroprotection. A review. *Neurotox. Res.* 5, 375–383. doi: 10.1007/BF03033166
- Lange, S. C., Winkler, U., Andresen, L., Byhrø, M., Waagepetersen, H. S., Hirrlinger, J., et al. (2015). Dynamic changes in cytosolic ATP levels in cultured glutamatergic neurons during NMDA-induced synaptic activity supported by glucose or lactate. *Neurochem. Res.* 40, 2517–2526. doi: 10.1007/s11064-015-1651-9
- Laughlin, S. B. (2001). Energy as a constraint on the coding and processing of sensory information. *Curr. Opin. Neurobiol.* 11, 475–480. doi: 10.1016/S0959-4388(00)00237-3
- le Feber, J., Stegenga, J., and Rutten, W. L. C. (2010). The effect of slow electrical stimuli to achieve learning in cultured networks of rat cortical neurons. *PLoS ONE* 5:e8871. doi: 10.1371/journal.pone.0008871
- Le Masson, G., Przedborski, P., and Abbott, L. F. (2014). A computational model of motor neuron degeneration. *Neuron* 83, 975–988. doi: 10.1016/j.neuron.2014.07.001
- Lemak, M. S., Voloshanenko, O., Draguhn, A., and Egorov, A. V. (2014). KATP channels modulate intrinsic firing activity of immature entorhinal cortex layer III neurons. *Front. Cell. Neurosci.* 8:255. doi: 10.3389/fncel.2014.00255
- Lennie, P. (2003). The cost of cortical computation. *Curr. Biol.* 13, 493–497. doi: 10.1016/S0960-9822(03)00135-0
- Lettvin, J., Maturana, H., McCulloch, W. S., and Pitts, W. H. (1959). “What the frog’s eye tells the frog’s brain,” in *Proceedings of the IRE*. Available online at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4065609 (accessed May 2, 2012).
- Lorenz, D. M., Jeng, A., and Deem, M. W. (2011). The emergence of modularity in biological systems. *Phys. Life Rev.* 8, 129–160. doi: 10.1016/j.plrev.2011.02.003
- Madhavan, R., Chao, Z. C., and Potter, S. M. (2007). Plasticity of recurring spatiotemporal activity patterns in cortical networks. *Phys. Biol.* 4, 181–193. doi: 10.1088/1478-3975/4/3/005
- Magistretti, P. J. (2006). Neuron – glia metabolic coupling and plasticity. *J. Exp. Biol.* 209, 2304–2311. doi: 10.1242/jeb.02208
- Magistretti, P. J., and Allaman, I. (2015). A cellular perspective on brain energy metabolism and functional imaging. *Neuron* 86, 883–901. doi: 10.1016/j.neuron.2015.03.035
- Magistretti, P. J., and Allaman, I. (2018). Lactate in the brain: from metabolic end-product to signalling molecule. *Nat. Rev. Neurosci.* 19, 235–249. doi: 10.1038/nrn.2018.19
- Marcaida, G., Min ana, M. D., Grisolia, S., and Felipo, V. (1995). Lack of correlation between glutamate-induced depletion of ATP and neuronal death in primary cultures of cerebellum. *Brain Res.* 695, 146–150. doi: 10.1016/0006-8993(95)00703-S
- Marcaida, G., Miñana, M.-D., Grisolia, S., and Felipo, V. (1997). Determination of intracellular ATP in primary cultures of neurons. *Brain Res. Protoc.* 1, 75–78. doi: 10.1016/S1385-299X(96)00009-8
- Marsland, R., and England, J. (2018). Limits of predictions in thermodynamic systems: a review. *Reports Prog. Phys.* 81:016601. doi: 10.1088/1361-6633/aa9101
- Miller, K. D., and MacKay, D. J. C. (1994). The role of constraints in hebbian learning. *Neural Comput.* 6, 100–126. doi: 10.1162/neco.1994.6.1.100
- Moser, M.-B., and Moser, E. I. (2015). Where am I? Where am I going? *Sci. Am.* 314, 26–33. doi: 10.1038/scientificamerican0116-26
- Moxon, K. A., and Foffani, G. (2015). Brain-machine interfaces beyond neuroprosthetics. *Neuron* 86, 55–67. doi: 10.1016/j.neuron.2015.03.036
- Mulas, M., and Massobrio, P. (2010). A simulated neuro-robotic environment for bi-directional closed-loop experiments. *Communication* 1, 179–186. doi: 10.2478/s13230-011-0004-x
- Nabavi, S., Fox, R., Proulx, C. D., Lin, J. Y., Tsien, R. Y., and Malinow, R. (2014). Engineering a memory with LTD and LTP. *Nature* 511, 348–352. doi: 10.1038/nature13294
- Nicholls, D. (2013). *Bioenergetics*, 4th Edn. London: Academic Press.
- Nicolelis, M. A. L. (2003). Brain-machine interfaces to restore motor function and probe neural circuits. *Nat. Rev. Neurosci.* 4, 417–422. doi: 10.1038/nrn1105
- Novellino, A., D’Angelo, P., Cozzi, L., Chiappalone, M., Sanguineti, V., and Martinoia, S. (2007). Connecting Neurons to a mobile robot: an in vitro bidirectional neural interface. *Comput. Intell. Neurosci.* 2007, 1–13. doi: 10.1155/2007/12725
- Olshausen, B. A., and Field, D. J. (2005). How close are we to understanding v1? *Neural Comput.* 17, 1665–1699. doi: 10.1162/0899766054026639
- Otsu, Y., Couchman, K., Lyons, D. G., Collot, M., Agarwal, A., Mallet, J.-M., et al. (2015). Calcium dynamics in astrocyte processes during neurovascular coupling. *Nat. Neurosci.* 18, 210–218. doi: 10.1038/nn.3906
- Ouldridge, T. E. (2018). The importance of thermodynamics for molecular systems, and the importance of molecular systems for thermodynamics. *Nat. Comput.* 17, 3–29. doi: 10.1007/s11047-017-9646-x
- Penfield, W. (1965). “Speech, perception and the uncommitted cortex,” in *Brain and Conscious Experience*, ed J. C. Eccles (Berlin; Heidelberg: Springer), 217–237.
- Penfield, W., and Boldrey, E. (1937). Somatic motor and sensory representation in man. *Brain*, 389–443. doi: 10.1093/brain/60.4.389
- Perry, E. K., McKeith, I., Thompson, P., Marshall, E., Kerwin, J., Jabeen, S., et al. (1991). Topography, extent, and clinical relevance of neurochemical deficits in dementia of Lewy body type, Parkinson’s disease, and Alzheimer’s disease. *Ann. N. Y. Acad. Sci.* 640, 197–202. doi: 10.1111/j.1749-6632.1991.tb00217.x
- Pino, O., Guílera, G., Gómez-Benito, J., Najas-García, A., Rufián, S., and Rojo, E. (2014). Neurodevelopment or neurodegeneration: Review of theories of schizophrenia. *Actas Esp. Psiquiatr.* 42, 185–195. Available online at: <https://www.actaspsiquiatria.es/repositorio/16/90/ENG/16-90-ENG-185-195-865491.pdf> (accessed July 6, 2019).
- Pissadaki, E. K., and Bolam, J. P. (2013). The energy cost of action potential propagation in dopamine neurons: clues to susceptibility in Parkinson’s disease. *Front. Comput. Neurosci.* 7:13. doi: 10.3389/fncom.2013.00013
- Potter, W. B., O’Riordan, K. J., Barnett, D., Osting, S. M. K., Wagoner, M., Burger, C., et al. (2010). Metabolic regulation of neuronal plasticity by the energy sensor AMPK. *PLoS ONE* 5:e8996. doi: 10.1371/journal.pone.0008996
- Ramamurthy, S., Chang, E., Cao, Y., Zhu, J., and Ronnett, G. V. (2014). AMPK activation regulates neuronal structure in developing hippocampal neurons. *Neuroscience* 259, 13–24. doi: 10.1016/j.neuroscience.2013.11.048
- Rangaraju, V., Calloway, N., and Ryan, T. A. (2014). Activity-driven local ATP synthesis is required for synaptic function. *Cell* 156, 825–835. doi: 10.1016/j.cell.2013.12.042
- Reid, R. C., and Shapley, R. M. (1992). Spatial structure of cone inputs to receptive fields in primate lateral geniculate nucleus. *Nature* 356, 716–718. doi: 10.1038/356716a0
- Riccomagno, M. M., and Kolodkin, A. L. (2015). Sculpting neural circuits by axon and dendrite pruning. *Annu. Rev. Cell Dev. Biol.* 31, 779–805. doi: 10.1146/annurev-cellbio-100913-013038

- Robbins, P. (2010). *Modularity of Mind. Stanford Encycl. Philosophy*. Available online at: <http://plato.stanford.edu/entries/modularity-mind/> (accessed October 19, 2011).
- Robinson, M. B., and Jackson, J. G. (2016). Astroglial glutamate transporters coordinate excitatory signaling and brain energetics. *Neurochem. Int.* 98, 56–71. doi: 10.1016/j.neuint.2016.03.014
- Ruth Clemo, H., and Stein, B. E. (1982). Somatosensory cortex: a “new” somatotopic representation. *Brain Res.* 235, 162–168. doi: 10.1016/0006-8993(82)90207-4
- Saez, I., Duran, J., Sinadinos, C., Beltran, A., Yanes, O., Tevy, M. F., et al. (2014). Neurons have an active glycogen metabolism that contributes to tolerance to hypoxia. *J. Cereb. Blood Flow Metab.* 34, 945–955. doi: 10.1038/jcbfm.2014.33
- Schroeder, C. E., Wilson, D. A., Radman, T., Scharfman, H., and Lakatos, P. (2010). Dynamics of Active Sensing and perceptual selection. *Curr. Opin. Neurobiol.* 20, 172–176. doi: 10.1016/j.conb.2010.02.010
- Searle, J. R. (1980). Minds, brains, and programs. *Behav. Brain Sci.* 3, 417–424. doi: 10.1017/S0140525X00005756
- Sengupta, B., Faisal, A. A., Laughlin, S. B., and Niven, J. E. (2013). The effect of cell size and channel density on neuronal information encoding and energy efficiency. *J. Cereb. Blood Flow Metab.* 33, 1465–1473. doi: 10.1038/jcbfm.2013.103
- Sheppard, G. M. (2016). *Foundations of the Neuron Doctrine: 25th Anniversary Edition*. Oxford: Oxford University Press. doi: 10.1093/med/9780190259389.001.0001
- Shulman, R. G., Rothman, D. L., Behar, K. L., and Hyder, F. (2004). Energetic basis of brain activity: Implications for neuroimaging. *Trends Neurosci.* 27, 489–495. doi: 10.1016/j.tins.2004.06.005
- Silbey, R. J., Alberty, R. A., and Mouni, B. G. (2004). *Physical Chemistry, 4th Edn*. Hoboken, NJ: Wiley.
- Sokoloff, L. (2008). The physiological and biochemical bases of functional brain imaging. *Cogn. Neurodyn.* 2, 1–5. doi: 10.1007/978-1-4020-8387-7_58
- Spratling, M. W. (2008). Reconciling predictive coding and biased competition models of cortical function. *Front. Comput. Neurosci.* 2:4. doi: 10.3389/neuro.10.004.2008
- Tarasov, A. I., Griffiths, E. J., and Rutter, G. A. (2012). Regulation of ATP production by mitochondrial Ca²⁺. *Cell Calcium.* 52, 28–35. doi: 10.1016/j.ceca.2012.03.003
- Tateno, T., and Jimbo, Y. (1999). Activity-dependent enhancement in the reliability of correlated spike timings in cultured cortical neurons. *Biol. Cybern.* 80, 45–55. doi: 10.1007/s004220050503
- Taylor, C. (1978). Theory of edge detection. *Proc. R. Soc. Lond. B Biol. Sci.* 207, 187–217. doi: 10.1098/rspb.1980.0020
- Tessadori, J., Bisio, M., Martinoia, S., and Chiappalone, M. (2012). Modular neuronal assemblies embodied in a closed-loop environment: toward future integration of brains and machines. *Front. Neural Circuits* 6:99. doi: 10.3389/fncir.2012.00099
- Tessadori, J., Venuta, D., Kumar, S. S., Bisio, M., Pasquale, V., and Chiappalone, M. (2013). “Embodied neuronal assemblies: a closed-loop environment for coding and decoding studies,” in *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)* (San Diego, CA), 899–902. doi: 10.1109/NER.2013.6696080
- Toloe, J., Mollajew, R., Kügler, S., and Mironov, S. L. (2014). Metabolic differences in hippocampal “Rett” neurons revealed by ATP imaging. *Mol. Cell. Neurosci.* 59, 47–56. doi: 10.1016/j.mcn.2013.12.008
- Trevisiol, A., Saab, A. S., Winkler, U., Marx, G., Imamura, H., Möbius, W., et al. (2017). Monitoring ATP dynamics in electrically active white matter tracts. *Life* 6:e24241. doi: 10.7554/eLife.24241
- Turrigiano, G. (2012). Homeostatic synaptic plasticity: local and global mechanisms for stabilizing neuronal function. *Cold Spring Harb. Perspect. Biol.* 4:a005736–a005736. doi: 10.1101/cshperspect.a005736
- Turrigiano, G., Leslie, K. R., Desai, N. S., Rutherford, L. C., and Nelson, S. B. (1998). Activity-dependent scaling of quantal amplitude in neocortical neurons. *Nature* 391, 892–896. doi: 10.1038/36103
- Wang, D. D., and Bordey, A. (2008). The astrocyte odyssey. *Prog. Neurobiol.* 86, 342–367. doi: 10.1016/j.pneurobio.2008.09.015
- Wang, S., Kobayashi, K., Kogure, Y., Yamanaka, H., Yamamoto, S., Yagi, H., et al. (2018). Negative regulation of TRPA1 by AMPK in primary sensory neurons as a potential mechanism of painful diabetic neuropathy. *Diabetes* 67, 98–109. doi: 10.2337/db17-0503
- Weber, B., and Barros, L. F. (2015). The astrocyte: powerhouse and recycling center. *Cold Spring Harb. Perspect. Biol.* 7:a020396. doi: 10.1101/cshperspect.a020396
- Yuan, Y., Huo, H., Zhao, P., Liu, J., Xing, F., et al. (2018). Constraints of metabolic energy on the number of synaptic connections of neurons and the density of neuronal networks. *Front. Comput. Neurosci.* 12:91. doi: 10.3389/fncom.2018.00091
- Yuste, R. (2015). From the neuron doctrine to neural networks. *Nat. Rev. Neurosci.* 16, 487–497. doi: 10.1038/nrn3962

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Vergara, Jaramillo-Riveri, Luarte, Moënne-Loccoz, Fuentes, Couve and Maldonado. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



A Closed-Loop Toolchain for Neural Network Simulations of Learning Autonomous Agents

Jakob Jordan^{1,2*†}, Philipp Weidel^{2,3,4†} and Abigail Morrison^{2,5}

¹ Department of Physiology, University of Bern, Bern, Switzerland, ² Institute of Neuroscience and Medicine (INM-6) & Institute for Advanced Simulation (IAS-6) & JARA-Institute Brain Structure Function Relationship (JBI 1/INM-10), Research Centre Jülich, Jülich, Germany, ³ aiCTX, Zurich, Switzerland, ⁴ Department of Computer Science, RWTH Aachen University, Aachen, Germany, ⁵ Faculty of Psychology, Institute of Cognitive Neuroscience, Ruhr-University Bochum, Bochum, Germany

OPEN ACCESS

Edited by:

Mario Senden,
Maastricht University, Netherlands

Reviewed by:

Laurent U. Perrinet,
UMR7289 Institut de Neurosciences
de la Timone (INT), France

Daniel Saunders,
University of Massachusetts Amherst,
United States

*Correspondence:

Jakob Jordan
jordan@pyl.unibe.ch

[†] These authors have contributed
equally to this work

Received: 31 May 2019

Accepted: 25 June 2019

Published: 02 August 2019

Citation:

Jordan J, Weidel P and Morrison A
(2019) A Closed-Loop Toolchain for
Neural Network Simulations of
Learning Autonomous Agents.
Front. Comput. Neurosci. 13:46.
doi: 10.3389/fncom.2019.00046

Neural network simulation is an important tool for generating and evaluating hypotheses on the structure, dynamics, and function of neural circuits. For scientific questions addressing organisms operating autonomously in their environments, in particular where learning is involved, it is crucial to be able to operate such simulations in a closed-loop fashion. In such a set-up, the neural agent continuously receives sensory stimuli from the environment and provides motor signals that manipulate the environment or move the agent within it. So far, most studies requiring such functionality have been conducted with custom simulation scripts and manually implemented tasks. This makes it difficult for other researchers to reproduce and build upon previous work and nearly impossible to compare the performance of different learning architectures. In this work, we present a novel approach to solve this problem, connecting benchmark tools from the field of machine learning and state-of-the-art neural network simulators from computational neuroscience. The resulting toolchain enables researchers in both fields to make use of well-tested high-performance simulation software supporting biologically plausible neuron, synapse and network models and allows them to evaluate and compare their approach on the basis of standardized environments with various levels of complexity. We demonstrate the functionality of the toolchain by implementing a neuronal actor-critic architecture for reinforcement learning in the NEST simulator and successfully training it on two different environments from the OpenAI Gym. We compare its performance to a previously suggested neural network model of reinforcement learning in the basal ganglia and a generic Q-learning algorithm.

Keywords: closed-loop simulation, reinforcement learning, spiking neuronal networks, virtual environments, computational neuroscience

1. INTRODUCTION

Simulation is a key component of modern neuroscience, constituting a third methodological pillar along with experiment and theory. Its uses include, but are not limited to, validation of theory, generation of hypotheses, production of surrogate data for data analysis tools, and discovery of structural and dynamical constraints for functional models. Thanks to a variety of initiatives, researchers now have access to well maintained, high performance simulators for all scales of

neural systems from molecular simulations (e.g., STEPS; Wils and De Schutter, 2009) over complex neuron (e.g., NEURON, Carnevale and Hines, 2006; GENESIS, Bower and Beeman, 2007) and network models (e.g., NEST, Gewaltig and Diesmann, 2007; BRIAN, Goodman and Brette, 2009; NENGO, Bekolay et al., 2013, SINABS, Sheik and Liu, 2019) to whole brain simulations using neural fields (e.g., TVB, Sanz Leon et al., 2013).

In the realm of spiking neural networks, development of simulators has been largely driven by two viewpoints: the physical, concerned with the dynamics of individual neurons and networks of neurons (e.g. relationship of correlation structure to connectivity, bifurcation landscapes), and the electrophysiological, concerned with the response of neurons and networks to stimuli (e.g. PSTHs, response variability). Thus, spiking neural network simulators provide good support for constructing structured networks of neurons with a variety of dynamics, applying arbitrarily complex stimuli and recording the evolution of dynamic variables for later analysis.

However, restricting our inquiries to the dynamical and transformational properties of neuronal networks neglects large classes of fundamental and exciting neuroscientific questions. In particular, investigations of embodied cognition, of organisms operating autonomously in an environment and learning how to optimize their behavior within it, require a different approach. Firstly, it is crucial to simulate agents that interact with their environments, thereby actively shaping their future sensations rather than merely passively consuming experimentally provided stimuli (see e.g., Wilson, 2002). This necessitates a *closed-loop* set-up, in which the neuronal network can be conceived of as an autonomous agent within an environment. The neuronal network receives sensory stimuli from the environment, which alter its network dynamics. The resulting activity of the network, or of specific subnetworks of it can be interpreted as motor commands which alter the agent's configuration with respect to its environment (e.g., rotation, lateral movement) or the configuration of the environment itself (e.g., operation of levers or buttons). The change in configuration brings about a change in the sensory stimuli, and thus the neuronal network interacts with the environment in a continuous cycle. Depending on the scientific question, the network activity can also drive plasticity processes in the network, causing alterations in its own configuration, and thus in its response to sensory stimuli. In this way, new behavior can be learned through interaction with the environment, rather than through extensive exposure to labeled training data.

Secondly, it is important to establish a set of standardized benchmarks which allow alternative models to be compared with each other and good models to be improved and extended. With regard to this latter point, a comparison of the progress of the fields of machine learning, and learning in neuronal networks, provides a useful illustration. The last decade has witnessed major progress in the field of machine learning, moving from small-scale toy problems to large-scale real-world applications including image (Krizhevsky et al., 2012) and speech recognition (Hinton G. et al., 2012), complex motor-control tasks (Mnih et al., 2016), and playing (video) games at super-human performance (Mnih et al., 2015; Silver et al., 2016). This progress

has been driven mainly by an increase in computing power, especially by training deep networks on graphics processing units (Raina et al., 2009), and conceptual breakthroughs like layer-wise pretraining (Hinton and Salakhutdinov, 2006; Bengio et al., 2007) or dropout (Hinton G.E. et al., 2012). Even so, this rate of progress would not have been possible without the wide availability of high-performance ready-to-use tools, e.g., Torch (Collobert et al., 2002), Theano (James et al., 2010), Caffe (Jia et al., 2014), TensorFlow (Abadi et al., 2016), and standardized datasets and environments for benchmarking, such as the MNIST (LeCun et al., 1998), CIFAR (Krizhevsky and Hinton, 2009), and ImageNET (Deng et al., 2009) datasets, and the MuJoCo (Todorov et al., 2012), ALE (Bellemare et al., 2015), and OpenAI Gym (Brockman et al., 2016) toolkits. While ready-to-use tools allow researchers to focus on important aspects rather than basic implementation details, standardized benchmarks have guided the community as a whole toward promising approaches, as for example in the case of convolutional networks through the ImageNET competition (Russakovsky et al., 2015).

Similarly, researchers in the field of computational neuroscience have benefited from the increase of computational power and achieved many conceptual breakthroughs over the last decade, with a plethora of new neuron, synapse and network models being developed. As mentioned above, a variety of simulators are available to the computational neuroscientist, yet so far no generally accepted set of benchmarks exist (but see Gerstner and Naud, 2009).

One particular area in which the lack of standardized benchmarks is apparent is research into reinforcement learning (RL) in neurobiological substrates. Inspired by behavioral experiments, RL is concerned with the ability of organisms to learn from previous experiences to optimize their behavior in order to maximize reward and avoid punishment (see e.g., Sutton and Barto, 1998). RL has a long tradition in the field of machine learning which has led to several powerful algorithms, such as SARSA and Q-learning (Watkins, 1989). Similarly, a large variety of neurobiological models have been proposed in recent years (Izhikevich, 2007; Potjans et al., 2009, 2011; Urbanczik and Senn, 2009; Vasilaki et al., 2009; Frémaux et al., 2010; Frémaux et al., 2013; Jitnev et al., 2012; Friedrich et al., 2014; Rasmussen and Eliasmith, 2014; Aswolinskiy and Pipa, 2015; Baladron and Hamker, 2015; Rombouts et al., 2015; Friedrich and Lengyel, 2016; Rueckert et al., 2016). However, only a small proportion of these rely on publicly available simulators and all of them employ custom built environments. Even for fairly simple environments, this has led to a situation where different network models are difficult to compare and reproduce, thus creating a fragmentation of research efforts. Instead of building upon and extending existing models, researchers are forced to spend too much time on recreating basic functionality for custom implementations.

The need for closed-loop simulation has led to the Human Brain Project (2014) (HBP) dedicating significant resources of a subproject (Neurorobotics) to the development of the necessary infrastructure that allows users to conduct robotic experiments in virtual environments and connect these to their neural network implementations with a web interface (Falotico et al., 2017). This approach specifically addresses the need of

researchers developing neuronal or neuro-inspired controllers for robotic applications. A more pared-down approach, suitable for researchers who are primarily concerned with understanding the neural circuits, rather than controlling sophisticated robotic actuators, is provided by Weidel et al. (2016). This approach allows any neuronal network simulator that implements the MUSIC (Djurfeldt et al., 2010) interface (including NEST and NEURON) to be coupled with any robotic simulator implementing the ROS (Quigley et al., 2009) interface [including Gazebo (Koenig and Howard, 2004), Morse (Echeverria et al., 2011), or Webots (Michel, 2004)].

However, neither approach directly addresses the issue of the lack of standardized benchmarks for neuronal agents operating autonomously and learning to optimize their behavior in an environment. Such benchmarks exist: the OpenAI Gym (Brockman et al., 2016) provides a rich and generic collection of standardized RL environments developed to support the machine learning community in evaluating and comparing algorithms. All environments are accessible via a simple, unified interface, that requires an agent to supply an action and returns an observation and reward for its current state. The toolkit includes a range of different environments with varying levels of complexity ranging from low-dimensional fully discrete (e.g., *FrozenLake*¹) to high-dimensional fully continuous tasks (e.g., *Humanoid*¹). The consistency of the OpenAI Gym environments across different releases supports researchers in reproducing and extending previous work and allows systematic benchmarking and comparison of learning algorithms and their implementations. The easy accessibility of different tasks fosters progress by allowing researchers to focus on learning algorithms instead of basic implementation details of particular environments, and prompts researchers to evaluate the performance of their algorithms on many different tasks.

One possibility to access this set of benchmarks is to implement spiking networks in tools that are natively compatible with the OpenAI Gym, such as Tensorflow (Abadi et al., 2016) or PyTorch (Paszke et al., 2017). However, as the components of spiking neural network models (e.g., neuron and plastic synapse models, stimulation, and recording devices) are typically not shipped with these tools, this once again places the burden of implementation on the user (but see Hazan et al., 2018 for a spiking-neural network orientated approach). In particular, since these tools focus on machine learning applications rather than exploring biological intelligence, several critical features for computational modeling of learning in biological neuronal networks, such as few-compartment neurons, conductance-based synaptic interactions or neuromodulated plasticity, lie outside the scope of these libraries. Therefore, to make a comprehensive resource of benchmarks available to the computational neuroscience community, we developed a toolchain to interface neural network simulators with the OpenAI Gym. Using this toolchain, researchers can rely on well-tested, high-performance simulation engines for spiking neural networks to power their models, and evaluate them against a curated set of standardized environments, allowing more

time to focus on neurobiological questions, such as the configuration and plasticity of neural circuits underlying exploration of the environment and exploitation of prior experience.

In the next section we introduce additional pre-existing components on which our toolchain relies, and afterwards discuss how it links the different tools. We demonstrate its functionality by implementing a neural actor-critic in NEST and successfully training it on two different environments from the OpenAI Gym.

2. PRE-EXISTING COMPONENTS

All network simulations in this manuscript are carried out with NEST² (Gewaltig and Diesmann, 2007), a neural simulator designed for the efficient simulation of large-scale networks of simple spiking neuron models with biophysically realistic connectivity. The simulation kernel scales from small simulations on a laptop to super computers, with the largest simulation to date containing about 10^9 neurons and 10^{13} synapses, corresponding to about 10% of the human cortex at the resolution of individual cells and connections (Kunkel et al., 2014; Jordan et al., 2018). NEST is actively developed and maintained by the NEST initiative³ in collaboration with the community, is freely available under the GPLv2 and is supported by the HBP with the explicit aim of widespread long-term availability and maintainability. The simulation set-up, e.g., definition of neurons and connections, can conveniently be performed via an interpreted language (e.g., PyNEST; Eppler et al., 2009) while the propagation of network dynamics is implemented in C++. OpenMP is used for node-local parallelization while MPI provides inter-node communication. While using a compiled language for the compute-intensive part provides significant performance gains compared to an interpreted language, it makes it less straightforward to interface the simulator with other tools not specifically designed for this.

The OpenAI Gym (Brockman et al., 2016) is a toolkit for reinforcement learning research focused on ease of use for machine learning researchers. An explicit goal of the OpenAI Gym is to compare different RL algorithms with each other in a consistent fashion. It provides a unified Python interface to a rich collection of curated RL environments, e.g., Atari games⁴ or continuous control tasks for robotic applications⁵.

An environment in the OpenAI Gym is updated in steps. In each step, the agent receives an observation representing the state of the environment, e.g., the agent's location within it, or other configurational information. This is typically a vector of real values. In addition, it receives a real-valued reward for entering the current environmental state. Depending on the environmental set-up, the reward may be zero for the majority of state transitions, and only non-zero (positive for rewards or negative for punishments) when the agent achieves a well-defined

²<http://nest-simulator.org/>

³<https://nest-initiative.org/>

⁴<https://gym.openai.com/envs/#atari>

⁵<https://gym.openai.com/envs/#mujoco>

¹<https://gym.openai.com/envs>

goal. On the basis of the current state and its internal policy, the agent provides an action to the environment to trigger the next state transition. The reward can be used as information to adjust the agent's policy, such that its behavior in the environment evolves, typically such that it receives more reward in future trials in the same environment.

While the network implementation that we present in the results section relies on the NEST simulator, the toolchain can also be used with other simulators that support the MUSIC library, for example NEURON (Carnevale and Hines, 2006). The Multi-Simulation Coordinator is a multi-purpose middleware for neural network simulators built on top of MPI (Message Passing Interface) that enables online interaction of different simulation engines (Djurfeldt et al., 2010). MUSIC takes care of starting all MUSIC-controlled executables (e.g., adapters and simulators) defined in a configuration file provided by the user in separate processes. During execution it makes sure that all processes evolve synchronously with a predefined real-time factor independent of the computational load of the individual processes (Moren et al., 2015). MUSIC provides named MPI channels, referred to as MUSIC ports, which allow the user to set up communication streams between several processes. While originally intended to distribute a single neural network model across different simulators, the MUSIC library can also be used to connect neural simulators to other applications.

For example, to connect neural simulators to robotic simulators, we recently developed the ROS-MUSIC Toolchain (RMT; Weidel et al., 2016) which provides an interface from MUSIC to the Robotic Operating System (ROS; Quigley et al., 2009). ROS is the most popular middleware in the robotic community and is able to interact with many robotic simulators and hardware platforms. The RMT allows exchange of well-defined messages between ROS and MUSIC via stand-alone executables, so called adapters, that were designed with a focus on modularity. The toolchain contains several different adapters each performing a rather simple operation on streams of inputs (e.g., filtering). By concatenating several adapters, the overall transformation of the original data can become more complex, for example converting high-dimensional continuous data (e.g., sensory data) to low-dimensional discrete data (e.g., action potentials) or vice-versa. More information and introductory examples can be found on GitHub⁶.

3. RESULTS

To enable the online interaction of neural network simulators and the OpenAI Gym, we rely on two different libraries: MUSIC, to interface with the neural simulator, and ZeroMQ (Hintjens, 2013) to exchange messages with the environment simulated in the OpenAI Gym. In the following, we describe these two parts of the toolchain and demonstrate their functionality by interfacing a neural network simulation in NEST with two different environments.

3.1. Extending the ROS—MUSIC Toolchain

We extended the RMT by adding adapters that support communication via ZeroMQ following a publish-subscribe pattern. ZeroMQ is a messaging library that allows applications to exchange messages at runtime via sockets. Continuously developed by a large community, it offers bindings for a variety of languages including C++ and Python, and supports most operating systems. A single communication adapter of the RMT sends (receives) data via a ZeroMQ socket and receives (sends) data via a MUSIC port. While the adapters can handle arbitrary data, we defined a set of specialized messages in JSON format (see **Supplementary Material**) specifically designed to communicate observations, rewards, and actions as discrete or continuous real-valued variables of arbitrary dimensions, as used in the OpenAI Gym. We chose the JSON format due to its simplicity, easy serialization and broad platform support.

In addition to the ZeroMQ adapters dedicated for communication with MUSIC, we developed several further adapters that can perform specific transformations of the data. OpenAI Gym places few restrictions on the nature of the environment: it can be continuous or discrete with arbitrary dimensionality. Thus, in order to generate the required closed-loop functionality, the observations provided by the environment must be consistently transformed to a format that can be fed into neural network simulations. Conversely, the activity of the neural network must be interpreted and transformed into valid actions which can be executed in the environment.

A standard way to address the first issue with some degree of biological plausibility is to introduce a layer of *place cells* (Moser et al., 2008). Each of these cells is tuned to a preferred (multidimensional) observation, i.e., is highly active for a specific input and less active for other inputs (see e.g., Frémaux et al., 2013). The dependence of the activity of a single place cell on observations is described by its tuning curve, often chosen as a multidimensional Gaussian. To perform the transformation of observations to activity of place cells, we implemented a *discretize adapter* that allows users to specify the position and width of the tuning curves of an arbitrary number of place cells. One disadvantage of this approach is that the number of place cells required to cover the whole observation space evenly scales exponentially in the number of dimensions of the observation. For observations with a small number of dimensions, however, this approach is very suitable.

To perform action selection, we added several adapters that can, respectively, select the most active neuron (*argmax adapter*), threshold the activity across neurons to create a binary vector (*threshold adapter*), or linearly combine the activity of neurons across many input channels (*linear decoder*). Depending on the type of action required by the environment (discrete/continuous), the user can select a single one or a combination of these. Specifications of the adapters can be found in the documentation of the RMT⁷.

In general, we followed the design principle behind the RMT and developed modular adapters. This makes each individual

⁶<https://github.com/incf-music/ros-music-adapters>

⁷<https://github.com/incf-music/music-adapters>

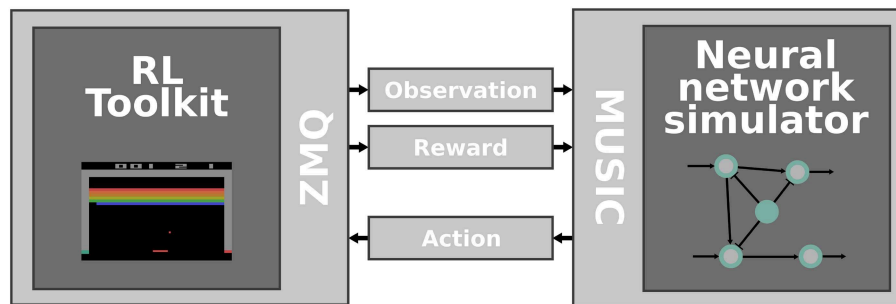


FIGURE 1 | Interfacing RL toolkits with neural network simulators. The RL toolkit (**left**) is responsible for emulating an environment that provides observations and rewards which are communicated via ZeroMQ sockets and MUSIC adapters (**middle**) to a neural network simulator (**right**). Concurrently, the activity of the simulated neural network is transformed to an action and fed back to the RL toolkit.

adapter easy to understand and enables users to quickly extend the toolchain with their own adapters. By combining several adapters, the RMT allows arbitrarily complex transformations of the data and can hence be applied to many use-cases.

3.2. ZeroMQ Wrapper for the OpenAI Gym

The second part of the toolchain is a Python wrapper around the OpenAI Gym that exposes ZeroMQ sockets (Hintjens, 2013) for communicating actions, observations and rewards (see section 2 and **Figure 1**). The wrapper consists of four different threads that coordinate: (i) performing steps in an environment, (ii) receiving actions via a ZeroMQ SUB socket, (iii) publishing observations via a ZeroMQ PUB socket, and (iv) publishing rewards via a ZeroMQ PUB socket.

Before spawning the threads, the wrapper starts a user-specified environment and creates the necessary communication buffers. The thread coordinating the environment reads actions from the corresponding buffer, performs single steps in the environment and updates the observation and reward buffers based on the return values of the environment. Upon detecting that a single episode has ended, e.g., by an agent reaching a certain goal position, it resets the environment and allows a break of user-specified duration before starting the next episode.

The communication threads continuously send (receive) messages via ZeroMQ and read from/write to the corresponding buffers. All threads can be run with different update intervals, for example, to slow down movement of the agent by performing steps on a coarse time grid whilst continuously receiving action choices from the neural network simulation running on a fine time grid. The user can specify a variety of parameters via a configuration file in JSON format (see **Supplementary Material**). Detailed specifications of the wrapper can be found in the documentation.

In contrast to MUSIC-controlled executables, the ZeroMQ wrapper is not started by the MUSIC library. As a result, the environment and the simulation evolve simultaneously but asynchronously. The simulator hence continuously receives input from the environment and vice versa. Due to the possibility of choosing a real-time factor for MUSIC-controlled processes, the user can easily achieve reliable interaction between the

environments and the network simulation. The loosely coupled, asynchronous nature of the toolchain has the benefit that one could, for example, train the same network on a wide variety of different environments without stopping the simulation, in order to investigate transfer learning in spiking neural networks.

3.3. Applications

To demonstrate the functionality of the toolchain, we implemented a neural network in NEST and trained it on two different environments simulated in the OpenAI Gym. In the first task the agent needs to learn to perform a sequence of actions in order to reach the top of a hill in a continuous environment. The second task is a classical grid-world in which an agent needs to learn to navigate to a goal position in a two-dimensional discrete environment with obstacles. We first describe the neural network architecture and learning rule and afterwards discuss the network's performance on the two tasks.

3.3.1. Neural Network Implementation

We consider a temporal-difference learning algorithm (Sutton and Barto, 1998) implemented as an actor-critic architecture based on the spiking neuronal network proposed by Frémaux et al. (2013). For the purpose of demonstrating the toolchain, we simplified the model by replacing the spiking neuron models with rate neurons, thereby avoiding issues arising from noise introduced by spiking neuron models (Potjans et al., 2011; Frémaux et al., 2013). Note, however, that the toolchain is not restricted to rate-based models; any neuron model available in the neural simulators with MUSIC interfaces can be used.

The neuron dynamics we considered here are given by the following stochastic differential equation:

$$\tau \frac{dz_i(t)}{dt} = -z_i(t) + \mu_i + f(h_i(t) - \theta_i) + \xi_i(t), \quad (1)$$

where τ is some positive time constant, μ_i a baseline activity level, $f(\cdot)$ some (arbitrary) activation function, $h_i(t)$ a time dependent input field, θ_i an input threshold and $\xi_i(t)$ Gaussian white noise with a certain standard deviation σ_ξ . The input field $h_i(t)$ is determined by the activity of other neurons according to $h_i(t) :=$

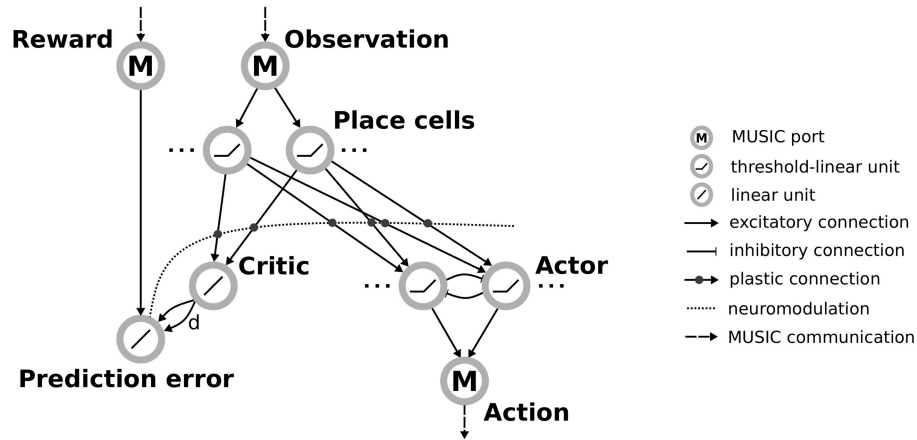


FIGURE 2 | Actor-critic architecture for reinforcement learning with rate neurons. Observations are communicated via a MUSIC input port to a population of place cells. These project both to a critic unit and to actor units arranged in a winner-take-all circuit. The critic and an additional MUSIC input port project to a unit representing the reward prediction error that modulates the plasticity between the place cells and their downstream targets, the critic and actors. The actor units project to a MUSIC output port encoding the selected action.

$\sum_j w_{ij} z_j(t)$, with w_{ij} denoting the strength of the connection (weight) from neuron j to neuron i . Here we will exclusively consider activation functions of the form $f(x) := x$ (linear case), and $f(x) := \Theta(x)x$ (threshold-linear case, “relu”). Here $\Theta(\cdot)$ denotes the Heaviside function, defined as

$$\Theta(x) := \begin{cases} 1 & x > 0 \\ 0 & \text{else} \end{cases} \quad (2)$$

Neuron dynamics are integrated in NEST on a fixed time-grid by a stochastic-exponential-Euler method with a step size determined by the resolution of the simulation. For more details on the neuron model implementation (see Hahne et al., 2017).

The input layer is a population of threshold-linear rate neurons which receive inputs through MUSIC and encode observations from the environment (see **Figure 2**). These place cells project via plastic connections to a single neuron representing the value that the network assigns to the current state (the *critic*). An additional neuron calculates the reward-prediction error by combining the reward received from the environment with input from the critic. Plasticity of the projections from inputs to the critic is modulated by this reward prediction error, as described below.

In addition, neurons in the input layer project to a population of neurons representing the available actions (the *actor*). To enforce selection of a specific action, the actor units are arranged in a winner-take-all (WTA) circuit. This is implemented by recurrent connections between actor units that correspond to short-range excitation and long-range inhibition, the distance reflecting the similarity of the action that actor units encode. The activity of actor units is transformed to an action supported by the environment and communicated to the environment via the RMT.

To derive a learning rule for the critic, we follow similar steps as described by Frémaux et al. (2013), but applied to rate

models (Equation 1). The critic activity should approximate a continuous-time value function defined by Doya (2000):

$$V^\pi(t) := \int_t^\infty r(s^\pi(t')) e^{-\frac{t'-t}{\tau_r}} dt'. \quad (3)$$

Here, $s(t)$ denotes the state of the agent at time t , $r(s^\pi(t))$ denotes the reward obtained in state $s(t)$, τ_r a discounting factor for future rewards and π the agent's policy. To achieve this, we define the following objective function which should be minimized by gradient descent on the weights from inputs to the critic:

$$E(t) := \frac{1}{2} (V^\pi(t) - z(t))^2, \quad (4)$$

where $z(t)$ represents the activity of the critic unit. By performing gradient descent on Equation (4), using a self-consistency equation for $V^\pi(t)$ from the derivative of Equation (3) and bootstrapping on the current prediction for the value (see **Supplementary Material** and Doya, 2000; Frémaux et al., 2013), we obtain the following local Hebbian three-factor learning rule that approximately minimizes the objective function (Equation 4):

$$\Delta w_j = \eta \delta(t) x_j(t) \Theta(z(t) - \theta_{\text{post}}), \quad (5)$$

where η is a learning rate, $x_j(t)$ represents the activity of the j th place cell, $\Theta(\cdot)$ the Heaviside function and θ_{post} a parameter that accounts for noise on the postsynaptic unit (see **Supplementary Material** for details). The term $\delta(t) = \dot{v}(t) + r(t) - \frac{1}{\tau_r} v(t)$ corresponds to the activity of the reward prediction error unit, acting as a neuromodulatory signal for the Hebbian plasticity between the presynaptic (x_j) and postsynaptic (z) units. To avoid explicit calculation of the derivative, we approximate $\delta(t)$ by:

$$\delta(t) \approx \left(\frac{1}{d} - \frac{1}{\tau_r} \right) v(t) - \frac{1}{d} v(t-d) + r(t). \quad (6)$$

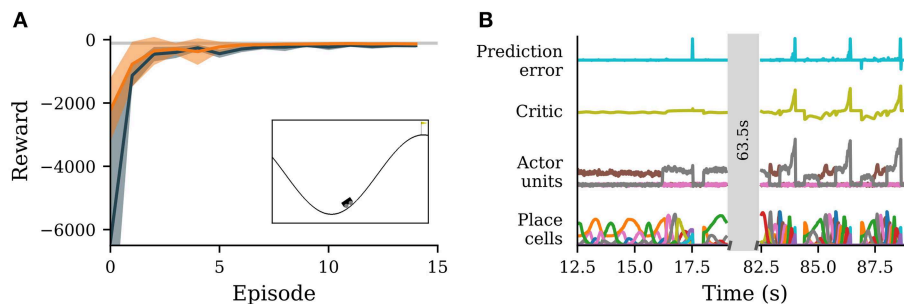


FIGURE 3 | A neuronal network simulated in NEST successfully learns to navigate in an environment with continuous states and discrete actions. **(A)** Reward obtained by the agent per episode averaged over 10 simulations with different seeds (solid orange curve). Orange band indicates \pm one standard deviation. Dark gray represents the reward obtained from Q-learning. The light gray line marks average reward per episode for which the environment is considered solved. Inset: screenshot of the environment with agent (stylized vehicle), environment with valley and two hills and goal position (yellow flag). The agent is close to a typical starting position at the trough. **(B)** Activity traces of place cells (bottom), actor units (second from bottom), critic unit (second from top) and reward prediction error unit (top). Shown are neural activities during 6.5 s early (left) and late (right) during learning. The neural network simulation was run with a real-time factor of one.

To compute the derivative we hence implement two connections from the critic to the reward-prediction error unit: one instantaneous, and one with delay $d > 0$.

As proposed by Frémaux et al. (2013), to learn an optimal policy, we exploit that the actor units follow the same dynamics as the critic. We hence apply the same learning rule to the connections between the inputs and the actor units. In order to assure that at least one actor unit is active, thus preventing a deadlock, we introduce a minimal weight for each connection between input and output units and add input noise to the actor units.

3.3.2. Mountain Car

As an example of an environment with continuous states, we consider the *MountainCar*⁸ environment. The task is to steer a toy vehicle that starts at a valley between two hills to the top of the right one (Figure 3A, inset). To make the task more challenging, the car's engine is not strong enough to reach the top in one go, so the agent needs to learn to gain momentum by swinging back and forth between the two hills. A single episode in this environment starts when the agent is placed in the valley and ends when it reaches the final position on the top of the right hill. The state of the agent is described by two continuous variables: the x-position $x(t)$ and the x-velocity $\dot{x}(t)$. The agent can choose from three different discrete actions that affect the velocity of the vehicle (accelerate left, no acceleration, accelerate right). It receives punishment (i.e., negative reward) from the environment in every step; the goal is to minimize the total punishment collected over the whole episode. Since it is challenging for a neuronal network implementation of the actor-critic architecture with exclusively excitatory synapses to learn the value function corresponding to a task with solely negative reinforcement (Potjans et al., 2011), we provide additional reward when the agent reaches the final position.

To translate the agent's current state into neuronal activity, we distribute 25 place cells evenly across the two-dimensional plane

of possible positions and velocities using the *discretize adapter* of the RMT. The actor is implemented by a WTA circuit of three units as shown in (3.3.1). The activity of these units is transformed into an action via the *argmax adapter* (3.1).

We compare the performance of our neuronal network to Q-learning (Watkins and Dayan, 1992) with function approximation via a multi-layer perceptron (see e.g., Tesauro, 1995; Mnih et al., 2013). The position and velocity of the car are projected to a population of hidden units with rectifying-linear activation function, which in turn project to three output units, encoding the estimated Q-value of each possible action. These Q-values are used by an epsilon-greedy strategy to select the next move. We use the ADAM optimizer (Kingma and Ba, 2014) and memory replay (Lin, 1993) to train the Q-function network (see **Supplementary Material** for details).

Initially, the agent explores the environment by selecting random actions. Due to the WTA circuit dynamics, a single actor neuron stays active over an extended period of time. The constant punishment gradually decreases the weights from the place cells to the corresponding actor unit, eventually leading to another actor unit becoming active (Figure 3B, left). After a while, the agent reaches the goal by performing actions that have not been significantly punished. For this task the stable nature of the WTA is advantageous, causing the agent to perform the same action repeatedly allowing efficient exploration of the state space. After the agent has found the goal once, the number of steps spent on exploring actions in the following episodes is much smaller. From the sixth episode on, the performance of the agent is already close to optimal (Figure 3A). After learning for about ten episodes, the agent's performance has converged. The value of the final state has been successfully propagated backwards over different states, leading to a ramping of activity of the critic unit from the start of an episode to the end (Figure 3B, right).

In comparison to Q-learning, the agent avoids high losses at the start of a training episode. This can most likely be traced back to two factors, which endow our agent with an advantage over Q-learning with function approximation. First, our agent starts with predefined place cells that reliably encode the position

⁸<https://gym.openai.com/envs/MountainCar-v0/>

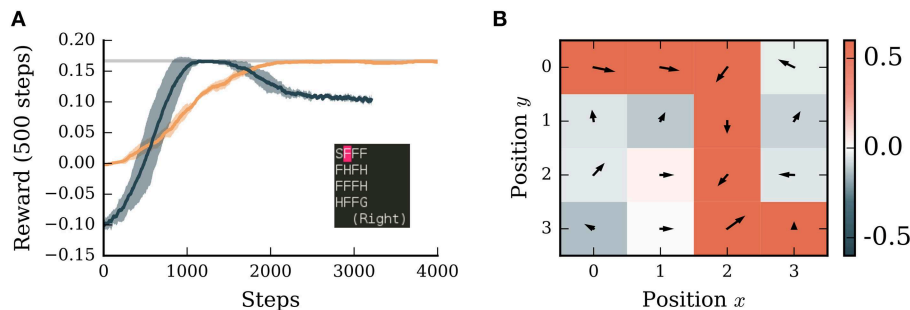


FIGURE 4 | A neuronal network simulated in NEST successfully learns to navigate in a grid world with discrete states and discrete actions. **(A)** Orange curve: average reward collected by the agent over the next 500 steps averaged over 5 simulations. Dark blue curve: performance of Potjans et al. (2011) model averaged over 5 simulations. Shaded bands indicate \pm one standard deviation. Gray line: theoretical optimum. Inset: screenshot of the environment with start state (S), frozen states (F), holes (H), and goal state (G). The position of the agent is indicated in pink. **(B)** The learned policy and value map of the environment. Red colors indicate positive, blue colors negative values. Arrows indicate the preferred direction of movement. The neural network simulation was run with a real-time factor of two.

in state space and it only has to learn to appropriately combine the activities of these place cells. In contrast, Q-learning starts from a completely blank slate, with no prior knowledge about the input space. It would be incorrect to conclude from this that place cells are generally the superior strategy: manually-defined place cells become infeasible in high-dimensional state spaces as their number increases exponentially in the number of input-space dimensions, whereas Q-learning with function approximation can be scaled to very high-dimensional input spaces (see e.g., Mnih et al., 2013). The second advantage of our agent are long transients in action selection. Before learning the correct sequence of actions, the agent tends to explore a single action for an extended period of time (see trajectories of actor units, **Figure 3B**, left), whereas Q-learning changes action often. For this particular environment sticking to one action for an extended period of time, especially during the early phases of learning, is advantageous as the final strategy involves few action changes (**Figure 3B**, right). This disadvantage can most likely be attenuated by using frame skipping or similar methods (cf. Mnih et al., 2013).

3.3.3. Frozen Lake

As a second application illustrating the use of the toolchain for discrete environments, we train the same network model on the *FrozenLake*⁹ environment. This consists of a discrete set of 16 states arranged in a four-by-four grid (**Figure 4A**, inset). Each state is either a start state (S), a goal state (G), a hole (H), or a frozen state (F). From the start position, the agent has to reach the rewarded state by navigating over the frozen states without falling into holes which reset the agent to the starting position. In each step the agent can choose from four different actions: move west, move north, move east and move south. Usually, the tiles are “slippery,” i.e., there is a chance that a random action is executed irrespective of the action chosen by the agent. However, to simplify learning for demonstration purposes we turn this feature off. Upon reaching the goal the agent receives a reward of magnitude one. Since the optimal path involves six steps from start to goal, the theoretical optimal reward per step is ~ 0.16 .

To encourage exploration the agent receives a small punishment in each state and, additionally, to speed up learning the agent is punished for falling into holes.

Unlike in the continuous *MountainCar* environment, the tuning curves of place cells do not overlap in the discrete case, leading to sharp transitions in the network activity. This leads to severe issues for associating values and actions with the respective states. To address this problem we introduced a simple eligibility trace by evaluating the activity of the pre- and post synaptic units in the learning rule with a small delay δt (see **Supplementary Material**). With this addition, the network model is able to find the optimal solution for this task within roughly 2,000 steps (**Figure 4A**). It also learns to associate holes with punishment and frozen states with reward if they are on the path to the goal (**Figure 4B**). Although there are two possible paths to the goal, the agent prefers the path with fewer corners, likely as a consequence of the WTA circuit which tends to select the same action repeatedly.

We compare the performance of our algorithm to an adapted spiking neural network model of the basal ganglia implementing reinforcement learning (Potjans et al., 2011; Jitsev et al., 2012). Learning in this algorithm is faster than our implementation and reaches the optimal solution after only 1,000 steps (**Figure 4A**). However, the performance of the spiking model drops after 2,000 steps to a sub-optimal value. As this model relies on a very high discount factor ($\gamma = 0.99$), which is close to ‘infinite horizon,’ the values of the states saturate in the vicinity to the goal. This can lead to a low contrast of preferred actions in those states and therefore to a sub-optimal policy. To resolve this issue is beyond the scope of this manuscript (see Kato and Morita, 2016 for an investigation of such matters), but underlines the importance of comparing alternative models on the same task. Only through such activities can we identify the strengths and weaknesses of different functional hypotheses and thus make more rapid progress in the field.

4. CONCLUSION

In this manuscript, we have argued that standardized benchmarks are of critical importance to compare and improve

⁹<https://gym.openai.com/envs/FrozenLake-v0/>

functional neural network models. Moreover, to investigate the characteristics of the neural circuits that allow agents to operate autonomously in their environments and learn appropriate behaviors, simulation infrastructure must enable closed-loop interaction between agent and environment.

To make such a set of closed-loop benchmarks available to the computational neuroscience community, we have developed a toolchain that closes the loop between the OpenAI Gym and neural network simulators implementing the MUSIC interface, notably NEST and NEURON. We demonstrated the functionality of the toolchain by implementing an actor-critic architecture in NEST and evaluating its performance on two different environments. The performance of the network quickly reached near-optimal performance on these two tasks.

Compared to creating customized environments within the framework of a neuronal simulator, using readily available, well-tested tools is considerably easier (and thus faster) for the researcher, often computationally more efficient, and most importantly, supports reproducible science. In addition, having the OpenAI Gym environments as common benchmarks in both fields encourages comparison between traditional machine learning and biologically plausible implementations. In contrast to models presented in previous studies, our toolchain makes it easy for other researchers to extend our implementation of an actor-critic architecture to other environments, replace neuron models or explore alternative learning rules. The simulation and visualization scripts to reproduce the results presented for the network model described here are publicly available¹⁰, and so can serve as a starting point for more complex models. In addition a dedicated tutorial introduces the toolchain step-by-step using NEST as an example simulator¹¹.

While the toolchain currently only supports the OpenAI Gym, the extension to other toolkits is simple due to a modular design of the wrapper. The RMT can be found on GitHub and is available under the GPLv3. The OpenAI Gym ZeroMQ wrapper is also available via GitHub under the MIT license. A complementary development to the work presented here is provided by SPORE, a framework for reward-based learning with spiking neurons in the NEST simulator¹². It provides support for synapse models with time-driven updates, additional support

for recording and evaluating traces of neuronal state variables and introduces MUSIC ports for communicating rewards to a running simulation.

With the work presented here we enable researchers to build more easily upon previous studies and evaluate novel models. We hope this boosts the progress in computational neuroscience in uncovering the biophysical mechanisms involved in autonomous behavior and learning.

DATA AVAILABILITY

The datasets generated for this study can be found on GitHub (<https://github.com/INM-6/closed-loop-learning-in-autonomous-agents>).

AUTHOR CONTRIBUTIONS

This study was conceived and designed by JJ, PW, and AM. The toolchain and tested models were implemented by JJ and PW. The simulations were carried out and analyzed by JJ and PW. The manuscript was written jointly by all authors.

FUNDING

We acknowledge partial support by the German Federal Ministry of Education through our German-Japanese Computational Neuroscience Project (BMBF Grant 01GQ1343), the Helmholtz Alliance through the Initiative and Networking Fund of the Helmholtz Association and the Helmholtz Portfolio theme Supercomputing and Modeling for the Human Brain, and the European Union's Horizon 2020 Framework Programme for Research and Innovation under Specific Grant Agreement Nos. 720270 and 785907 (Human Brain Project SGA1 and SGA2).

ACKNOWLEDGMENTS

We warmly thank our colleagues in the NEST development team for constructive discussions and support.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fncom.2019.00046/full#supplementary-material>

¹⁰<https://github.com/INM-6/closed-loop-learning-in-autonomous-agents>

¹¹<https://github.com/INM-6/nestrl-tutorial/>

¹²<https://github.com/IGITUGraz/spore-nest-module>

REFERENCES

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.
- Aswolinskiy, W., and Pipa, G. (2015). RM-SORN: a reward-modulated self-organizing recurrent neural network. *Front. Comput. Neurosci.* 9:36. doi: 10.3389/fncom.2015.00036
- Baladron, J., and Hamker, F. H. (2015). A spiking neural network based on the basal ganglia functional anatomy. *Neural Netw.* 67, 1–13. doi: 10.1016/j.neunet.2015.03.002

- Bekolay, T., Bergstra, J., Hunsberger, E., DeWolf, T., Stewart, T. C., Rasmussen, D., et al. (2013). Nengo: a python tool for building large-scale functional brain models. *Front. Neuroinform.* 7:48. doi: 10.3389/fninf.2013.00048
- Bellemare, M., Naddaf, Y., Veness, J., and Bowling, M. (2015). "The arcade learning environment: an evaluation platform for general agents," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*.
- Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H., et al. (2007). Greedy layer-wise training of deep networks. *Adv. Neural Inform. Process. Syst.* 19:153.
- Bower, J. M., and Beeman, D. (2007). GENESIS (simulation environment). *Scholarpedia* 2:1383. doi: 10.4249/scholarpedia.1383

- Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., et al. (2016). OpenAI Gym. *ArXiv e-prints. arXiv preprint arXiv:1606.01540*.
- Carnevale, T., and Hines, M. (2006). *The NEURON Book*. Cambridge: Cambridge University Press.
- Collobert, R., Bengio, S., and Mariéthoz, J. (2002). *Torch: A Modular Machine Learning Software Library*. Technical report, Idiap.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). "ImageNet: a large-scale hierarchical image database," in *CVPR09*.
- Djurfeldt, M., Hjorth, J., Eppler, J. M., Dudani, N., Helias, M., Potjans, T. C., et al. (2010). Run-time interoperability between neuronal simulators based on the music framework. *Neuroinformatics* 8, 43–60. doi: 10.1007/s12021-010-9064-z
- Doya, K. (2000). Reinforcement learning in continuous time and space. *Neural Comput.* 12, 219–245. doi: 10.1162/089976600300015961
- Echeverria, G., Lassabe, N., Degroote, A., and Lemaignan, S. (2011). "Modular open robots simulation engine: Morse," in *2011 IEEE International Conference on Robotics and Automation* (Citeseer), 46–51.
- Eppler, J. M., Helias, M., Muller, E., Diesmann, M., and Gewaltig, M. (2009). PyNEST: a convenient interface to the NEST simulator. *Front. Neuroinform.* 2:12. doi: 10.3389/neuro.11.012.2008
- Falotico, E., Vannucci, L., Ambrosano, A., Albanese, U., Ulbrich, S., Vasquez Tieck, J. C., et al. (2017). Connecting artificial brains to robots in a comprehensive simulation framework: the neurobotics platform. *Front. Neurobot.* 11:2. doi: 10.3389/fnbot.2017.00002
- Frémaux, N., Sprekeler, H., and Gerstner, W. (2010). Functional requirements for reward-modulated spike-timing-dependent plasticity. *J. Neurosci.* 30, 13326–13337. doi: 10.1523/JNEUROSCI.6249-09.2010
- Frémaux, N., Sprekeler, H., and Gerstner, W. (2013). Reinforcement learning using a continuous time actor-critic framework with spiking neurons. *PLoS Comput. Biol.* 9:e1003024. doi: 10.1371/journal.pcbi.1003024
- Friedrich, J., and Lengyel, M. (2016). Goal-directed decision making with spiking neurons. *J. Neurosci.* 36, 1529–1546. doi: 10.1523/JNEUROSCI.2854-15.2016
- Friedrich, J., Urbanczik, R., and Senn, W. (2014). Code-specific learning rules improve action selection by populations of spiking neurons. *Int. J. Neural Syst.* 24:1450002. doi: 10.1142/S0129065714500026
- Gerstner, W., and Naud, R. (2009). How good are neuron models? *Science* 326, 379–380. doi: 10.1126/science.1181936
- Gewaltig, M.-O., and Diesmann, M. (2007). NEST (NEural Simulation Tool). *Scholarpedia* 2:1430. doi: 10.4249/scholarpedia.1430
- Goodman, D. F., and Brette, R. (2009). The brian simulator. *Front. Neurosci.* 3, 192–197. doi: 10.3389/neuro.01.026.2009
- Hahne, J., Dahmen, D., Schuecker, J., Frommer, A., Bolten, M., Helias, M., et al. (2017). Integration of continuous-time dynamics in a spiking neural network simulator. *Front. Neuroinform.* 11:34. doi: 10.3389/fninf.2017.00034
- Hazan, H., Saunders, D. J., Khan, H., Patel, D., Sanghavi, D. T., Siegelmann, H. T., and Kozma, R. (2018). Bindsnet: a machine learning-oriented spiking neural networks library in python. *Front. Neuroinform.* 12:89. doi: 10.3389/fninf.2018.00089
- Hintjens, P. (2013). *ZeroMQ: Messaging for Many Applications*. O'Reilly Media, Inc.
- Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-R., Jaitly, N., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. *IEEE Signal Process. Mag.* 29, 82–97. doi: 10.1109/MSP.2012.2205597
- Hinton, G. E., and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science* 313, 504–507. doi: 10.1126/science.1127647
- Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.
- Human Brain Project (2014). *Project Website*. Available online at: <http://www.humanbrainproject.eu>
- Izhikevich, E. M. (2007). Solving the distal reward problem through linkage of STDP and dopamine signaling. *Cereb. Cortex* 17, 2443–2452. doi: 10.1093/cercor/bhl152
- James, B., Olivier, B., Frédéric, B., Pascal, L., and Razvan, P. (2010). "Theano: a CPU and GPU math expression compiler," in *Proceedings of the Python for Scientific Computing Conference (SciPy)*.
- Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., et al. (2014). Caffe: convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*. doi: 10.1145/2647868.2654889
- Jitsev, J., Morrison, A., and Tittgemeyer, M. (2012). "Learning from positive and negative rewards in a spiking neural network model of basal ganglia," in *The 2012 International Joint Conference on Neural Networks (IJCNN)* (IEEE), 1–8.
- Jordan, J., Ippen, T., Helias, M., Kitayama, I., Sato, M., Igarashi, J., et al. (2018). Extremely scalable spiking neural network simulation code: from laptops to exascale computers. *Front. Neuroinform.* 12:2. doi: 10.3389/fninf.2018.00034
- Kato, A., and Morita, K. (2016). Forgetting in reinforcement learning links sustained dopamine signals to motivation. *PLoS Comput. Biol.* 12:e1005145. doi: 10.1371/journal.pcbi.1005145
- Kingma, D. P., and Ba, J. (2014). Adam: a method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Koenig, N., and Howard, A. (2004). "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE Cat. No. 04CH37566), Vol. 3 (IEEE), 2149–2154.
- Krizhevsky, A., and Hinton, G. (2009). *Learning Multiple Layers of Features From Tiny Images*. Technical Report 4, University of Toronto.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 1097–1105.
- Kunkel, S., Schmidt, M., Eppler, J. M., Plesser, H. E., Masumoto, G., Igarashi, J., Ishii, S., et al. (2014). Spiking network simulation code for petascale computers. *Front. Neuroinform.* 8:78. doi: 10.3389/fninf.2014.00078
- LeCun, Y. (1998). The MNIST Database of Handwritten Digits. Available online at: <http://yann.lecun.com/exdb/mnist>
- Lin, L.-J. (1993). *Reinforcement Learning for Robots Using Neural Networks*. Technical report, Carnegie-Mellon Univ Pittsburgh PA School of Computer Science.
- Michel, O. (2004). Cyberbotics Ltd. Webots™ : professional mobile robot simulation. *Int. J. Adv. Robot. Syst.* 1:5. doi: 10.5772/5618
- Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., et al. (2016). "Asynchronous methods for deep reinforcement learning," in *Int. Conf. Mach. Learn.* 1928–1937.
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., et al. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., et al. (2015). Human-level control through deep reinforcement learning. *Nature* 518, 529–533. doi: 10.1038/nature14236
- Moren, J., Sugimoto, N., and Doya, K. (2015). Real-time utilization of system-scale neuroscience models. *J. Jap. Neural Netw. Soc.* 22, 125–132. doi: 10.3902/jnns.22.125
- Moser, E. I., Kropff, E., and Moser, M.-B. (2008). Place cells, grid cells, and the brain's spatial representation system. *Annu. Rev. Neurosci.* 31, 69–89. doi: 10.1146/annurev.neuro.31.061307.090723
- Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., et al. (2017). Automatic Differentiation in Pytorch.
- Potjans, W., Diesmann, M., and Morrison, A. (2011). An imperfect dopaminergic error signal can drive temporal-difference learning. *PLoS Comput. Biol.* 7:e1001133. doi: 10.1371/journal.pcbi.1001133
- Potjans, W., Morrison, A., and Diesmann, M. (2009). A spiking neural network model of an actor-critic learning agent. *Neural Comput.* 21, 301–339. doi: 10.1162/neco.2008.08-07-593
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., et al. (2009). "ROS: an open-source Robot Operating System," in *ICRA Workshop on Open Source Software*, Vol. 3 (Kobe).
- Raina, R., Madhavan, A., and Ng, A. Y. (2009). "Large-scale deep unsupervised learning using graphics processors," in *Proceedings of the 26th Annual International Conference on Machine Learning (ACM)*, 873–880.
- Rasmussen, D., and Eliasmith, C. (2014). "A neural model of hierarchical reinforcement learning," in *CogSci*.
- Rombouts, J. o., Bohte, S. M., and Roelfsema, P. R. (2015). How attention can create synaptic tags for the learning of working memories in sequential tasks. *PLoS Comput. Biol.* 11:e1004060. doi: 10.1371/journal.pcbi.1004060

- Rueckert, E., Kappel, D., Tanneberg, D., Pecevski, D., and Peters, J. (2016). Recurrent spiking networks solve planning tasks. *Sci. Rep.* 6:21142. doi: 10.1038/srep21142
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* 115, 211–252. doi: 10.1007/s11263-015-0816-y
- Sanz Leon, P., Knock, S. A., Woodman, M. M., Domide, L., Mersmann, J., McIntosh, A. R., et al. (2013). The virtual brain: a simulator of primate brain network dynamics. *Front. Neuroinform.* 7:10. doi: 10.3389/fninf.2013.00010
- Sheik S., and Liu, Q. (2019). *SINABS - A Spiking Deep Neural Network Inference Emulator*. doi: 10.5281/zenodo.3218477
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., et al. (2016). Mastering the game of go with deep neural networks and tree search. *Nature* 529, 484–489. doi: 10.1038/nature16961
- Sutton, R. S., and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. Adaptive Computation and Machine Learning. The MIT Press.
- Tesauro, G. (1995). Temporal difference learning and TD-gammon. *Commun. ACM* 38, 58–68.
- Todorov, E., Erez, T., and Tassa, Y. (2012). “Mujoco: a physics engine for model-based control,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5026–5033.
- Urbanczik, R., and Senn, W. (2009). Reinforcement learning in populations of spiking neurons. *Nat. Neurosci.* 12:250. doi: 10.1038/nn.2264
- Vasilaki, E., Frémaux, N., Urbanczik, R., Senn, W., and Gerstner, W. (2009). Spike-based reinforcement learning in continuous state and action space: when policy gradient methods fail. *PLoS Comput. Biol.* 5:e1000586. doi: 10.1371/journal.pcbi.1000586
- Watkins, C. J., and Dayan, P. (1992). Q-learning. *Mach. Learn.* 8, 279–292.
- Watkins, C. J. C. H. (1989). *Learning from delayed rewards* (Ph.D. thesis). University of Cambridge.
- Weidel, P., Djurfeldt, M., Duarte, R. C., and Morrison, A. (2016). Closed loop interactions between spiking neural network and robotic simulators based on music and ros. *Front. Neuroinform.* 10:31. doi: 10.3389/fninf.2016.00031
- Wils, S., and De Schutter, E. (2009). STEPS: modeling and simulating complex reaction-diffusion systems with Python. *Front. Neuroinform.* 3:15. doi: 10.3389/neuro.11.015.2009
- Wilson, M. (2002). Six views of embodied cognition. *Psychon. Bull. Rev.* 9, 625–636. doi: 10.3758/BF03196322

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Jordan, Weidel and Morrison. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



A Biomimetic Control Method Increases the Adaptability of a Humanoid Robot Acting in a Dynamic Environment

Marie Claire Capolei^{1*}, Emmanouil Angelidis², Egidio Falotico³, Henrik Hautop Lund¹ and Silvia Tolu^{1*}

¹ Automation and Control Group, Department of Electrical Engineering, Technical University of Denmark, Copenhagen, Denmark, ² Landesforschungsinstitut des Freistaats Bayern, An-Institut, Technical University of Munich, Munich, Germany, ³ The BioRobotics Institute, Scuola Superiore Sant'Anna, Pisa, Italy

OPEN ACCESS

Edited by:

Florian Röhrbein,
Technical University of Munich,
Germany

Reviewed by:

Juyang Weng,
Michigan State University,
United States
Qiuxuan Wu,
Hangzhou Dianzi University, China

*Correspondence:

Marie Claire Capolei
macca@elektro.dtu.dk
Silvia Tolu
stolu@elektro.dtu.dk

Received: 29 January 2019

Accepted: 12 August 2019

Published: 28 August 2019

Citation:

Capolei MC, Angelidis E, Falotico E, Lund HH and Tolu S (2019) A Biomimetic Control Method Increases the Adaptability of a Humanoid Robot Acting in a Dynamic Environment. *Front. Neurobot.* 13:70. doi: 10.3389/fnbot.2019.00070

One of the big challenges in robotics is to endow agents with autonomous and adaptive capabilities. With this purpose, we embedded a cerebellum-based control system into a humanoid robot that becomes capable of handling dynamical external and internal complexity. The cerebellum is the area of the brain that coordinates and predicts the body movements throughout the body-environment interactions. Different biologically plausible cerebellar models are available in literature and have been employed for motor learning and control of simplified objects. We built the canonical cerebellar microcircuit by combining machine learning and computational neuroscience techniques. The control system is composed of the adaptive cerebellar module and a classic control method; their combination allows a fast adaptive learning and robust control of the robotic movements when external disturbances appear. The control structure is built offline, but the dynamic parameters are learned during an online-phase training. The aforementioned adaptive control system has been tested in the Neuro-robotics Platform with the virtual humanoid robot iCub. In the experiment, the robot iCub has to balance with the hand a table with a ball running on it. In contrast with previous attempts of solving this task, the proposed neural controller resulted able to quickly adapt when the internal and external conditions change. Our bio-inspired and flexible control architecture can be applied to different robotic configurations without an excessive tuning of the parameters or customization. The cerebellum-based control system is indeed able to deal with changing dynamics and interactions with the environment. Important insights regarding the relationship between the bio-inspired control system functioning and the complexity of the task to be performed are obtained.

Keywords: biomimetic, cerebellar control, motor learning, humanoid robot, adaptive system, forward model, bio-inspired, neurobotics

1. INTRODUCTION

Controlling a robotic system that operates in an uncertain environment can be a difficult task if the analytical model of the system is not accurate. Models are the most essential tools in robotic control (Francis and Wonham, 1976), however, modeling errors are frequently inevitable in complex robots, for instance humanoids and soft robots. Such redundant modern robots are mechanically complex and often interacts with unstructured dynamical environments (Nakanishi et al., 2008; Nguyen-Tuong et al., 2009). Traditional hand-crafted models and standard physics-based modeling techniques do not sufficiently take into account all the unknown nonlinearities and complexities that these system present. This lack consequentially leads to a reduced tracking accuracy or, in the worst case, to unstable null-space behavior.

Modern autonomous and cognitive robots are requested to adapt not only the decisions but also the forces exerted in any varying condition and environment. The selected movement can not be executed properly if the robot does not adjust the forces according to the changing dynamics. Because of this, modern learning control methods should automatically generate model based on sensor data streams, so that the robot is not a closed entity, but a system that interacts, and evolves through the interaction with a dynamic environment.

In this paper, we intend to design an adaptive learning algorithm to control the movements of a complex nonlinear dynamical system. In particular, we assume that: the Jacobian poorly describes the actual system; the robot interacts with one or more unmodeled external objects; the sensor-actuator system is distributed and not all the states are observable or can be describe with parametric function designed off-line; the action/state space is continuous and high-dimensional. The control system should solve the inverse dynamics control problem of a multiple-joint robotic system affected by static and dynamic external disturbances during the execution of a repeated task. The controller is envisioned to reduce the tracking accuracy of each actuator through force-based control input.

In early days of adaptive self-tuning control, models were learned by fitting open parameters of predefined parametric models (Atkeson et al., 1986; Annaswamy and Narendra, 1989; Wittenmark, 1995; Khalil and Dombre, 2002). Although this method had great success in system identification and adaptive control techniques (Ljung, 2007), the estimation of the open parameters can lead to several problems, such as: slow adaptation; unmodeled behavior and persistent excitation issue (Narendra and Annaswamy, 1987); inconsistency of the estimated physical parameters (Ting et al., 2006); unstable reaction to high estimation error. In recent years, non-parametric approach has been shown to be an efficient tool in the resolution and prevention of the aforementioned problems thanks to the adaptation of the model to the data complexity (Nguyen-Tuong and Peters, 2011), and several methods have been proposed (Farrell and Polycarpou, 2006), such as neural networks (Patino et al., 2002), and statistical methods (Kocijan et al., 2004; Nakanishi and Schaal, 2004; Nakanishi et al., 2005).

In the eighties, Narendra's research group at Yale University exploited the adaptability of artificial neural networks (ANNs) to identify and control nonlinear dynamical systems (Narendra and Mukhopadhyay, 1991a,b, 1997; Narendra and Parthasarathy, 1991). Their experiments showed that the versatility of the ANNs resulted beneficial for controlling the different behaviors that characterize complex dynamical systems. Although the robustness of the classic parametric method in most of the control scenarios, ANNs were largely used in adaptive control to overcome uncertainties, unmodeled nonlinearities and to handle more complex state space systems (Glanz et al., 1991; Sontag, 1992; Zhang et al., 2000; Patino et al., 2002; He et al., 2016, 2018). As matter of fact, the non-linear components and the layered structure that distinguish the ANNs facilitate the mapping and constrain the effects of nonlinearities. Furthermore, the on-line adjustment of the parameters respect to the input-output relationship without any strict structural parameterization results advantageous for adapting to time-dependent changes.

In the Eighties thanks to the extended application of ANNs in robotics, Juyang Weng introduced the Autonomous Mental Development approach (AMD) to artificial intelligence (Weng et al., 1999a; Weng and Hwang, 2006). Weng theories were mainly inspired by how the biological systems efficiently calibrate their movements under internal and environmental changes. Accordingly to AMD the robot have to be embodied in the environment, and its processing is not preprogrammed but is the result of the continuous and real-time interaction within the two systems (Weng et al., 1999b, 2000; Weng, 2002). Respect to classic parametric approaches, the developing artificial agent creates and adapts models describing itself and its relation with the environment rather than learning and estimating parameters of a mathematical model built off-line. These theories found large application for high level cognition tasks (see Vernon et al., 2007 for a review) but were also applied to low level control in visually-guided robots (Metta et al., 1999; Ugur et al., 2015; Luo et al., 2018).

With the aim of mimicking artificially the motor efficiency of the biological system, James S. Albus proposed a neural network-based learning algorithm for robotic controller based on theories of central nervous system (CNS) structure and function: the "cerebellar model articulation controller," commonly known as CMAC module (Albus, 1972). Several studies in literature demonstrated that, the anatomy and physiology of the cerebellum is suitable for the acquisition, development, storage and use of the internal models describing the interaction within body and environment (Wolpert et al., 1998). Moreover, the cerebellum is composed by separated regions which functionality relies both on the internal structure of the circuit and on the connection with other CNS areas (Houk and Wise, 1995; Caligiore et al., 2017): each region receives both the desired movements from the cortex and the sensory information from tendons, joints and muscles spindles and elaborates a signal that corrects whereas other CNS region are lacking. As matter of fact, subjects affected by cerebellum damage often present motor deficit, such as uncoordinated and ballistic multiple-joint movements (Schmahmann, 2004). For this reason in the last

decades, scientists tried to explain the roles of the cerebellum in motor control, especially its contribution to sensory acquisition and timing and its involvement in the prediction of the sensory consequences of action. Moreover, this adaptive control nature motivated several researchers toward a deeper understanding of the cerebellum for robotics application.

Two main research lines born since Marr and Albus proposed the first artificial cerebellum-like network as pattern-classifier for controlling a robotic manipulator (Marr, 1969; Albus, 1972): the first research line focuses on purely industrial application and has as major representative W. Thomas Miller; the second research line, mainly represented by Mitsuo Kawato, deep-rooted in neuroscience and kept investigating on the biological evidence of the cerebellum structure and functionalities in relation to other CNS areas (Kawato et al., 1987; Kawato, 1999).

Miller applied the CMAC module in a closed loop vision-based controller to solve the forward mapping with direct modeling (Miller, 1987). Although the advantages, such as the rapid algorithmic computation based on least-mean-square training and the fast incremental learning, this approach lack of generalization and is sensitive to noise and large error (Miller et al., 1990). Over the years, researchers have been focusing on solving these drawbacks and the CMAC module has been mostly used as non-linear function approximator to boost the tracking accuracy of the adaptive controller and mitigate the effects of the approximation errors (Lin and Chen, 2007; Chen, 2009; Guan et al., 2018; Jiang et al., 2018). Although the promising results obtained by these applications of the CMAC network, this industrial research line did not completely exploit the overall capabilities and components of the cerebellum. It is worthy to note that the CMAC module mimic the cerebellar circuit only at the granular-purkinje level, for this reason only the mapping and classification functionalities are exploited.

The neuroscientific research line has been investigating mainly on the layered structure of the cerebellar circuit proposing several synaptic plasticity models (Luque et al., 2011, 2014, 2016; Casellato et al., 2015; D'Angelo et al., 2016; Antonietti et al., 2017), network models (Chapeau-Blondeau and Chauvet, 1991; Buonomano and Mauk, 1994; Ito, 1997; Mauk and Donegan, 1997; Yamazaki and Tanaka, 2007; Dean et al., 2010), adaptive linear filter model (Fujita, 1982; Barto et al., 1999; Fujiki et al., 2015), and combination of both (Tolu et al., 2012, 2013). These cerebellar-like models were embedded into bio-inspired control architectures to analyze how the cerebellum adjusts the output of the descending motor system of the brain during the generation of movements (Kawato et al., 1987; Ito, 2008), and how it predicts the action, minimizes the sensory discrepancy and cancels the noise (Nowak et al., 2007; Porrill and Dean, 2007). The experiments regarded the generation of voluntary movements with both simulated and real robots, e.g., eye blinking classical conditioning (Antonietti et al., 2017), vestibulo-ocular task (Casellato et al., 2014), the gaze stabilization (Vannucci et al., 2016), and perturbed arm reaching task operating in closed-loop (Garrido Alcazar et al., 2013; Tolu et al., 2013; Luque et al., 2016; Ojeda et al., 2017). From the analysis of the literature, it then emerged that research groups have treated the robots as stand-alone systems without interactions with the

environment, while the real world is more complex and every external interaction counts. It is worth mentioning that the previous works have been employed for motor learning and control of simplified objects.

In this paper we present a robotic control architecture to overcome modeling error and to constrain the effects of uncertainties and external disturbances. The proposed controller is composed of a static component based on a classic feedback control methods, and of an adaptive decentralized neural network that mimic the functionality and morphology of the cerebellar circuit. The cerebellar-like module add feed-forward corrective torque to the feedback controller action (Ito, 1984; Miyamoto et al., 1988). A non-parametric nonlinear function approximation algorithm have been employed to map on-line and to reduce the high dimensional and redundant input space. The algorithm creates the internal model describing the interaction within system and environment. This model is kept under development throughout the execution of the task. The neural network mimic the composition of the cerebellar microcircuit. The layered structure of the network constrains the effects of nonlinearities and external perturbations. The network weights are based on non-linear and multidimensional learning rules that mimic the cerebellar synaptic plasticities (Garrido Alcazar et al., 2013; Luque et al., 2014).

This manuscript extends the previous works under three main aspects: 1. cerebellar-like network topology and input data; 2. feedback control-input; 3. dynamic control under external changing conditions. With the aim at giving more insights into the capacity of the cerebellum of generating control terms in the framework of accurate control tasks, the following research questions come naturally to mind: can a control system be generalized to control robotic agents by endowing them with adaptive capabilities? Can accurate and smooth actions in a dynamic environment be performed by the extrapolation of valuable sensory-motor information from heterogeneous dynamical stimuli? Does this sensory-motor information extrapolation facilitate the motor prediction and adaptation in changing conditions? The tests were carried out in the Neuro-robotics Platform (Falotico et al., 2017) with the virtual humanoid robot iCub. The robot arm has to follow a planned movement overcoming the disturbances provoked by a table attached to the hand and a ball running on it. A similar example was solved by employing a conventional control law together with computer vision techniques (Awtar et al., 2002; Levinson et al., 2010). However, this approach assumes a fixed robot morphology defined and described before running the experiment, and there is no run-time adaptation to the “biological changes” as we see in human beings. Balancing a table with a ball running on it is a relevant example of how humans learn to calibrate, coordinate, and adapt their movements, hence, we investigate how robots can achieve this task following the biological approach. Probst et al. (2012) also followed the biological approach; they tackled the problem taking into account the dynamics of the system, four different forces are found by means of a liquid state machine and applied in four different points of the table to achieve the balancing task. A supervised learning rule is used for the training step, which

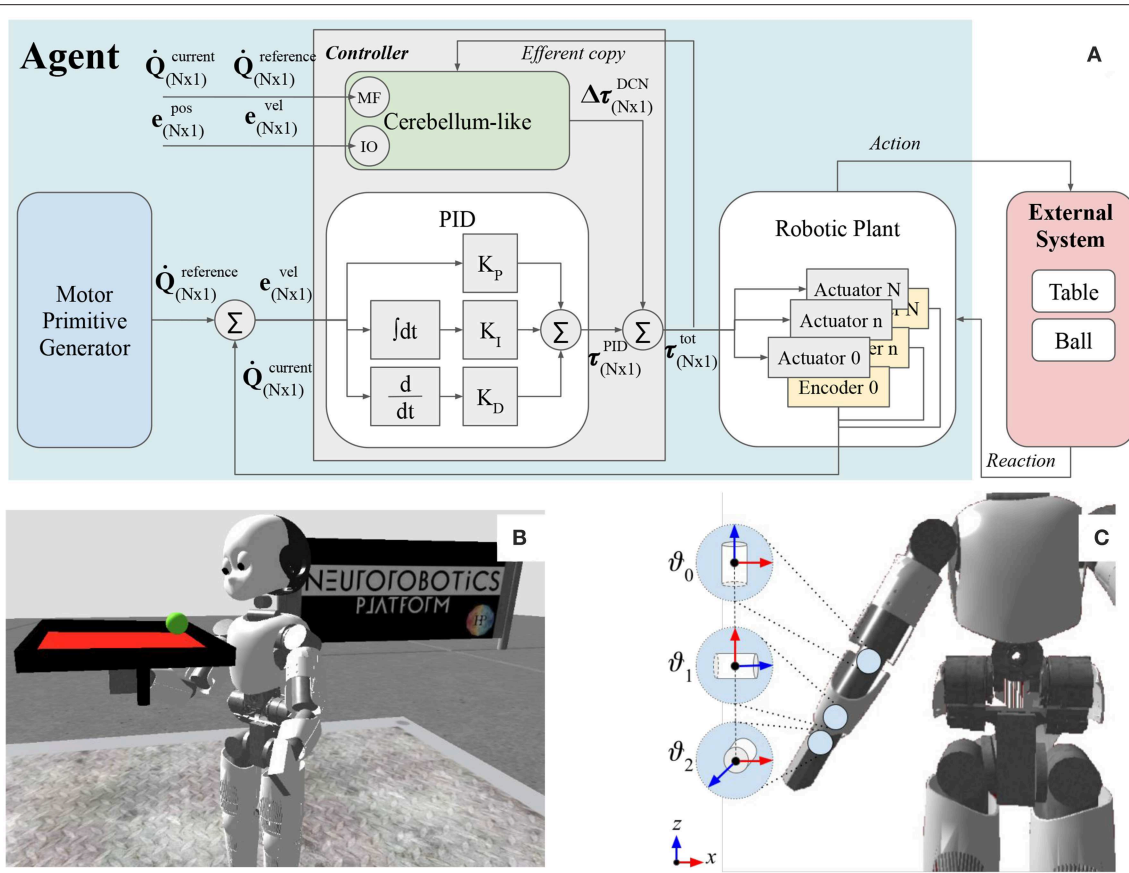


FIGURE 1 | (A) The figure illustrates the main components of the functional architecture scheme and the link with the artificial robot agent and the external system. **(B)** The humanoid Icube holding the table-ball system in the simulation environment NRP. **(C)** Three controlled joints: wrist prosup θ_0 , wrist yaw θ_1 , wrist pitch θ_2 .

concludes that after 2,500 s no further improvement of the performance is obtained.

Hence, the main advantages of our model are: the low amount of (sometimes implausible) prior information for the control, a fast reactive robotic control system, an on-line self-adaptive learning system. Thanks to these features the robot can perform a determinate physical task and adapt to changing conditions. In conclusion, this approach introduces a fast and flexible control architecture that can be applied to different robotic platforms without any/excessive customization.

In the first section that follows, we present the control architecture, the adopted cerebellar-like model and the description of the method. In the second section, we report the experimental setup as well as the results of the comparison study of four control system approaches including the respective analysis. Finally, we will discuss the main findings of the study correlating them to previous literature.

2. MATERIALS AND METHODS

In this section, we present our bio-inspired approach to solve the problem of controlling the right arm of the ICub humanoid robot despite the occurrence of an external perturbation. The

experiment consists of a simulated humanoid robot that executes a requested movement using three controlled joints of the right arm. During the simulation, a ball is launched on the table that is attached to the robot's right hand; the ball is free to roll on the table, as illustrated in **Figure 1B**. The movements of the ball are provoked by the shaking of the robot arm and consequentially of the table. The key information about the external system components (e.g., the ball and table) are reported in **Table 1**.

The proposed control architecture (**Figure 1A**) is composed of three main building blocks: the *robotic plant*, which is the physical structure (section 2.1); the *motor primitive generator*, which is responsible of the trajectory generation (section 2.2); the *controller*, which elaborates the torque commands to move each motor to the desired set point (section 2.3).

2.1. Robotic Plant

The Icube humanoid robot is 104 cm tall and it is equipped with a large variety of sensors (such as gyroscopes, accelerometers, F/T sensors, encoders, two digital cameras) and 53 actuated joints that move the waist, head, eyes, legs, arms, and hands. During the experimental tests, eight revolute joints of the right arm were actuated: four joints were kept constant to maintain the arm up (e.g., elbow, shoulder roll, shoulder yaw, and shoulder

TABLE 1 | External system features.

	Mass [Kg]	Volume [m ³]	Static friction coefficient	Dynamic friction coefficient
Ball	0.01	6.54×10^{-5}	0.02	0.01
Table top	0.1	9×10^{-4}	0.01	0.01

pitch), and three joints were controlled in effort by the proposed control system (namely wrist prosup, wrist yaw and wrist pitch). The axis orientation of the controlled actuators are illustrated in **Figure 1C**. Additional information about the actuated joints are reported in **Table 2**. In this work, we used the encoder to only read the state of the controlled joints (e.g., angular position, and velocity) and save it in the process variables,

$$\mathbf{Q}_{N \times 1}^c(t) = \begin{bmatrix} \vartheta_{c,0}(t) \\ \dots \\ \vartheta_{c,N}(t) \end{bmatrix} \text{ where } N = 2, \quad (1)$$

$$\dot{\mathbf{Q}}_{N \times 1}^c(t) = \begin{bmatrix} \dot{\vartheta}_{c,0}(t) \\ \dots \\ \dot{\vartheta}_{c,N}(t) \end{bmatrix} \text{ where } N = 2, \quad (2)$$

2.2. Motor Primitive Generator

The motor primitive generator plans the trajectory for each actuated joint and communicates the reference value to the control system at each time step. The reference angular position and velocity of each joint are defined as oscillators with fixed amplitude, natural frequency and phase,

$$\mathbf{Q}_{N \times 1}^r(t) = \begin{bmatrix} \vartheta_{r,0}(t) \\ \dots \\ \vartheta_{r,N}(t) \end{bmatrix} = \begin{bmatrix} A_0 \cdot \sin(2\pi ft + \varphi_0) \\ \dots \\ A_N \cdot \sin(2\pi ft + \varphi_N) \end{bmatrix}, \quad (3)$$

$$\dot{\mathbf{Q}}_{N \times 1}^r(t) = \begin{bmatrix} \dot{\vartheta}_{r,0}(t) \\ \dots \\ \dot{\vartheta}_{r,N}(t) \end{bmatrix} = \begin{bmatrix} 2\pi f A_0 \cdot \cos(2\pi ft + \varphi_0) \\ \dots \\ 2\pi f A_N \cdot \cos(2\pi ft + \varphi_N) \end{bmatrix}, \quad (4)$$

where $N = 2$. The temporal frequency is $f = 0.25\text{Hz}$, while the oscillations A amplitude and φ phase of each joint are set to:

$$\mathbf{A}_{1 \times N} = [A_0, A_1, A_2] = [0.1727, 0.1363, 0.0345] \text{ rad}$$

$$\varphi_{1 \times N} = [\varphi_0, \varphi_1, \varphi_2] = [0.5\pi, 0.5\pi, 0.0] \text{ rad.}$$

2.3. Controller

The *controller* block (**Figure 1A**) is composed of a static component based on classic control methods (section 2.3.1), and of an adaptive decentralized block representing the bio-inspired regulator, i.e., the cerebellar-like circuit (section 2.3.2). Both sub-blocks receive information about the \mathbf{Q}^c , $\dot{\mathbf{Q}}^c$ process variables measured from the encoders located in the *robotic plant* (Equations 1, 2), and the \mathbf{Q}^r , $\dot{\mathbf{Q}}^r$ reference trajectory signals from the *motor primitive generator* (Equations 3, 4).

The controller directly sends the τ^{tot} total control input to the robot servo controller which actuates the joints for $\delta t = 0.5\text{s}$. The τ^{tot} total control input is expressed as the result of a feed-forward compensation (as the AFEL architecture proposed by Tolu et al., 2012),

$$\tau_{N \times 1}^{tot} = \begin{bmatrix} \tau_0^{tot} \\ \dots \\ \tau_N^{tot} \end{bmatrix} = \begin{bmatrix} \tau_0^{PID} + \Delta\tau_0^{DCN} \\ \dots \\ \tau_N^{PID} + \Delta\tau_N^{DCN} \end{bmatrix}, \quad (5)$$

where τ_n^{PID} and $\Delta\tau_n^{DCN}$ (where $n = 0, \dots, N$) are the contributions from the static and the adaptive bio-inspired controller respectively.

2.3.1. Feedback Controller

The static control system refers to the classic feedback control scheme with PID regulator. It is defined static due to its time-constant control terms. The closed-loop system continuously computes the $\mathbf{e}_{\dot{\vartheta}_n}$ angular velocity error of each joint as the difference between the $\dot{\vartheta}_{r,n}$ reference (Equation 4) and the $\dot{\vartheta}_{c,n}$ process variable (Equation 2),

$$\mathbf{e}_{N \times 1}^{vel} = \begin{bmatrix} e_{\dot{\vartheta}_0} \\ \dots \\ e_{\dot{\vartheta}_N} \end{bmatrix} = \begin{bmatrix} \dot{\vartheta}_{r,0} - \dot{\vartheta}_{c,0} \\ \dots \\ \dot{\vartheta}_{r,N} - \dot{\vartheta}_{c,N} \end{bmatrix}. \quad (6)$$

The $\mathbf{e}_{\dot{\vartheta}_n}$ error (where $n = 0, \dots, N$) is used to apply correction to each controlled joint in terms of effort,

$$\tau_{N \times 1}^{PID} = [\tau_0^{PID}, \dots, \tau_N^{PID}]^T, \quad (7)$$

according to the independent joint control law expressed as:

$$\tau_n^{PID}(t) = K_{P,n} \cdot e_{\dot{\vartheta}_n} + K_{I,n} \cdot \int_{t-\Delta t}^t e_{\dot{\vartheta}_n}(t') dt' + K_{D,n} \cdot \frac{d\dot{\vartheta}_n(t)}{dt}$$

for $n = 0, \dots, N$, (8)

where the integration time window is $\Delta t = 10$ samples. The regulator is tuned to weakly operate in a linearized condition which excludes the presence and disturbance of the ball, hence the proportional, integrative and derivative terms are static and set respectively to,

$$\mathbf{K}_P = [K_{P,0}, K_{P,1}, K_{P,2}] = [2.9000, 2.3000, 2.3500]$$

$$\mathbf{K}_I = [K_{I,0}, K_{I,1}, K_{I,2}] = [1.9400, 1.9000, 1.9000]$$

$$\mathbf{K}_D = [K_{D,0}, K_{D,1}, K_{D,2}] = [0.0050, 0.0001, 0.0004].$$

2.3.2. Cerebellar-Like Model

The proposed cerebellar-like network has been designed to solve robotic problems (**Figure 2**). In particular, the sensory input and the corrective action in output refer to entities regarding the actuated motors, such as motor angular position, velocity or effort. Electrophysiological evidence about the encoding of movement kinematics has been found at all levels of the cerebellum; for example, in this review (Ebner et al., 2011), reported that the mossy fibers (MF) inputs encode the position,

TABLE 2 | Actuated joints information: the wrist actuators (highlighted in yellow) are controlled in effort while the elbow and shoulder motors are kept to a constant angular position.

	Min $\dot{\theta}$ [rad]	Max $\dot{\theta}$ [rad]	Max $\dot{\theta}$ [rad · sec ⁻¹]	Control	Value
Wrist prosup $n = 0$	-0.8726	0.8726	100	Effort	Controlled variable
Wrist yaw $n = 1$	-0.4363	0.4363	100	Effort	Controlled variable
Wrist pitch $n = 2$	-1.1344	0.1745	100	Effort	Controlled variable
Elbow	0.0959	1.8500	100	Position	Constant = 1.14 [rad]
Shoulder roll	0.0000	2.80649	100	Position	Constant = 0.1 [rad]
Shoulder yaw	-0.645772	1.74533	100	Position	Constant = -0.1 [rad]
Shoulder pitch	-1.65806	0.0872665	100	Position	Constant = -0.9 [rad]

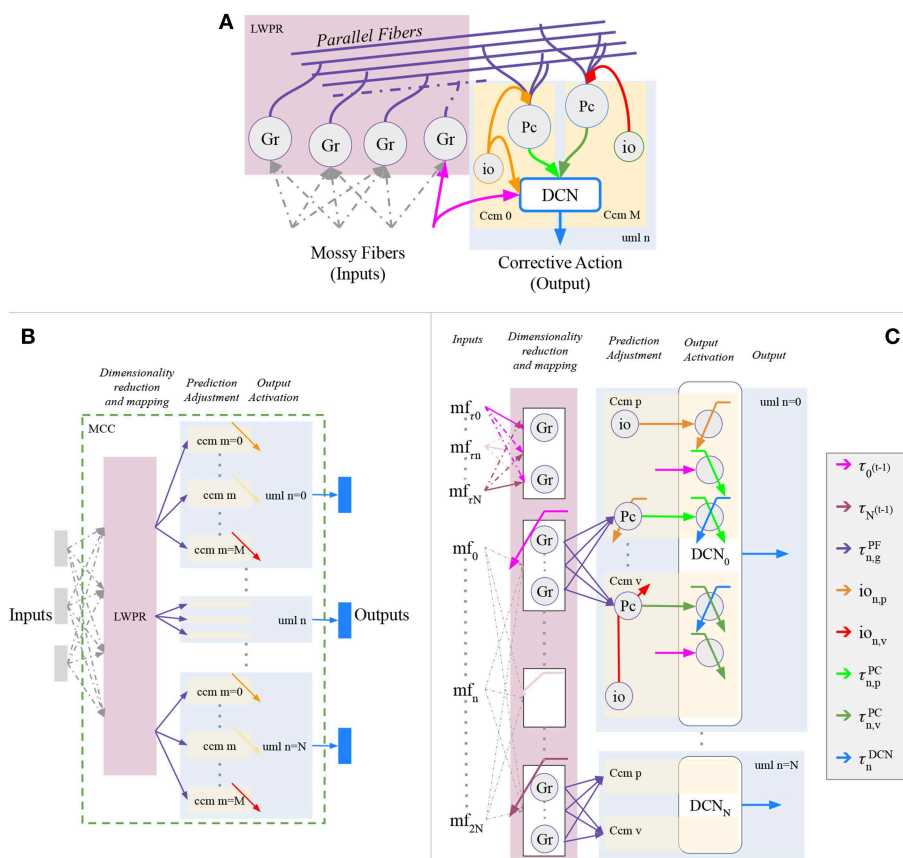


FIGURE 2 | Proposed cerebellar-like circuit in analogy with D'Angelo et al. (2016). (A) canonical micro-circuit. Proposed cerebellar-like neural network (B) structural partition and (C) details.

direction, and velocity of limb movements. Moreover, many hypotheses suggest that the cerebellum directly contributes to the motor command required to produce a movement. In our model, the input-output relationship is based on the previous suggestions and the signal propagation throughout the cerebellar network layers is in accordance with the robotic control application. The main design concept is that the signal propagating inside the circuit have the same dimension of the $\Delta\tau^{DCN}$ output signal from the Deep Cerebellar Nuclei (DCN). The propagated signal is modulated inside the network by other

signals that are correlated with the intrinsic features of the controlled plant, such as position and velocity terms, in order to have a complete description of the state.

The neural network structure is divided into separated modules (Figure 2B), or namely *Unit Learning Machine* (uml) (Tolu et al., 2012, 2013). Assuming that the robot plant is composed by N controllable object, then each uml is specialized on the n -th controlled object (where $n = 0, \dots, N$), or rather the DCN output of the uml will be the cerebellar contribution for the specific object. The uml itself is

separated into M sub-modules which represent the *canonical cerebellar microcircuit* (ccm). Each ccm is specialized with respect to a specific feature describing the behavior of the n -th controlled object. The overall umls and other structures, that are dedicated to the dimensionality reduction and mapping of the sensory information, compose together the *Modular Cerebellar Circuit* (MCC).

In the proposed experiment, the canonical cerebellar microcircuits (ccm) of each controlled object are specialized in p position and in v velocity. In details, the Purkinje layer of each n -th uml presents a pair of Purkinje cells (PC) (**Figure 2C**), specialized in position $Pc_{n,p}$ and velocity $Pc_{n,v}$ respectively through different climbing fibers ($io_{n,p}$ and $io_{n,v}$). Moreover, the bio-inspired controller receives the same sensory information

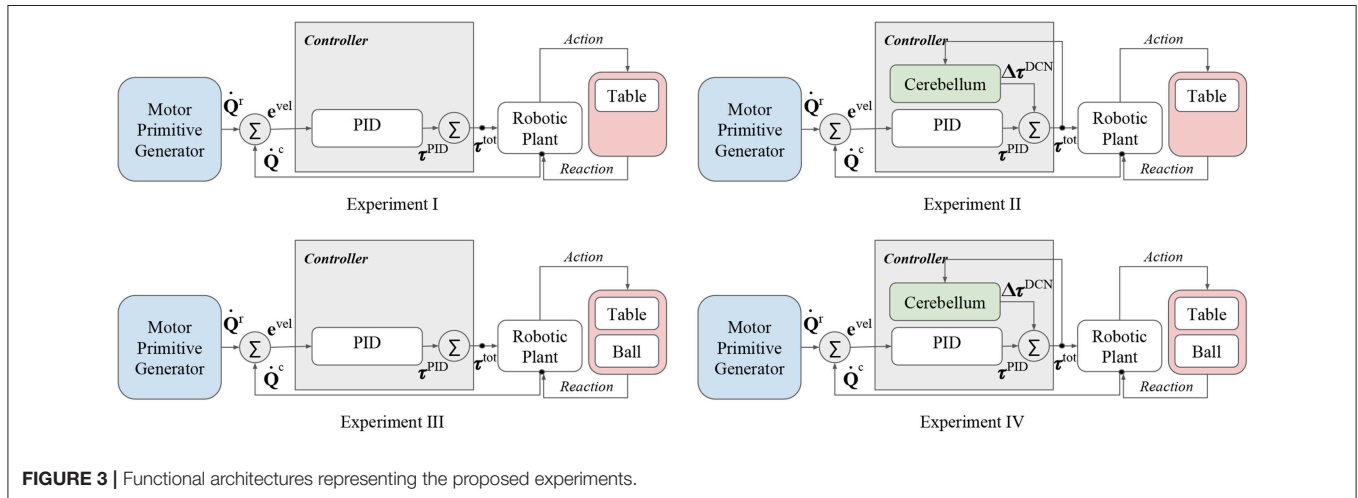


FIGURE 3 | Functional architectures representing the proposed experiments.

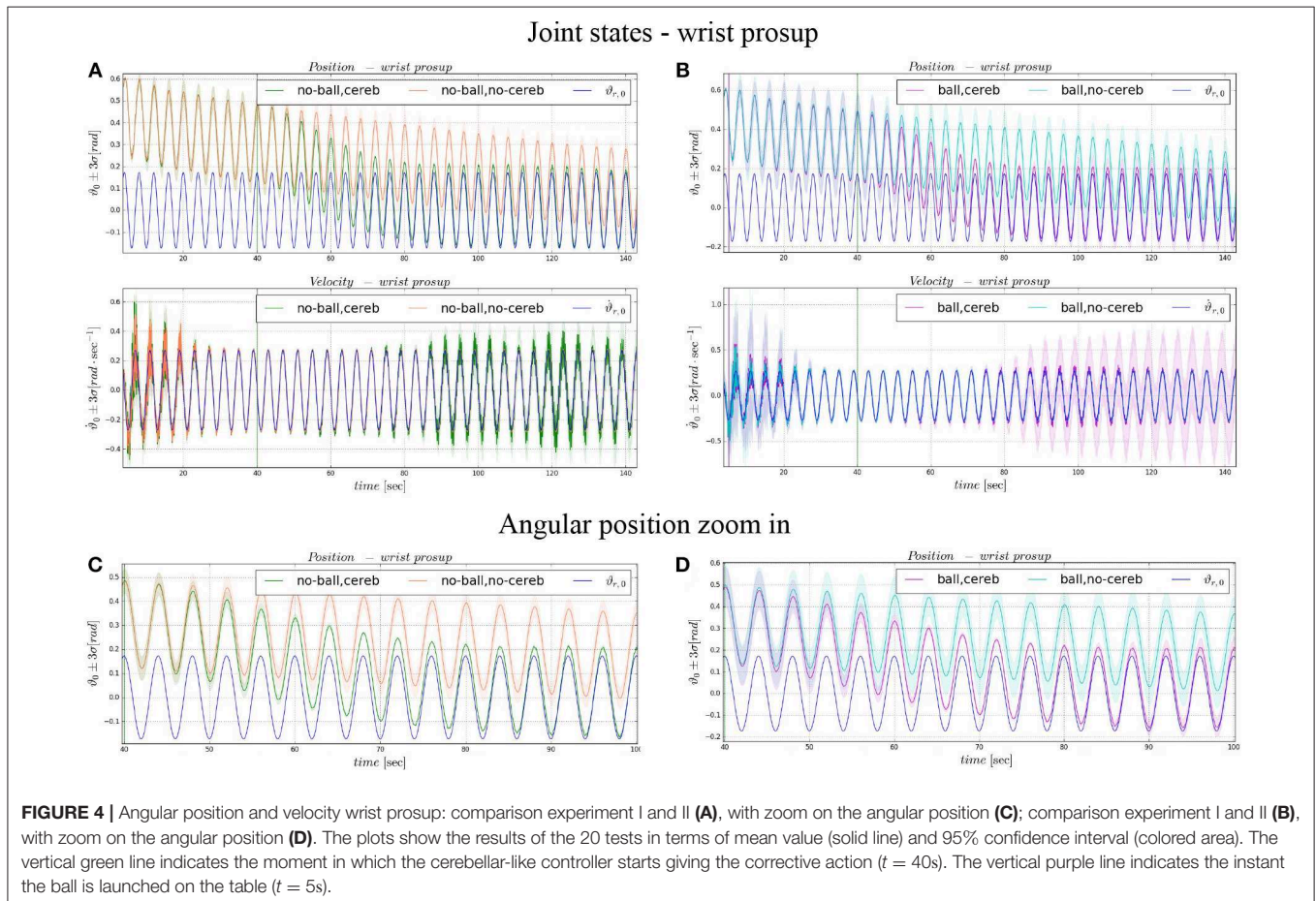


FIGURE 4 | Angular position and velocity wrist prosup: comparison experiment I and II (A), with zoom on the angular position (C); comparison experiment I and II (B), with zoom on the angular position (D). The plots show the results of the 20 tests in terms of mean value (solid line) and 95% confidence interval (colored area). The vertical green line indicates the moment in which the cerebellar-like controller starts giving the corrective action ($t = 40$ s). The vertical purple line indicates the instant the ball is launched on the table ($t = 5$ s).

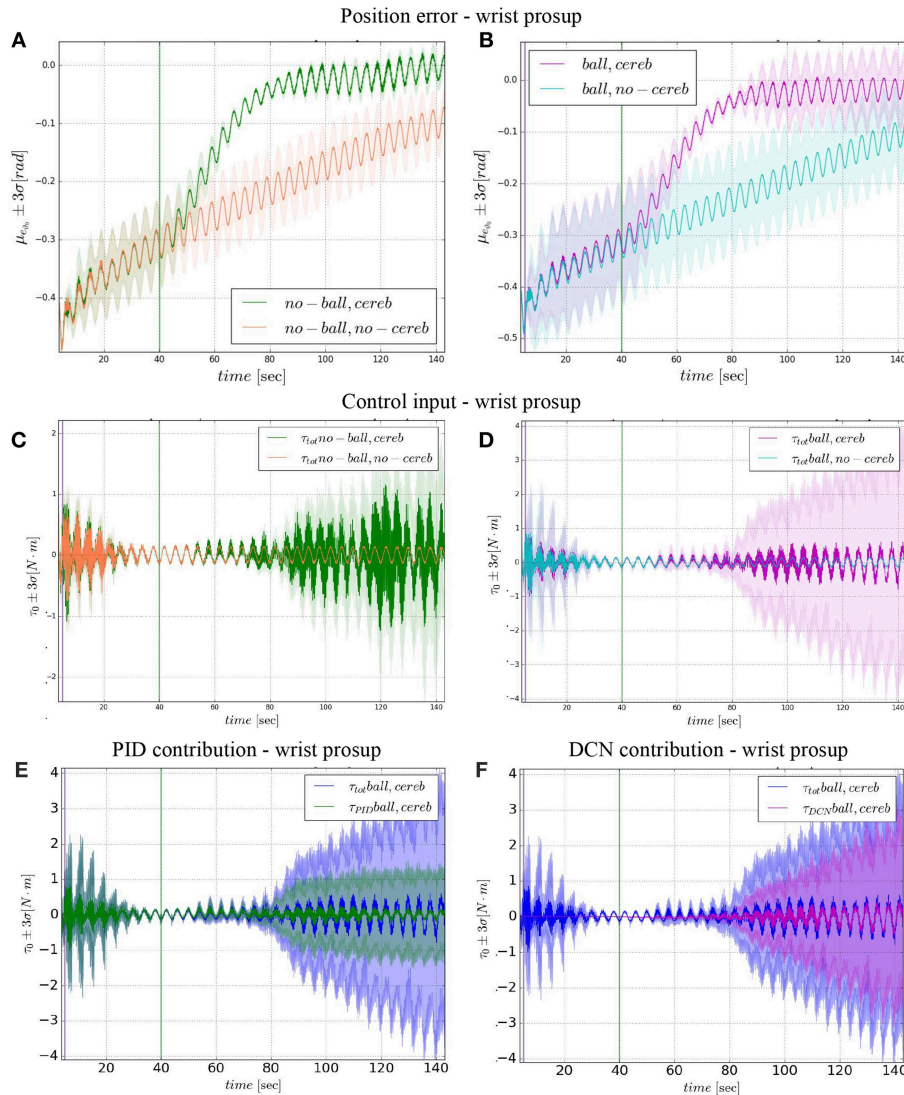


FIGURE 5 | Wrist prosup experimental results. Resulting angular position error e_{ϑ_0} , comparison experiments I and II **(A)**, comparison experiments III IV **(B)**. Control input τ_0^{tot} evolution, comparison experiments I and II **(C)**, comparison experiments III IV **(D)**. Control input contributions in experiment IV comparisons between: τ_0^{tot} and τ_0^{PID} **(E)**; τ_0^{tot} and τ_0^{DCN} **(F)**. The plots show the results of the 20 tests in terms of mean value (solid line) and 95% confidence interval (colored area). The vertical green line indicates the moment the cerebellar-like controller starts giving the corrective action ($t = 40$ s). The vertical purple line indicates the instant the ball is launched on the table ($t = 5$ s).

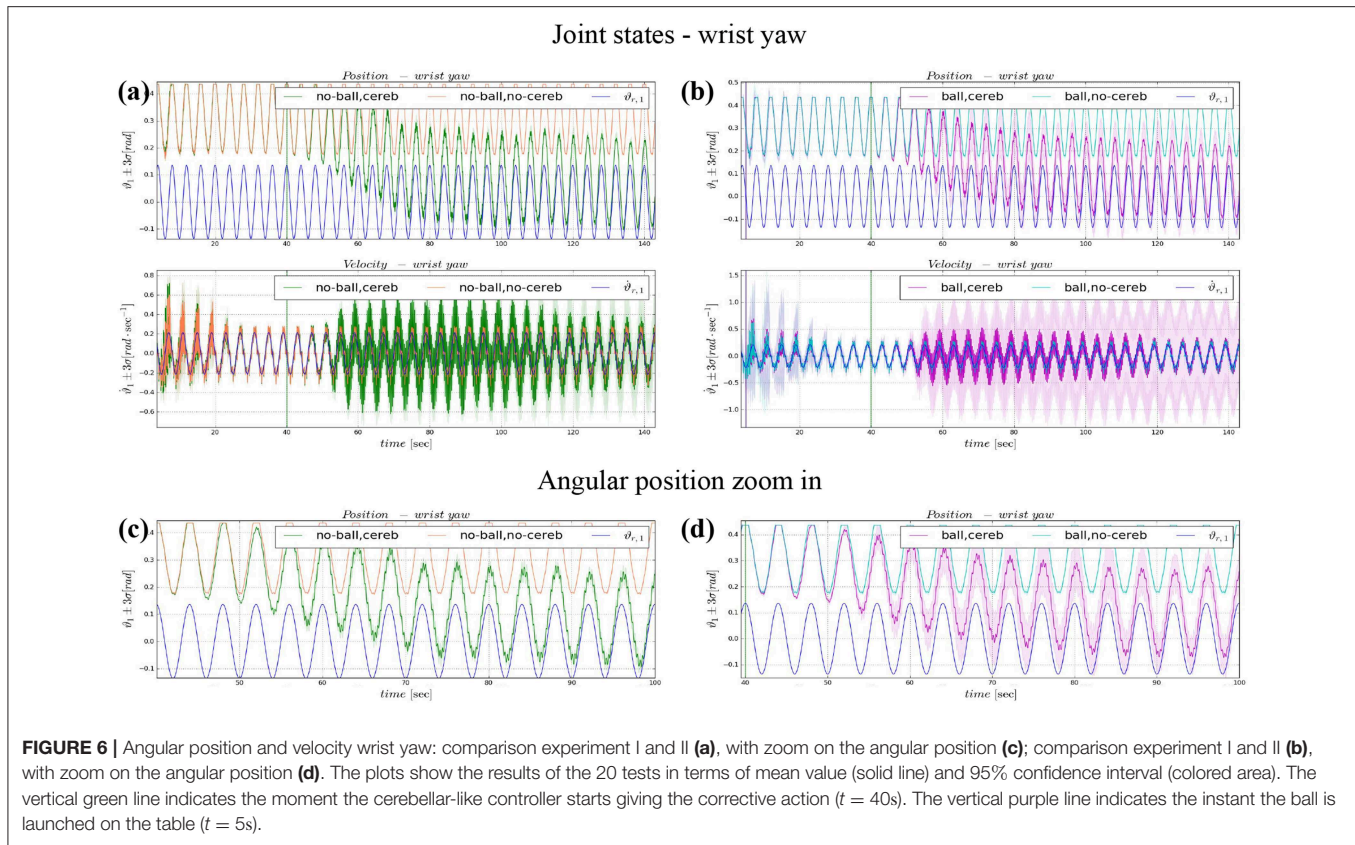
of the feedback controller (section 2.3.1), but it is intended to correct the e_{ϑ_n} angular position error, whereas the PID corrects the $e_{\dot{\vartheta}_n}$ angular velocity error. This is solved through the connection inferior olive-deep cerebellar nuclei (IO-DCN), which conveys information about the angular position error. An additional aspect, the inferior olive signals differs from Kawato's feedback error learning theory (Kawato, 1990) and our previous experiments (Tolu et al., 2012, 2013), because the Jacobian does not correctly approximate the system, therefore the required conditions are not satisfied and it is not efficient to compare the motor signals.

The mossy fibers transmit the information about the current and reference state of the controlled

joints in terms of angular velocity to the granular cells (Gr),

$$\mathbf{MF}_{2N \times 1}(t) = \begin{bmatrix} mf_0(t) \\ \vdots \\ mf_{2N}(t) \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{Q}}_{N \times 1}^r(t) \\ \dot{\mathbf{Q}}_{N \times 1}^c(t) \end{bmatrix} = \begin{bmatrix} \dot{\vartheta}_{r,0}(t) \\ \vdots \\ \dot{\vartheta}_{r,N}(t) \\ \dot{\vartheta}_{c,0}(t) \\ \vdots \\ \dot{\vartheta}_{c,N}(t) \end{bmatrix}. \quad (9)$$

The granular layer-parallel fibers network is the circuit area committed to the mapping of the mossy fibers signals and to the prediction of the next output given the current



sensory input (Marr, 1969; Albus, 1971). As in our previous works (Tolu et al., 2012, 2013), we artificially represented this network with the Locally Weighted Projection Regression algorithm (LWPR) (Vijayakumar and Schaal, 2000). The LWPR resulted an efficient method for the fast on-line approximation of non-linear functions in high dimensional spaces. Given the $\mathbf{MF}(t)$ mossy fibers input vector (Equation 9), the LWPR creates G local linear models that in our scheme represent the Gr_g granular cells (for $g = 0, \dots, G$). Each linear model employs the $\mathbf{MF}(t)$ to make a $\hat{\tau}_{n,g}^{gr}(t)$ prediction of the control input $\tau_n^{tot}(t-1)$ (where $n=1, \dots, N$). The total output of the granular-parallel fibers network is the weighted mean of all the linear models specialized in velocity,

$$\hat{\tau}_n^{PF}(t) = \frac{\sum_{g=1}^{G} w_{n,g}^{gr}(t) \cdot \hat{\tau}_{n,g}^{gr}(t)}{\sum_{g=1}^{G} w_{n,g}^{gr}(t)} \text{ for } n = 1, \dots, N, \quad (10)$$

where $w_{n,g}^{gr}$ and $\hat{\tau}_{n,g}^{gr}$ are defined in Vijayakumar and Schaal (2000).

In our scheme, there are two Purkinje cells per controlled joint $Pc_{n,p}$ and $Pc_{n,v}$ (where $n = 0, \dots, N$). The $w_{n,p}^{pf-pc1}$ synapses connecting the parallel fibers and the $Pc_{n,p}$ (PF-PC connection) (Garrido Alcazar et al., 2013), are modulated by the

$io_{n,p}$ inferior olive (IO) signal,

$$io_{n,p}(t) = \tilde{e}_{\vartheta_n}(t), \quad (11)$$

that transmits the information about the \tilde{e}_{ϑ_n} normalized angular position error of the n -th joint,

$$e_{\vartheta_n}(t) = \vartheta_{r,n}(t) - \vartheta_{c,n}(t), \quad (12)$$

while the $w_{n,v}^{pf-pc1}$ synaptic strengths between the parallel fibers and the $Pc_{n,v}$, are modulated by the $io_{n,v}$ inferior olive signal,

$$io_{n,v}(t) = \tilde{e}_{\dot{\vartheta}_n}(t), \quad (13)$$

that transmits the information about the $\tilde{e}_{\dot{\vartheta}_n}$ normalized angular velocity error of the n -th joint (Equation 6). The $w_{n,p}^{pf-pc}$ weighting kernel tends to support the control actions that lead to an error lower than a specific threshold e^{thresh} ,

$$\begin{aligned} \mathbf{e}_{\vartheta}^{thresh,pc} &= \begin{bmatrix} e_{\vartheta_0}^{thresh,pc} \\ \dots \\ e_{\vartheta_N}^{thresh,pc} \end{bmatrix} = \begin{bmatrix} w_{0,p}^{pf-pc}(t, io_0^p(t) = 0) \cdot \max(e_{\vartheta_0}) \\ \dots \\ w_{N,p}^{pf-pc}(t, io_{N,p}(t) = 0) \cdot \max(e_{\vartheta_N}) \end{bmatrix} \\ &= \begin{bmatrix} 0.012 \\ 0.008 \\ 0.002 \end{bmatrix} [\text{rad}], \end{aligned} \quad (14)$$

¹ w^{pf-pc} weighting kernel parameters: $LTD_{max} = 10^{-3}$, $LTP_{max} = 10^{-3}$, $\alpha = 170$.

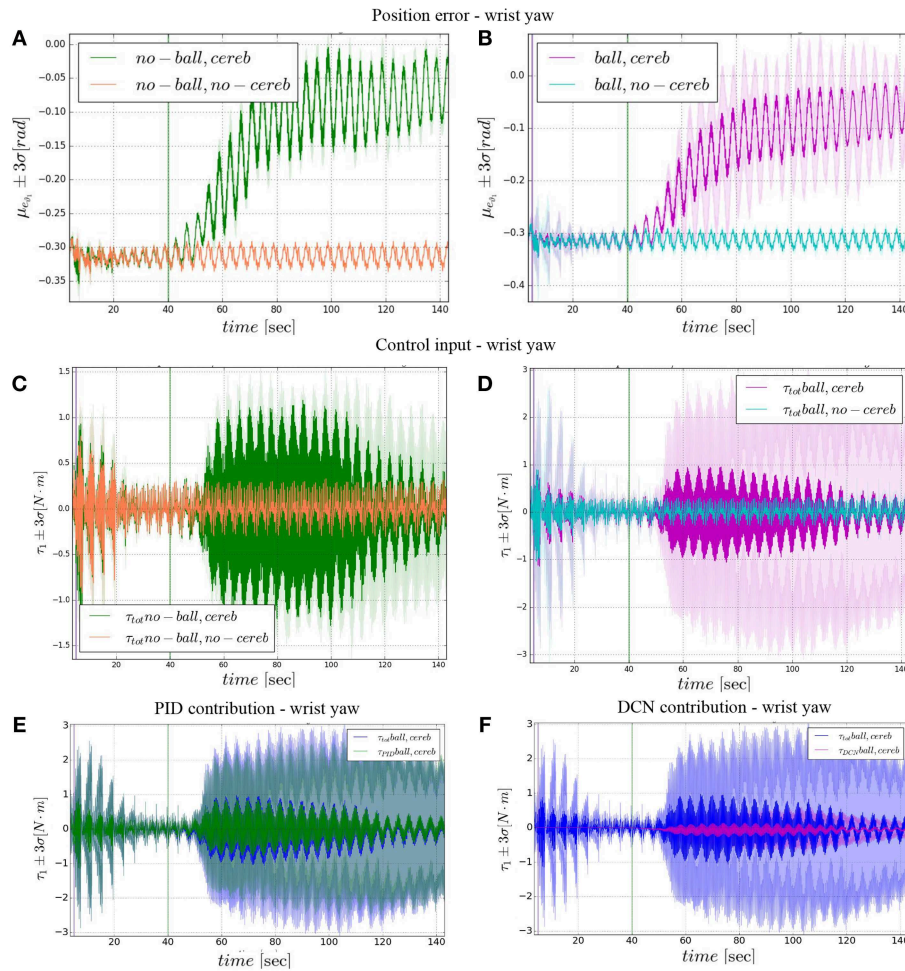


FIGURE 7 | Wrist yaw experimental results. Resulting angular position error e_{ϑ_1} , comparison experiments I and II (A), comparison experiments III IV (B). The τ_1^{tot} control input evolution, comparison experiments I and II (C), comparison experiments III IV (D). Control input contributions in experiment IV comparisons between: τ_1^{tot} and τ_1^{PID} (E); τ_1^{tot} and τ_1^{DCN} (F). The plots show the results of the 20 tests in terms of mean value (solid line) and 95% confidence interval (colored area). The vertical green line indicates the moment the cerebellar-like controller starts providing the corrective action ($t = 40$ s). The vertical purple line indicates the instant the ball is launched on the table ($t = 5$ s).

$$\begin{aligned}
 \mathbf{e}_{\dot{\vartheta}}^{thresh,pc} &= \begin{bmatrix} e_{\dot{\vartheta}_0}^{thresh,pc} \\ \dots \\ e_{\dot{\vartheta}_N}^{thresh,pc} \end{bmatrix} = \begin{bmatrix} w_{0,v}^{pf-pc}(t, i_{\dot{\vartheta}_0}(t) = 0) \cdot \max(e_{\dot{\vartheta}_0}) \\ \dots \\ w_{N,v}^{pf-pc}(t, i_{\dot{\vartheta}_N}(t) = 0) \cdot \max(e_{\dot{\vartheta}_N}) \end{bmatrix} \\
 &= \begin{bmatrix} 0.012 \\ 0.008 \\ 0.002 \end{bmatrix} [\text{rad} \cdot \text{sec}^{-1}]. \quad (15)
 \end{aligned}$$

Respect to our previous work (Tolu et al., 2012, 2013) the output signals of the Purkinje cells are directly function of the $\hat{\tau}_n^{PF}(t)$ prediction instead of the $w_{n,g}^{gr}$ weights,

$$\tau_{n,p}^{PC}(t) = w_{n,p}^{pf-pc}(t, i_{\dot{\vartheta}_p}(t)) \cdot \hat{\tau}_n^{PF}(t) \quad (16)$$

$$\tau_{n,v}^{PC}(t) = w_{n,v}^{pf-pc}(t, i_{\dot{\vartheta}_v}(t)) \cdot \hat{\tau}_n^{PF}(t). \quad (17)$$

Afterwards, the $\tau_{n,p}^{PC}(t)$ $\tau_{n,v}^{PC}(t)$ Purkinje cells signals are scaled by the synaptic weights $w_{n,p}^{pc-dcn}$ and $w_{n,v}^{pc-dcn}$ (Garrido Alcazar et al., 2013), that are modulated by the Purkinje cells and the deep cerebellar nuclei activities (PC-DCN),

$$w_{n,p}^{pc-dcn} = f(t, \tau_{n,p}^{PC}(t), \Delta \tau_n^{DCN}(t-1)), \quad (18)$$

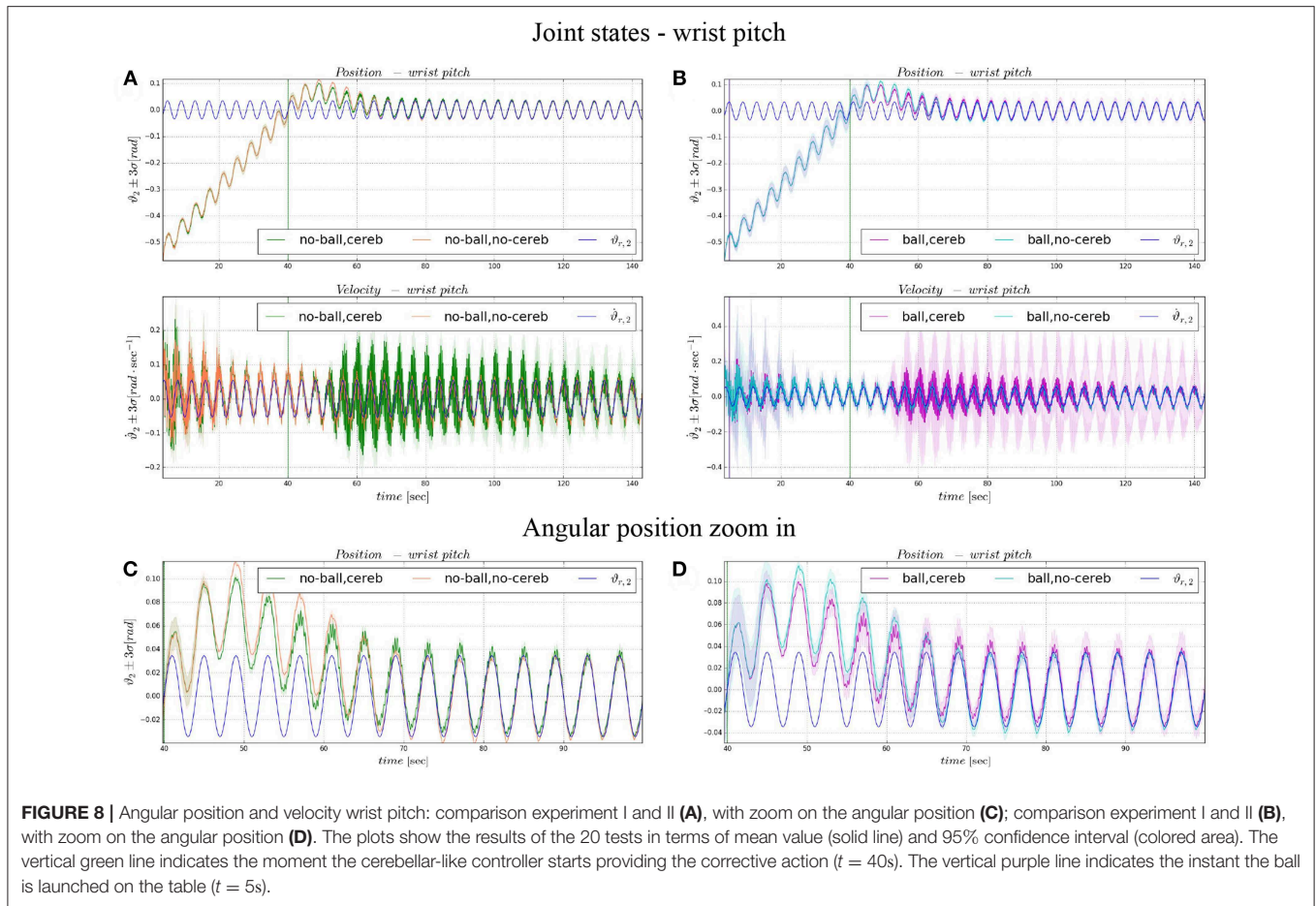
$$w_{n,v}^{pc-dcn} = f(t, \tau_{n,v}^{PC}(t), \Delta \tau_n^{DCN}(t-1)). \quad (19)$$

resulting in the input signals,

$$\tau_{n,p}^{PC-DCN}(t) = w_{n,p}^{pc-dcn} \cdot \tau_{n,p}^{PC}(t) \quad (20)$$

$$\tau_{n,v}^{PC-DCN}(t) = w_{n,v}^{pc-dcn} \cdot \tau_{n,v}^{PC}(t). \quad (21)$$

² $w_{n,p}^{pc-dcn}$ weighting kernel parameters: $LTD_{max} = 10^{-4}$, $LTP_{max} = 10^{-4}$, $\alpha = 2$.



In addition, the deep cerebellar nuclei receives the input signals $\tau_{n,p}^{MF-DCN}$, $\tau_{n,v}^{MF-DCN}$ from the mossy fibers and $\tau_{n,p}^{IO-DCN}$ from the inferior olive. In our proposed circuit, the mossy fibers connected to the deep cerebellar nuclei (MF-DCN) conveys the information about the $\tau_n^{tot}(t-1)$ last control input sent to each controlled joint (Equation 5). This input is scaled by the synaptic weights $w_{n,p}^{mf-dcn}$ and $w_{n,v}^{mf-dcn}$ (Garrido Alcazar et al., 2013), modulated by the respective n -th Purkinje cells activities,

$$\tau_{n,p}^{MF-DCN}(t) = w_{n,p}^{mf-dcn}(t, \tau_{n,p}^{PC}(t)) \cdot \tau_n^{tot}(t-1), \quad (22)$$

$$\tau_{n,v}^{MF-DCN}(t) = w_{n,v}^{mf-dcn}(t, \tau_{n,v}^{PC}(t)) \cdot \tau_n^{tot}(t-1). \quad (23)$$

The $\tau_{n,p}^{IO-DCN}$ inferior olive contribution in the deep cerebellar nuclei (IO-DCN) is given by the $io_{n,p}$ (Equation 11), which is modulated by the $w_{n,p}^{io-dcn}$ synaptic weight (Luque et al., 2014),

$$\tau_{n,p}^{IO-DCN} = w_{n,p}^{io-dcn}(t, io_{n,p}(t)) \cdot io_{n,p}(t). \quad (24)$$

³ $w_{n,p}^{mf-dcn}$ weighting kernel parameters: $LTD_{max} = 10^{-4}$, $LTP_{max} = 10^{-4}$, $\alpha = 2$.

⁴ $w_{n,p}^{io-dcn}$ weighting kernel parameters: $MTD_{max} = -10^{-4}$, $MTP_{max} = -10^{-5}$, $\alpha = 100$.

The final $\Delta\tau_n^{DCN}$ cerebellar corrective term is the result of the $\tau_{n,p}^{MF-DCN}$ modulated control input subtracted by the $\tau_{n,p}^{PC-DCN}$ prediction modulated by the current error together with the $\tau_{n,p}^{IO-DCN}$ modulated contribution of the error itself,

$$\Delta\tau_n^{DCN} = (\tau_{n,p}^{MF-DCN} + \tau_{n,v}^{MF-DCN}) - (\tau_{n,p}^{PC-DCN} + \tau_{n,v}^{PC-DCN}) + \tau_{n,p}^{IO-DCN}, \quad (25)$$

or rather,

$$\Delta\tau_n^{DCN} = (\tau_n(\vartheta_n, \tau^{tot}) + \tau_n(\dot{\vartheta}_n, \tau^{tot})) - (\hat{\tau}_n^{tot}(e_{\vartheta_n}) + \hat{\tau}_n^{tot}(e_{\dot{\vartheta}})) + \tau_n(e_{\vartheta_n}).$$

2.4. Proposed Experiments and Performance Measures

The proposed control scheme has been applied in four different experiments with the aim at analyzing the advantages of the bio-inspired controller in presence of dynamical disturbances. In details, the four experiments differ from the presence of the ball and the cerebellar-like controller contribution (Figure 3):

- **Experiment I:** control input without both cerebellum contribution and ball disturbance,

$$\tau^{tot} = \tau^{PID} \text{ (no ball)}; \quad (26)$$

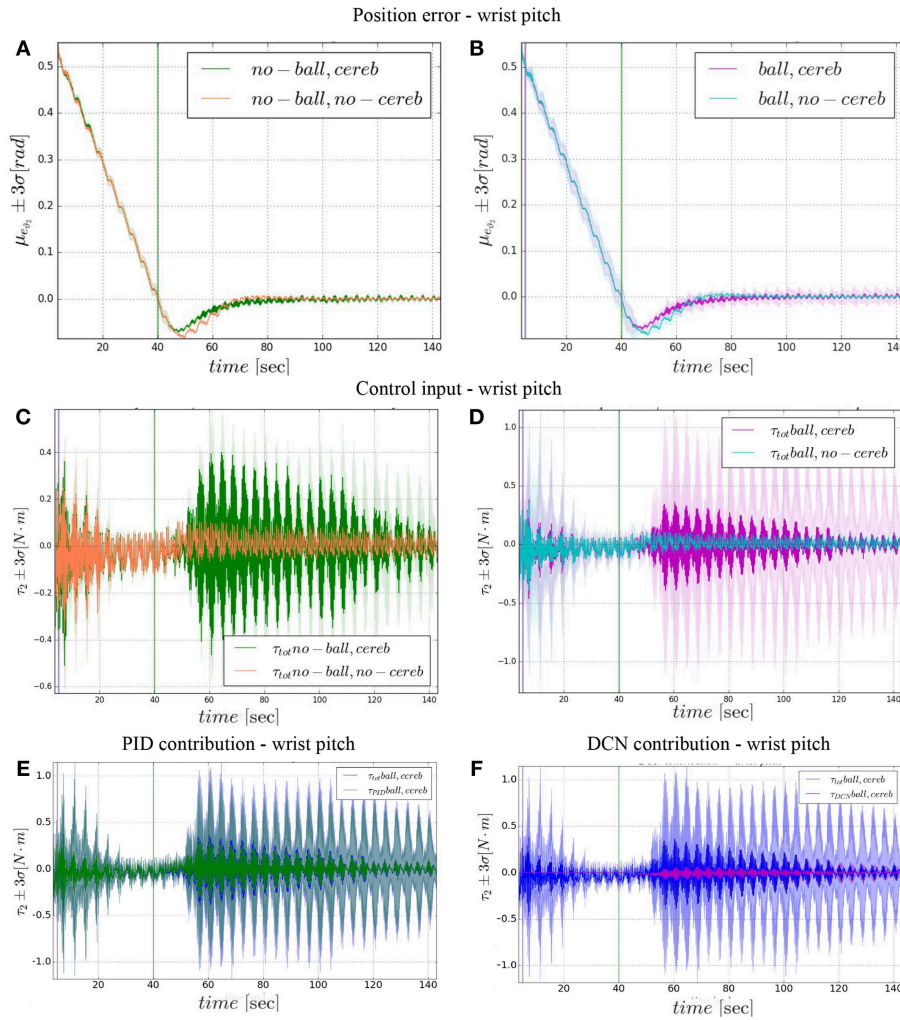


FIGURE 9 | Wrist pitch experimental results. Resulting angular position error e_{ϑ_2} , comparison experiments I and II (A), comparison experiments III IV (B). The τ_2^{tot} control input evolution, comparison experiments I and II (C), comparison experiments III IV (D). Control input contributions in experiment IV comparisons between: τ_2^{tot} and τ_2^{PID} (E); τ_2^{tot} and τ_2^{DCN} (F). The plots show the results of the 20 tests in terms of mean value (solid line) and 95% confidence interval (colored area). The vertical green line indicates the moment the cerebellar-like controller starts providing the corrective action ($t = 40$ s). The vertical purple line indicates the instant the ball is launched on the table ($t = 5$ s).

- **Experiment II:** control input with cerebellum contribution, without ball disturbance,

$$\tau^{tot} = \tau^{PID} + \Delta\tau^{DCN} \text{ (no ball);} \quad (27)$$

- **Experiment III:** control input without cerebellum contribution, with ball disturbance,

$$\tau^{tot} = \tau^{PID} \text{ (ball);} \quad (28)$$

- **Experiment IV:** control input with both cerebellum contribution and ball disturbance,

$$\tau^{tot} = \tau^{PID} + \Delta\tau^{DCN} \text{ (ball).} \quad (29)$$

The performance of each experiment will be measured by analysis of the *mean absolute error* (MAE) evolution computed for the angular position error of each controlled joint (Equation 12),

$$mae_{\vartheta_n}(k) = \frac{\sum_{i=t}^{t+T} |e_{\vartheta_n}(i)|}{T} \text{ for } n = 0, \dots, N. \quad (30)$$

The MAE is computed for every trajectory period $T = 8$ s (Equation 3).

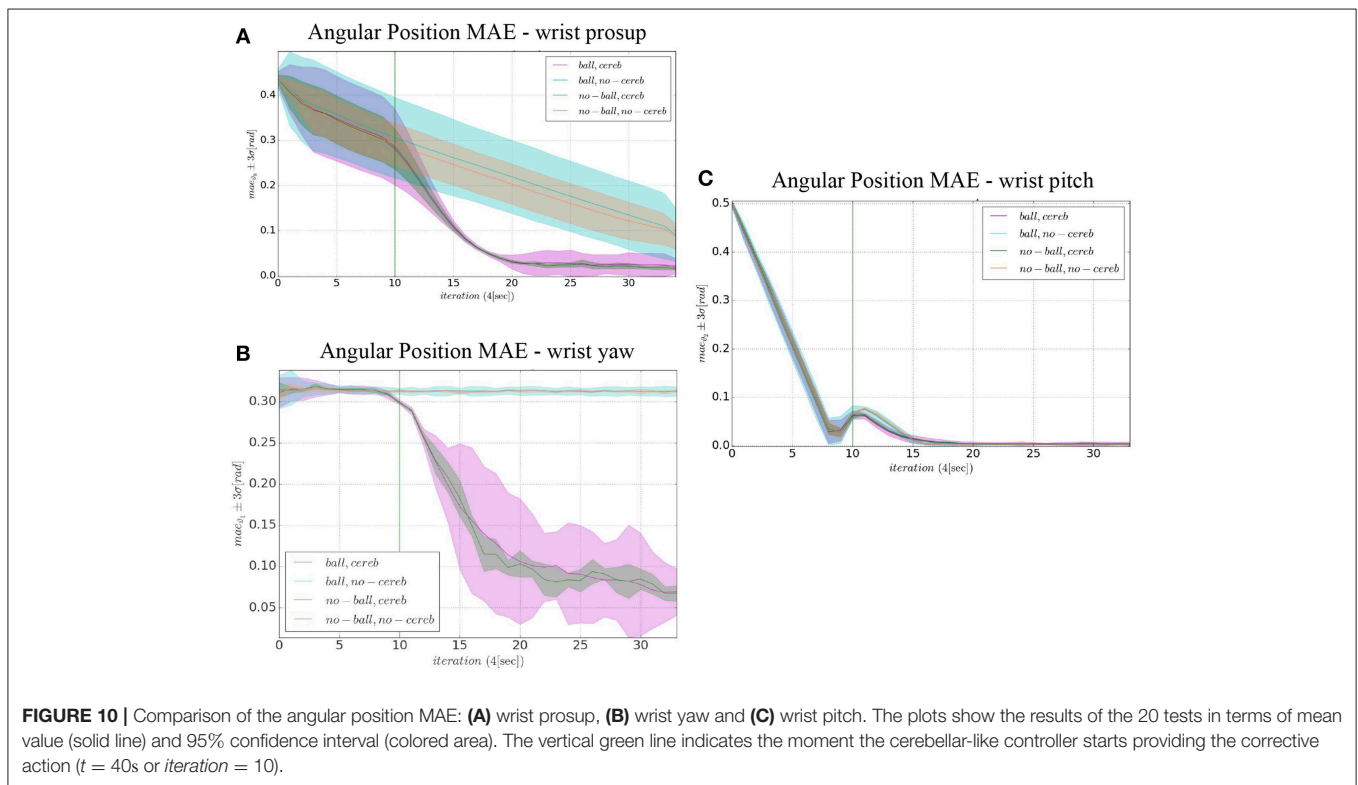
3. RESULTS

The software describing the system is based on the ROS (Quigley et al., 2009) messaging architecture and is integrated in the Neurobotics Platform (NRP) (Falotico et al., 2017).

TABLE 3 | The mean absolute error (MAE) of the initial and final period ($T = 4$ s).

		Ball				No ball			
		Cerebellum		No cerebellum		Cerebellum		No cerebellum	
		Initial	Final	Initial	Final	Initial	Final	Initial	Final
mae_{ϑ_0} [rad]	μ	0.4136	0.0241	0.4148	0.1104	0.4068	0.0188	0.4136	0.1037
	σ	0.0181	0.0085	0.0273	0.0242	0.0120	0.0017	0.0319	0.0075
mae_{ϑ_1} [rad]	μ	0.3148	0.0689	0.3168	0.3123	0.3139	0.0671	0.3172	0.3130
	σ	0.0048	0.0093	0.0070	0.0018	0.0012	0.0031	0.0016	0.0005
mae_{ϑ_2} [rad]	μ	0.4380	0.0042	0.4395	0.0019	0.4437	0.0037	0.4401	0.0020
	σ	0.0045	0.0012	0.0063	0.0001	0.0026	0.0003	0.0013	2.9177e-05

The results express the mean value μ and standard deviation σ of the 20 tests run for the four experiments.



The NRP is a simulation environment based on ROS and Gazebo (Koenig and Howard, 2004) which includes a variety of robots, environments and a detailed physics simulator. The three wrist motors are controlled in effort through the Gazebo service *ApplyJointEffort*, while the elbow and the three shoulder motors are controlled in position through their specific ROS topic. The sensory information from the encoders are received with a sampling frequency of $f_{\text{sample}} = 50$ Hz. The computer used for the test has the Ubuntu 16.04 Operating system (OS type 64-bit), the Intel Core™ i7-7700HQ CPU@2.80GHz \times 8 processor, and the GeForce GTX 1050/PCIe/SSE2 graphics card.

Each experiment was performed 20 times with a total duration of about 3 min. The recorded data was saved in.csv files and processed for the analysis. The results are expressed as mean value of the 20 tests, and σ standard deviation or 95% confidence interval. In each experiment, the cerebellar-like circuit is

activated after $t = 40$ s (or 10th iteration), which is the moment all the actuated joints reach a stable configuration (included the shoulder joints and the elbow). In experiments II and IV, the ball is launched on the table after $t = 5$ s (purple vertical line in the figures).

The comparison of the 4 experiments for each controlled joint are presented separately in 3 parts. In each part, we analyze the joint states, i.e., $\vartheta_{c,n}(t)$ angular position and $\dot{\vartheta}_{c,n}(t)$ velocity (Figures 4, 6, 8), respect to the control action (Figures 5, 7, 9). Moreover, we compared the mean absolute error MAE to measure the performance of the different cases (as reported in Table 3 and illustrated in Figure 10).

3.1. Wrist Prosup

In the details of Figures 4A,B, the corrective action of the cerebellar-like circuit (Experiments II, IV) leads $\vartheta_{c,0}$ faster to

the desired trajectory $\vartheta_{r,0}$ with respect to the case without corrections (Experiments I, III). $\vartheta_{c,0}(t)$ starts getting closer to the desired position in about one period $T = 4$ s after the activation of the cerebellum (Figures 4C,D). In Figures 5A,B it is evident how the angular position error e_{ϑ_0} drops when the cerebellum action grows (Figures 5C,D). In particular, the mean absolute error drastically decreased by the 95 and 94% in experiment II and IV respectively, while it only decreased by the 74 and 73% in Experiment I and III (Figure 10A, the numerical results are reported in Table 3). The main difference between experiments with and without ball is the σ standard deviation. In the final period, the experiments with the ball present a larger standard deviation which is 30% (without cerebellum) and 19% (with cerebellum) respect to the NO ball-case.

3.2. Wrist Yaw

The wrist yaw joint is the most affected by the cerebellum action. In Figure 6, it is evident how with only the PID contribution $\vartheta_{c,1}(t)$ presents a constant and large offset with respect to $\vartheta_{r,1}(t)$. As soon as the cerebellum contribution $\Delta\tau_1^{DCN}$ grows (around the 50 s, Figures 7C,D) the error descends (Figures 7A,B). The mean absolute error decreases by the 78% in experiment II and IV, while it only drops 1% in experiments I and III (Figure 10B). In the last period, the experiments with the ball have a standard deviation 30–33% larger than the NO ball-cases.

3.3. Wrist Pitch

On the other hand, the wrist pitch gains from the cerebellar action only when the error is larger than $e_{\vartheta_2}^{thresh}$, which is around 40–60 s (Figure 8), taking into account that the cerebellum is started at $t = 40$ s. The $\Delta\tau_1^{DCN}$ gets more silent (Figures 9C–E) when the angular position error is small (Figures 9A,B). In Figure 10C is more evident how the cerebellum accelerates the corrective action between iteration 10 and 15 where the MAE with the cerebellum (experiment II) is 17% lower respect to experiment I (in experiment IV the MAE is 16% lower respect to experiment III).

4. DISCUSSIONS

In this work, a bio-mimetic control scheme is presented in the framework of a robotic task, in which simultaneous control of the object dynamics and of the internal force exerted by the robot arm to follow a trajectory with the object attached to it is required. To address multi-joint corrective responses, we induced and combined three-joint wrist motions. Thus adaptation skills are required especially to deal with an external perturbation acting on the robot-object system. The main observation is that plastic mechanisms given by a feed-forward cerebellum-like controller effectively contribute to the learning of the dynamics model of the robot arm-object system and to the adaptive corrections in terms of torque commands applied to the joints. These cerebellar torque contributions, together with feedback (PID) torque outcome, allow the progressive error reduction by incorporating distributed synaptic plasticity based on the feedback from the actual movement.

The results about the three controlled joints showed a fast reactive control in the test cases when the cerebellum-like model is active, which is even more evident when the ball (random perturbation) is present as shown in Figures 4, 6, 8B,D. An incremental velocity control input is then provided to the controller of the system to deal with the perturbation. The purpose of considering a heterogeneous stochastic dynamical stimuli (board and ball) was to test and examine the activation of incremental learning and adaptation of the cerebellum-like controller and at the same time to confirm its coupling with the feedback control inputs. Previous studies have shown that the feedback processes are omnipresent in voluntary motor actions (Scott et al., 2015) and rapid corrective responses occur even for very small disturbances that approach the natural variability of limb motion. In human beings, these corrections commonly require increases in muscle activity generated i.e., by applied loads (Nashed et al., 2015). By analogy, a similar effect can be noticed at joint-level in our system. In the experimental situation, the joints that are more influenced by the limb dynamics (wrist prosup and yaw joints) under the effect of the table and ball increase their control input activity as represented in Figures 5, 7C,D, while the wrist pitch joint has a much more reduced activity re influenced by the limb dynamics (wrist prosup and yaw joints) under the effect of the table and ball increase their control input activity as represented in Figures 9C,D compared to the previous two joints. This phenomena is also reflected in the control input provided by the cerebellum-like model. The bigger the position error is at the beginning of the simulation with only the PID control case (experiments I and III) the more effective the cerebellar-like corrections are (experiments II and IV) as shown in Figures 5, 7, 9A,B. It should be noted that for the wrist pitch joint the PID controller leads to ~ 0.0 (rad) MAE around 40 s from the beginning of the simulation. However, among all the joints, the fundamental role of the cerebellum in motor control is confirmed by its anticipatory response for decreasing the error as it is appreciated in Figure 10. The control system achieved these result by creating up to 9 Gr receptive fields per uml at the granular level (or rather LWPR). In Figure 11, it is possible to appreciate how the IO inferior olive signals (in blue) of each ccm promptly influence the synaptic weights (in red) between the PF parallel fibers and the PC Purkinje cells (left column), and the contribution of the inferior olive itself on the DCN Deep Cerebellar nuclei corrective action (right column). In the IO-DCN connection details, the synaptic weights rapidly increment in the first tract around 40–60 s where the error is higher and then keep increasing slowly for the final adjustments. On the other hand, the PF-PC connection tends to not over-react at the beginning of the simulation around 40–60 s, while it strengthen when the error decline. We assume that this opposite influence of the IO on the synaptic weights makes possible the filtering and the dumping of any external disturbances or high error.

This control model proposes a plausible explanation on how control feedback is used by the central nervous system (CNS) to correct for intrinsic as well as external sources of disturbances. Furthermore, the bio-mimetic model represents a plausible control scheme for voluntary movements that can be generalized to control robotic agents without mayor tuning of

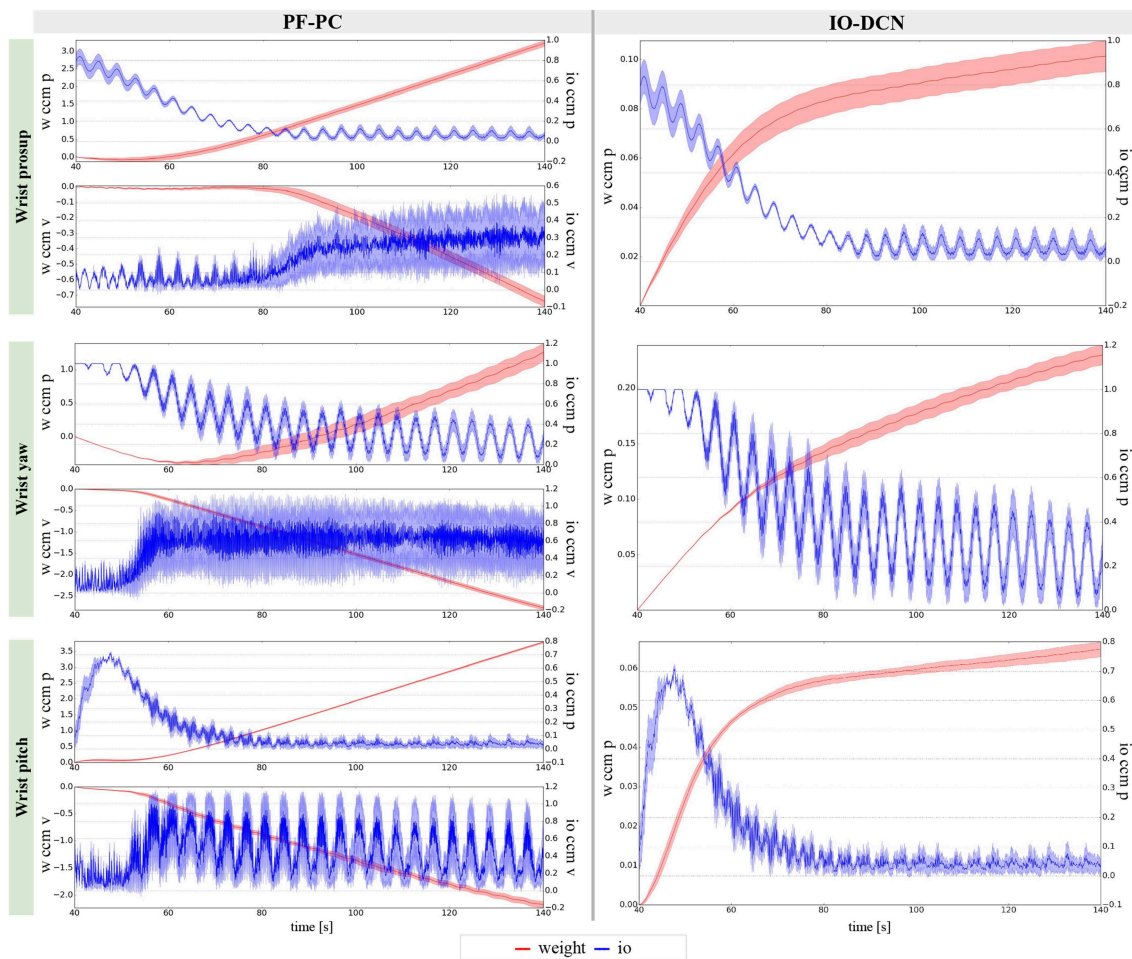


FIGURE 11 | Learning evolution of the cerebellar-like network in experiment IV: influence of the inferior olive on the PC-PF parallel fibers-Purkinje cells and IO-DCN inferior olive-Deep cerebellar nuclei connections. The plots show the results of the 20 tests in terms of mean value (solid line) and 95% confidence interval (colored area).

the parameters. Our controller with distributed plasticity allows efficient adjustment of the corrective signal regardless of the dynamic features of the robot arm and of the way the added perturbations affect the dynamics of the arm plant involved. According to this, the controller (cerebellum-like and PID) is adaptable by providing adjustable torque commands among the joints to overcome external dynamic and stochastic perturbations and to have a both fast and precise movement. This replies to our question about if the sensory-motor information extrapolation made by the cerebellum-like facilitates motor prediction and adaptation in changing conditions. It should be noted that the adaptation mechanism adopted here is not constrained to any specific plant or testing framework, and could therefore be extrapolated to other common testing paradigms.

D'Angelo et al. (2016) illustrated in their paper the schematic representation of how the core cerebellar microcircuit is wired inside the whole brain. The proposed cerebellar-like model has been designed in analogy with it. In contrast with Garrido Alcazar et al. (2013), Casellato et al. (2014), Antonietti et al. (2017), the

proposed model encodes the movement kinematics at the mossy fibers level (Ebner et al., 2011), and presents a coupling at the Purkinje layer for velocity and position terms representation. Likewise, the synaptic strengths at PC-DCN level as well the synaptic strengths at IO-DCN level are modulated by signals related to position or velocity. The mossy fibers are connected to the DCN and to some granular cells to convey the efference copy or motor command information. The IO cells are devoted to teaching signal error transmission in terms of position and velocity errors. The teaching errors modulate the synaptic strengths at PF-PC and IO-DCN levels.

Tokuda et al. (2017) postulated that high dimensionality problem (high-dimensional sensory-motor inputs vs. low training data) is accomplished by the cerebellum by regulating the synchronous firing activities of the inferior olive (IO) neurons. Though the implementation of coupling mechanisms at the inferior olive cells would be an interesting work to have a better explanation on multiple joint control. This extension could also provide additional insights into the internal connectivity

of the cerebellar microcomplex. Further investigation will be possible in the future of how specific properties of the cells, of the network topology and synaptic adaptation mechanisms complement each other in the bio-inspired architecture.

4.1. Neural Basis of Feedback Control for Voluntary Movements

Feedback control of movement is essential to guarantee movement success especially to compensate for perturbation arising from the interaction with the external world. Different brain areas (primary motor cortex, primary somatosensory cortex, cerebellum, supplementary motor area, etc.) are involved during a voluntary movement and cooperate in many levels of hierarchy. Feedback control theory might be the key for understanding how the previous areas plan and control the movement hierarchically. By using control terminology, during the voluntary movement of a limb, the primary motor cortex acts as a controller, and the limb connected to neuronal circuits becomes the controlled object.

The cerebellum learns and provides the internal models that reproduce the inverse or direct dynamics of the body part. Thanks to the cerebellar internal model learning, the primary motor cortex performs the control without an external

feedback (Koziol et al., 2014). By our simulations, we suggest that such behavior can be confirmed. Indeed, the cerebellar-like contributions drive the feedback controller toward better accuracy and precision of the movement. In the future, a visual feedback input will be considered to probe the sophistication of feedback control processing and cerebellar-like learning consolidation.

AUTHOR CONTRIBUTIONS

MC and ST conceived and designed the experiments, analyzed the data, and wrote the paper. MC implemented the architecture and performed the experiments. EA contributed to materials and analysis tools. HL and EF reviewed the paper.

FUNDING

This work has received funding from the Marie Curie project No. 705100 (Biomodular), from the EU-H2020 Framework Programme for Research and Innovation under the specific grant agreement No. 720270 (Human Brain Project SGA1), and No. 785907 (Human Brain Project SGA2).

REFERENCES

- Albus, J. S. (1971). A theory of cerebellar function. *Math. Biosci.* 10, 25–61. doi: 10.1016/0025-5564(71)90051-4
- Albus, J. S. (1972). Theoretical and experimental aspects of a cerebellar model. *Dissert. Abstr. Int.* 33.
- Annaswamy, A. M., and Narendra, K. S. (1989). “Adaptive control of simple time-varying systems,” in *Proceedings of the 28th IEEE Conference on Decision and Control* (Tampa, FL: IEEE), 1014–1018.
- Antonietti, A., Casellato, C., D’Angelo, E., and Pedrocchi, A. (2017). Model-driven analysis of eyeblink classical conditioning reveals the underlying structure of cerebellar plasticity and neuronal activity. *IEEE Trans. Neural Netw. Learn. Syst.* 28, 2748–2762. doi: 10.1109/TNNLS.2016.2598190
- Atkeson, C. G., An, C. H., and Hollerbach, J. M. (1986). Estimation of inertial parameters of manipulator loads and links. *Int. J. Robot. Res.* 5, 101–119.
- Awtar, S., Bernard, C., Boklund, N., Master, A., Ueda, D., and Craig, K. (2002). Mechatronic design of a ball-on-plate balancing system. *Mechatronics* 12, 217–228. doi: 10.1016/S0957-4158(01)00062-9
- Barto, A. G., Fagg, A. H., Sitkoff, N., and Houk, J. C. (1999). A cerebellar model of timing and prediction in the control of reaching. *Neural Comput.* 11, 565–594.
- Buonomano, D. V., and Mauk, M. D. (1994). Neural network model of the cerebellum: temporal discrimination and the timing of motor responses. *Neural Comput.* 6, 38–55.
- Caligiore, D., Pezzulo, G., Baldassarre, G., Bostan, A. C., Strick, P. L., Doya, K., et al. (2017). Consensus paper: towards a systems-level view of cerebellar function: the interplay between cerebellum, basal ganglia, and cortex. *Cerebellum* 16, 203–229. doi: 10.1007/s12311-016-0763-3
- Casellato, C., Antonietti, A., Garrido, J. A., Carrillo, R. R., Luque, N. R., Ros, E., et al. (2014). Adaptive robotic control driven by a versatile spiking cerebellar network. *PLoS ONE* 9:e112265. doi: 10.1371/journal.pone.0112265
- Casellato, C., Antonietti, A., Garrido, J. A., Ferrigno, G., D’Angelo, E., and Pedrocchi, A. (2015). Distributed cerebellar plasticity implements generalized multiple-scale memory components in real-robot sensorimotor tasks. *Front. Comput. Neurosci.* 9:24. doi: 10.3389/fncom.2015.00024
- Chapeau-Blondeau, F., and Chauvet, G. (1991). A neural network model of the cerebellar cortex performing dynamic associations. *Biol. Cybernet.* 65, 267–279.
- Chen, C.-S. (2009). “Sliding-mode-based fuzzy cmac controller design for a class of uncertain nonlinear system,” in *2009 IEEE International Conference on Systems, Man and Cybernetics* (San Antonio, TX: IEEE), 3030–3034.
- D’Angelo, E., Antonietti, A., Casali, S., Casellato, C., Garrido, J. A., Luque, N. R., et al. (2016). Modeling the cerebellar microcircuit: new strategies for a long-standing issue. *Front. Cell. Neurosci.* 10:176. doi: 10.3389/fncel.2016.00176
- Dean, P., Porrill, J., Ekerot, C. F., and Jörntell, H. (2010). The cerebellar microcircuit as an adaptive filter: experimental and computational evidence. *Nat. Rev. Neurosci.* 11, 30–43. doi: 10.1038/nrn2756
- Ebner, T. J., Hewitt, A. L., and Popa, L. S. (2011). What features of limb movements are encoded in the discharge of cerebellar neurons? *Cerebellum* 10, 683–693. doi: 10.1007/s12311-010-0243-0
- Falotico, E., Vannucci, L., Ambrosano, A., Albanese, U., Ulbrich, S., Vasquez Tieck, J. C., et al. (2017). Connecting artificial brains to robots in a comprehensive simulation framework: the neurobotics platform. *Front. Neurobot.* 11:2. doi: 10.3389/fnbot.2017.00002
- Farrell, J. A., and Polycarpou, M. M. (2006). *Adaptive Approximation Based Control: Unifying Neural, Fuzzy and Traditional Adaptive Approximation Approaches*, Vol 48. Hoboken, NJ: John Wiley & Sons.
- Francis, B. A., and Wonham, W. M. (1976). The internal model principle of control theory. *Automatica* 12, 457–465.
- Fujiki, S., Aoi, S., Funato, T., Tomita, N., Senda, K., and Tsuchiya, K. (2015). Adaptation mechanism of interlimb coordination in human split-belt treadmill walking through learning of foot contact timing: a robotics study. *J. R. Soc. Interface* 12:20150542. doi: 10.1098/rsif.2015.0542
- Fujita, M. (1982). Simulation of adaptive modification of the vestibulo-ocular reflex with an adaptive filter model of the cerebellum. *Biol. Cybernet.* 45, 207–214.
- Garrido Alcazar, J. A., Luque, N. R., D’Angelo, E., and Ros, E. (2013). Distributed cerebellar plasticity implements adaptable gain control in a manipulation task: a closed-loop robotic simulation. *Front. Neural Circuits* 7:159. doi: 10.3389/fncir.2013.00159
- Glanz, F. H., Miller, W. T., and Kraft, L. G. (1991). “An overview of the cmac neural network,” in *[1991 Proceedings] IEEE Conference on Neural Networks for Ocean Engineering* (Washington, DC: IEEE), 301–308.
- Guan, J., Lin, C.-M., Ji, G.-L., Qian, L.-W., and Zheng, Y.-M. (2018). Robust adaptive tracking control for manipulators based on a task fuzzy

- cerebellar model articulation controller. *IEEE Access* 6, 1670–1679. doi: 10.1109/ACCESS.2017.2779940
- He, W., Chen, Y., and Yin, Z. (2016). Adaptive neural network control of an uncertain robot with full-state constraints. *IEEE Trans. Cybernet.* 46, 620–629. doi: 10.1109/TCYB.2015.2411285
- He, W., Huang, B., Dong, Y., Li, Z., and Su, C. Y. (2018). Adaptive neural network control for robotic manipulators with unknown deadzone. *IEEE Trans. Cybernet.* 48, 2670–2682. doi: 10.1109/TCYB.2017.2748418
- Houk, J. C., and Wise, S. P. (1995). Distributed modular architectures linking basal ganglia, cerebellum, and cerebral cortex: their role in planning and controlling action. *Cereb. Cortex* 5, 95–110.
- Ito, M. (1984). *The Cerebellum and Neural Control*. New York, NY: Raven Press.
- Ito, M. (1997). “Cerebellar microcomplexes,” in *International Review of Neurobiology*, Vol 41 (Cambridge, MA: Academic Press; Elsevier), 475–487.
- Ito, M. (2008). Control of mental activities by internal models in the cerebellum. *Nat. Rev. Neurosci.* 9:304. doi: 10.1038/nrn2332
- Jiang, Y., Yang, C., Wang, M., Wang, N., and Liu, X. (2018). Bioinspired control design using cerebellar model articulation controller network for omnidirectional mobile robots. *Adv. Mech. Eng.* 10:1687814018794349. doi: 10.1177/1687814018794349
- Kawato, M. (1990). “Feedback-error-learning neural network for supervised motor learning,” in *Advanced Neural Computers*, ed R. Eckmiller (Amsterdam: Elsevier; North-Holland), 365–372. doi: 10.1016/B978-0-444-88400-8.50047-9
- Kawato, M. (1999). Internal models for motor control and trajectory planning. *Curr. Opin. Neurobiol.* 9, 718 – 727.
- Kawato, M., Furukawa, K., and Suzuki, R. (1987). A hierarchical neural-network model for control and learning of voluntary movement. *Biol. Cybernet.* 57, 169–185.
- Khalil, W., and Dombre, E. (2002). *Modeling, Identification and Control of Robots*. London, UK: Butterworth-Heinemann.
- Kocijan, J., Murray-Smith, R., Rasmussen, C. E., and Girard, A. (2004). “Gaussian process model based predictive control,” in *Proceedings of the 2004 American Control Conference*, Vol 3 (Boston, MA: IEEE), 2214–2219.
- Koenig, N. P., and Howard, A. (2004). “Design and use paradigms for gazebo, an open-source multi-robot simulator,” in *IROS*, Vol 4 (Sendai: Citeseer), 2149–2154.
- Kozioł, L. F., Budding, D., Andreasen, N., D’Arrigo, S., Bulgheroni, S., Imamizu, H., et al. (2014). Consensus paper: the cerebellum’s role in movement and cognition. *Cerebellum* 13, 151–177. doi: 10.1007/s12311-013-0511-x
- Levinson, S. E., Silver, A. F., and Wendt, L. A. (2010). “Vision based balancing tasks for the icub platform: a case study for learning external dynamics,” in *Workshop on Open Source Robotics: iCub & Friends. IEEE Int’l. Conf. on Humanoid Robotics* (Nashville, TN).
- Lin, C.-M., and Chen, C.-H. (2007). Robust fault-tolerant control for a biped robot using a recurrent cerebellar model articulation controller. *IEEE Trans. Syst. Man Cybernet. Part B* 37, 110–123. doi: 10.1109/TSMCB.2006.881905
- Ljung, L. (2007). *Identification of Nonlinear Systems*. Linköping: Linköping University Electronic Press.
- Luo, D., Hu, F., Zhang, T., Deng, Y., and Wu, X. (2018). How does a robot develop its reaching ability like human infants do? *IEEE Trans. Cognit. Dev. Syst.* 10, 795–809. doi: 10.1109/TCDS.2018.2861893
- Luque, N. R., Garrido, J. A., Carrillo, R. R., D’Angelo, E., and Ros, E. (2014). Fast convergence of learning requires plasticity between inferior olive and deep cerebellar nuclei in a manipulation task: a closed-loop robotic simulation. *Front. Comput. Neurosci.* 8:97. doi: 10.3389/fncom.2014.00097
- Luque, N. R., Garrido, J. A., Carrillo, R. R., Olivier, J.-M. C., and Ros, E. (2011). Cerebellar-like corrective model inference engine for manipulation tasks. *IEEE Trans. Syst. Man. Cybernet. Part B* 41, 1299–1312. doi: 10.1109/TSMCB.2011.2138693
- Luque, N. R., Garrido, J. A., Naveros, F., Carrillo, R. R., D’Angelo, E., and Ros, E. (2016). Distributed cerebellar motor learning: a spike-timing-dependent plasticity model. *Front. Comput. Neurosci.* 10:17. doi: 10.3389/fncom.2016.00017
- Marr, D. (1969). A theory of cerebellar cortex. *J. Physiol.* 202, 437–470.
- Mauk, M. D., and Donegan, N. H. (1997). A model of pavlovian eyelid conditioning based on the synaptic organization of the cerebellum. *Learn. Memory* 4, 130–158.
- Metta, G., Sandini, G., and Konczak, J. (1999). A developmental approach to visually-guided reaching in artificial systems. *Neural Netw.* 12, 1413–1427.
- Miller, W. (1987). Sensor-based control of robotic manipulators using a general learning algorithm. *IEEE J. Robot. Automat.* 3, 157–165.
- Miller, W. T., Glanz, F. H., and Kraft, L. G. (1990). Cmas: an associative neural network alternative to backpropagation. *Proc. IEEE* 78, 1561–1567.
- Miyamoto, H., Kawato, M., Setoyama, T., and Suzuki, R. (1988). Feedback-error-learning neural network for trajectory control of a robotic manipulator. *Neural Netw.* 1, 251–265.
- Nakanishi, J., Cory, R., Mistry, M., Peters, J., and Schaal, S. (2008). Operational space control: a theoretical and empirical comparison. *Int. J. Robot. Res.* 27, 737–757. doi: 10.1177/0278364908091463
- Nakanishi, J., Farrell, J. A., and Schaal, S. (2005). Composite adaptive control with locally weighted statistical learning. *Neural Netw.* 18, 71–90. doi: 10.1016/j.neunet.2004.08.009
- Nakanishi, J., and Schaal, S. (2004). Feedback error learning and nonlinear adaptive control. *Neural Netw.* 17, 1453–1465. doi: 10.1016/j.neunet.2004.05.003
- Narendra, K. S., and Annaswamy, A. M. (1987). Persistent excitation in adaptive systems. *Int. J. Cont.* 45, 127–160.
- Narendra, K. S., and Mukhopadhyay, S. (1991a). Associative learning in random environments using neural networks. *IEEE Trans. Neural Netw.* 2, 20–31.
- Narendra, K. S., and Mukhopadhyay, S. (1991b). “Multilevel control of a dynamical systems using neural networks,” in [1991] *Proceedings of the 30th IEEE Conference on Decision and Control* (Brighton, UK: IEEE), 170–171.
- Narendra, K. S., and Mukhopadhyay, S. (1997). Adaptive control using neural networks and approximate models. *IEEE Trans. Neural Netw.* 8, 475–485.
- Narendra, K. S., and Parthasarathy, K. (1991). Gradient methods for the optimization of dynamical systems containing neural networks. *IEEE Trans. Neural Netw.* 2, 252–262.
- Nashed, J. Y., Kurtzer, I. L., and Scott, S. H. (2015). Context-dependent inhibition of unloaded muscles during the long-latency epoch. *J. Neurophysiol.* 113, 192–202. doi: 10.1152/jn.00339.2014
- Nguyen-Tuong, D., and Peters, J. (2011). Model learning for robot control: a survey. *Cogn. Proc.* 12, 319–340. doi: 10.1007/s10339-011-0404-1
- Nguyen-Tuong, D., Seeger, M., and Peters, J. (2009). Model learning with local gaussian process regression. *Adv. Robot.* 23, 2015–2034. doi: 10.1163/016918609X12529286896877
- Nowak, D. A., Topka, H., Timmann, D., Boecker, H., and Hermsdörfer, J. (2007). The role of the cerebellum for predictive control of grasping. *Cerebellum* 6:7. doi: 10.1080/14734220600776379
- Ojeda, I. B., Tolu, S., and Lund, H. H. (2017). “A scalable neuro-inspired robot controller integrating a machine learning algorithm and a spiking cerebellar-like network,” in *Biomimetic and Biohybrid Systems - 6th International Conference, Living Machines 2017, Stanford, CA, USA, July 26-28, 2017, Proceedings* (Stanford, CA), 375–386.
- Patino, H. D., Carelli, R., and Kuchen, B. R. (2002). Neural networks for advanced control of robot manipulators. *IEEE Trans. Neural Netw.* 13, 343–354. doi: 10.1109/72.991420
- Porri, J., and Dean, P. (2007). Recurrent cerebellar loops simplify adaptive control of redundant and nonlinear motor systems. *Neural Comput.* 19, 170–193. doi: 10.1162/neco.2007.19.1.170
- Probst, D., Maass, W., Markram, H., and Gewaltig, M.-O. (2012). “Liquid computing in a simplified model of cortical layer iv: Learning to balance a ball,” in *Artificial Neural Networks and Machine Learning – ICANN 2012*, eds A. E. P. Villa, W. Duch, P. Erdi, F. Masulli, and G. Palm (Berlin; Heidelberg: Springer Berlin Heidelberg), 209–216.
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., et al. (2009). “Ros: an open-source robot operating system,” in *ICRA Workshop on Open Source Software*, Vol. 3 (Kobe: IEEE), 5.
- Schmahmann, J. D. (2004). Disorders of the cerebellum: ataxia, dysmetria of thought, and the cerebellar cognitive affective syndrome. *J. Neuropsych. Clin. Neurosci.* 16, 367–378. doi: 10.1176/jnp.16.3.367
- Scott, S. H., Cluff, T., Lowrey, C. R., and Takei, T. (2015). Feedback control during voluntary motor actions. *Curr. Opin. Neurobiol.* 33, 85 – 94. doi: 10.1016/j.conb.2015.03.006
- Sontag, E. D. (1992). Feedback stabilization using two-hidden-layer nets. *IEEE Trans. Neural Netw.* 3, 981–990.

- Ting, J. A., Mistry, M., Peters, J., Schaal, S., and Nakanishi, J. (2006). "A bayesian approach to nonlinear parameter identification for rigid body dynamics," in *Robotics: Science and Systems*, eds G. S. Sukhatme, S. Schaal, W. Burgard, and D. Fox (Philadelphia, PA; Cambridge, MA: MIT Press), 32–39.
- Tokuda, I. T., Hoang, H., and Kawato, M. (2017). New insights into olivocerebellar circuits for learning from a small training sample. *Curr. Opin. Neurobiol.* 46, 58–67. doi: 10.1016/j.conb.2017.07.010
- Tolu, S., Vanegas, M., Garrido, J. A., Luque, N. R., and Ros, E. (2013). Adaptive and predictive control of a simulated robot arm. *Int. J. Neural Syst.* 23:1350010. doi: 10.1142/S012906571350010X
- Tolu, S., Vanegas, M., Luque, N. R., Garrido, J. A., and Ros, E. (2012). Bio-inspired adaptive feedback error learning architecture for motor control. *Biol. Cybernet.* 106, 507–522. doi: 10.1007/s00422-012-0515-5
- Ugur, E., Nagai, Y., Sahin, E., and Oztop, E. (2015). Staged development of robot skills: behavior formation, affordance learning and imitation with motionese. *IEEE Trans. Autonom. Mental Dev.* 7, 119–139. doi: 10.1109/TAMD.2015.2426192
- Vannucci, L., Tolu, S., Falotico, E., Dario, P., Lund, H. H., and Laschi, C. (2016). "Adaptive gaze stabilization through cerebellar internal models in a humanoid robot," in *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics* (Singapore), 25–30.
- Vernon, D., Metta, G., and Sandini, G. (2007). A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents. *IEEE Trans. Evol. Comput.* 11, 151–180. doi: 10.1109/TEVC.2006.890274
- Vijayakumar, S., and Schaal, S. (2000). "Locally weighted projection regression: an o(n) algorithm for incremental real time learning in high dimensional space," in *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000)* (Stanford, CA), 288–293.
- Weng, J. (2002). "A theory for mentally developing robots," in *Proceedings 2nd International Conference on Development and Learning. ICDL 2002* (Cambridge, MA: IEEE), 131–140.
- Weng, J., Evans, C., Hwang, W. S., and Lee, Y.-B. (1999a). "The developmental approach to artificial intelligence: Concepts, developmental algorithms and experimental results," in *NSF Design and Manufacturing Grantees Conference, Long Beach, CA*.
- Weng, J., and Hwang, W.-S. (2006). From neural networks to the brain: Autonomous mental development. *IEEE Comput. Intell. Mag.* 1, 15–31. doi: 10.1109/MCI.2006.1672985
- Weng, J., Hwang, W. S., Zhang, Y., and Evans, C. (1999b). "Developmental robots: theory, method and experimental results," in *Proceedings 2nd International Conference on Humanoid Robots* (Tokyo: Citeseer), 57–64.
- Weng, J., McClelland, J., Pentland, A., Sporns, O., Stockman, I., Sur, M., and Thelen, E. (2000). *Computational Autonomous Mental Development*. Citeseer.
- Wittenmark, B. (1995). "Adaptive dual control methods: an overview," in *Adaptive Systems in Control and Signal Processing 1995*, ed Cs. Bányász (Elsevier), 67–72.
- Wolpert, D. M., Miall, R. C., and Kawato, M. (1998). Internal models in the cerebellum. *Trends Cogn. Sci.* 2, 338–347.
- Yamazaki, T., and Tanaka, S. (2007). The cerebellum as a liquid state machine. *Neural Netw.* 20, 290–297. doi: 10.1016/j.neunet.2007.04.004
- Zhang, T., Ge, S. S., and Hang, C. C. (2000). Adaptive neural network control for strict-feedback nonlinear systems using backstepping design. *Automatica* 36, 1835–1846. doi: 10.1016/S0005-1098(00)00116-3

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Capolei, Angelidis, Falotico, Lund and Tolu. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Combining Evolutionary and Adaptive Control Strategies for Quadruped Robotic Locomotion

Elisa Massi^{1*}, Lorenzo Vannucci¹, Ugo Albanese¹, Marie Claire Capolei², Alexander Vandesompele³, Gabriel Urbain³, Angelo Maria Sabatini², Joni Dambre¹, Cecilia Laschi¹, Silvia Tolu^{2†} and Egidio Falotico^{1†}

¹ The BioRobotics Institute, Scuola Superiore Sant'Anna, Pontedera, Italy, ² Automation and Control Group, Department of Electrical Engineering, Technical University of Denmark, Copenhagen, Denmark, ³ AIRO, Electronics and Information Systems Department, Ghent University - imec, Ghent, Belgium

OPEN ACCESS

Edited by:

Mario Senden,
Maastricht University, Netherlands

Reviewed by:

Guoyuan Li,
NTNU Ålesund, Norway
Chengju Liu,
Yangpu Hospital, Tongji University,
China

*Correspondence:

Elisa Massi
elisa.massi@santannapisa.it

[†]These authors have contributed
equally to this work

Received: 15 April 2019

Accepted: 14 August 2019

Published: 29 August 2019

Citation:

Massi E, Vannucci L, Albanese U, Capolei MC, Vandesompele A, Urbain G, Sabatini AM, Dambre J, Laschi C, Tolu S and Falotico E (2019) Combining Evolutionary and Adaptive Control Strategies for Quadruped Robotic Locomotion. *Front. Neurobot.* 13:71. doi: 10.3389/fnbot.2019.00071

In traditional robotics, model-based controllers are usually needed in order to bring a robotic plant to the next desired state, but they present critical issues when the dimensionality of the control problem increases and disturbances from the external environment affect the system behavior, in particular during locomotion tasks. It is generally accepted that the motion control of quadruped animals is performed by neural circuits located in the spinal cord that act as a Central Pattern Generator and can generate appropriate locomotion patterns. This is thought to be the result of evolutionary processes that have optimized this network. On top of this, fine motor control is learned during the lifetime of the animal thanks to the plastic connections of the cerebellum that provide descending corrective inputs. This research aims at understanding and identifying the possible advantages of using learning during an evolution-inspired optimization for finding the best locomotion patterns in a robotic locomotion task. Accordingly, we propose a comparative study between two bio-inspired control architectures for quadruped legged robots where learning takes place either during the evolutionary search or only after that. The evolutionary process is carried out in a simulated environment, on a quadruped legged robot. To verify the possibility of overcoming the reality gap, the performance of both systems has been analyzed by changing the robot dynamics and its interaction with the external environment. Results show better performance metrics for the robotic agent whose locomotion method has been discovered by applying the adaptive module during the evolutionary exploration for the locomotion trajectories. Even when the motion dynamics and the interaction with the environment is altered, the locomotion patterns found on the learning robotic system are more stable, both in the joint and in the task space.

Keywords: evolutionary algorithm, bio-inspired controller, cerebellum-inspired algorithm, robotic locomotion, neurobotics, central pattern generator

1. INTRODUCTION

From the outside, locomotion appears to be performed spontaneously and effortlessly by both animals and humans, but a complex neural system controls it. Movements are mainly controlled by the Central Nervous System (CNS) which generates commands at a cortical and spinal level and integrate those commands based on different sensory feedback. All the muscular activation and coordination processes can be unexpectedly produced without the need for conscious control (Takakusaki, 2013). In quadrupeds, the neural control of locomotion happens along with all the CNS, involving the contribution of cortical areas as the pre-motor and motor cortices and also more peripheral areas such as the spinal cord. In particular, the existence of a Central Pattern Generator (CPG) in the spinal cord has been first demonstrated in the middle of the twentieth century (Hughes and Wiersma, 1960). It is a network of cells that generates basic locomotion patterns by the repetitive contraction of different muscle groups thanks to its periodic oscillations in exciting or inhibiting certain motoneurons.

The cerebellum plays an important role, too, in both quadruped and human locomotion. It improves the accuracy in motor learning, adaptation and cognition on the control commands from the motor cortex (Ito, 2000), computing the inverse dynamics of a body component and delivering a contribution to the present neural signals from the motor cortex (Kawato and Gomi, 1992; Wolpert et al., 1998). In nature, the optimal locomotion strategies are discovered by the long process of evolution. Evolution bases its research on a no-random selection of randomly generated individuals and the final evaluation strictly depends on the agent and its interaction with the surrounding environment. By inspiration from the biological evolution process, the new concept called *Embodied intelligence* or *Embodied brain* emerged more recently (Starzyk, 2008). The idea conveys the importance of the body to properly learn the interaction between intelligence and outer world. Evolution and learning operate on different time scales but both are forms of biological adaptation from which is important to take inspiration from. Evolution reacts to slow environmental changes whereas learning produces adaptive reactions in an individual during its lifetime (Pratihara, 2003).

In robotics, finding effective locomotion strategies has always been a challenge and this task gets even more complicated when the environmental conditions change. To face dynamical external conditions, different methods have been developed, in robotics, and leg-based motion is one of the most effective locomotion mechanism to deal with changing terrains (Full and Koditschek, 1999). However, legged locomotion is usually very complex to be modeled and controlled due to the high-dimensional, nonlinear and dynamically coupled interactions between the robot and the environment. New approaches, employing synergies and symmetries, have been proposed to simplify the problem and decrease its redundancy (Ijspeert, 2008). In some cases, bio-inspired CPG-based controllers have been used to prove how a primitive neural circuit used for generating periodic motion patterns can be extended for generating different types of locomotion. For instance, the research work from Ijspeert et al.

(2007) shows a CPG model which switches between swimming-like to walking-like locomotion by just changing a few parameters of the model, as the oscillation threshold of the system.

The need for refined motor control pushed bio-inspired robotics to deeply study the cerebellar contribution and design mathematical models to mimic some of its biological functions in motion control (Wolpert et al., 1998). Cerebellar-like neuro-controllers have also been implemented recently. The cerebellum exploits long-term synaptic plasticity (LTP) to store information about body-object dynamics and to generate internal models of movements. This evidence has been studied by Garrido Alcazar et al. (2013) and implemented for adaptable gain control for robotic manipulation tasks. In this case, it is useful to have cerebellar corrective torques which are self-adaptable, operate over multiple time scales and improve learning accuracy, in order to minimize the motor error. An error-dependent signal operating as a teaching contribution is needed for this purpose.

The interesting interaction between CPG-based oscillators and cerebellar inspired networks has been implemented in bio-inspired control design, too. In the research work proposed by Fujiki et al. (2015), the spinal model generates rhythmic motor commands using an oscillator network based on a Central Pattern Generator and modulates the commands formulated in immediate response to foot contact, while the cerebellar model modifies motor commands, through learning, based on error information related to the difference between the predicted and the actual foot contact timings of each leg.

Another interesting research branch is *evolutionary robotics* which is becoming a very popular approach in the search for new robotic morphology and controllers. The main advantage of this approach is that it is “*prejudice-free*,” in the sense that it mainly depends on the behavior of an agent in interaction with the external environment. In fact, genetic algorithms derive from the kind of long-term adaptation that humans share with other species. This idea of adaptation is meant as a relational property that involves the agent, its environment, and the maintenance of some constraints and can be in the wide sense described as the ability of an agent of interacting with its environment to maintain some existence constraints. Thus, the idea is exploiting the sensorimotor interactions with a dynamic environment to minimize the prior assumptions that are built into a “human-made” model, which reduces the capability of the model itself to count for new and unknown relevant features or artifacts in the system (Harvey et al., 2005). Many enhancements have been done recently, in finding either optimal robotic morphologies (Corucci et al., 2016) and adaptable robotic brains (Flozano et al., 2008). Hence, exploiting the interplay *robot-environment*, the evolutionary approach represents a model-free method to discover optimal locomotion patterns based on the interaction robot-terrain.

In this work, we present a new bio-inspired and model-free control architecture for quadruped robotic locomotion which takes advantages from the collaboration of evolution and adaptation. The evolutionary approach part for optimizing the Central Pattern Generator model on a simulated robot has already been investigated and tested (Urbain et al., 2018), while the cerebellar-like adaptive controller has been proven to be

effective on both control of voluntary movements, such as control of a robotic arm (Tolu et al., 2012, 2013), and control of reflexes, such as in gaze stabilization tasks (Vannucci et al., 2016, 2017).

In comparison to the previous research works, where the evolutionary scenario is applied on the CPG parameters of the quadruped robot Tigrillo (Urbain et al., 2018), we proposed a comparative research proving the advantages of performing the evolution on an adaptive quadruped system *body + brain*. In the controller, the adaptive part is a cerebellar-inspired circuit (Tolu et al., 2012), which presents a modular structure for the quadruped locomotion task case. Further, for the first time, the paper shows the benefits of using the Cerebellar-inspired layer, already proposed by Ojeda et al. (2017), for robotic locomotion task.

To conclude and extend the result to a more general perspective, it is analyzed a comparison to the case where the evolution is performed just on the *body*, while the adaptive control part is included after the definition of the locomotion patterns, so after the findings of the locomotion trajectories by the evolutionary algorithm.

A comparison of the locomotion stability of the two bio-inspired controllers is then performed under different experimental constraints, to assess the generalizability of the results. These final experiments are very important because of the difficulty to transfer results found in simulation to the real world due to differences in sensing, actuation, and in the dynamic interactions between robot and environment. This phenomenon is called *reality gap* (Lipson and Pollack, 2000) and it is even more evident in adaptive approaches, where the control system is gradually designed and tuned through the repeated interactions between the agent and the surrounding scenario. Robots might evolve to match the specificities of the simulation, which differ from the real-world constraints. To prevent this problem, many approaches can be possible, such as adding independent noise to the values of the sensors or changing the robot dynamic model and its interaction with the environment (Nolfi et al., 2000; Vandesompele et al., 2019). In comparison to the classical approach where this simulation variability is added during the evolutionary optimization, in this research, the possibility of overcoming the reality gap and the transferability of the approach is demonstrated afterwards. Furthermore, to test the robustness of the proposed control architecture in the interaction with the environment, the static contact friction with the ground is changed during the test experiments. Usually, adaptive closed-loop CPG are exploited to counteract the changes in the environment (Kousuke et al., 2007; Ryu et al., 2010) while, in this research work, the learning and the adaptation of a cerebellar-inspired control module (Tolu et al., 2012) are applied instead to face the dynamically changing interaction with the external world.

The paper is structured as follows: in section 2 we describe the architecture of the controller, the evolutionary process employed and the implementation details; in section 3 we show the results of the evolutionary procedure and of the subsequent tests that have been performed; finally, in section 4 we discuss the obtained results and we draw the conclusions on the advantages of combining evolutionary processes and adaptive control.

2. MATERIALS AND METHODS

In this work, a bio-inspired control architecture is implemented for the quadruped configuration of Fable robot (Pacheco et al., 2014), simulated on the NeuroRobotics Platform (Falotico et al., 2017).

Figure 1 shows the system which consists of two parts: the controller, which is a simplified model of the CNS, comprising the CPG and the cerebellar circuit, and a simulated model of a quadruped robot, the Fable robot (Pacheco et al., 2014).

The robot has two degrees of freedom (DoF) for each leg (**Figure 2A**), but only one is actuated (the hip joint), while keeping the other fixed (**Figure 2B**) in order to reduce the number of parameters and simplifying the evolutionary process. This simplification does not pose a problem, as locomotion patterns can still be achieved by only using the hip joints.

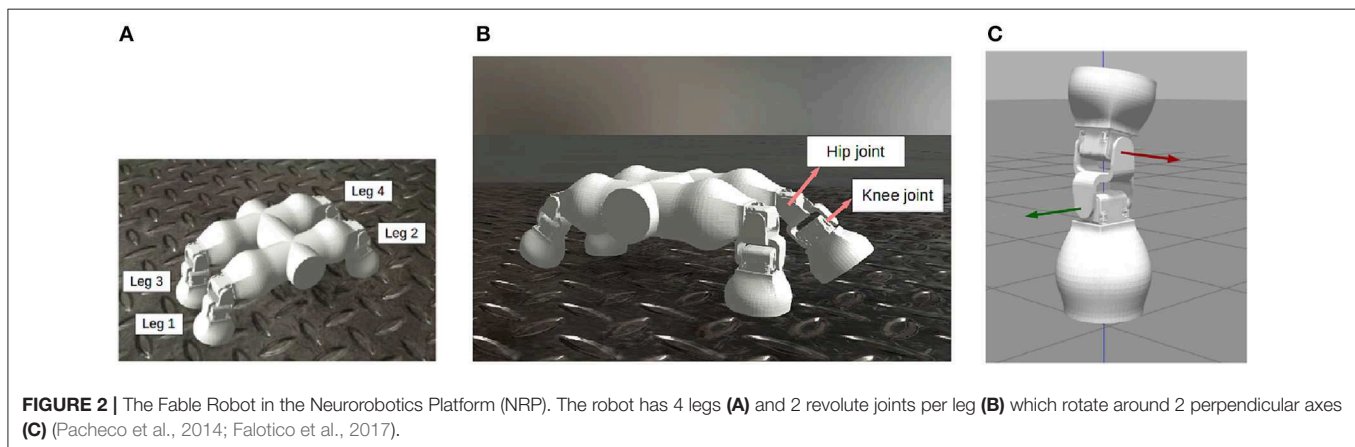
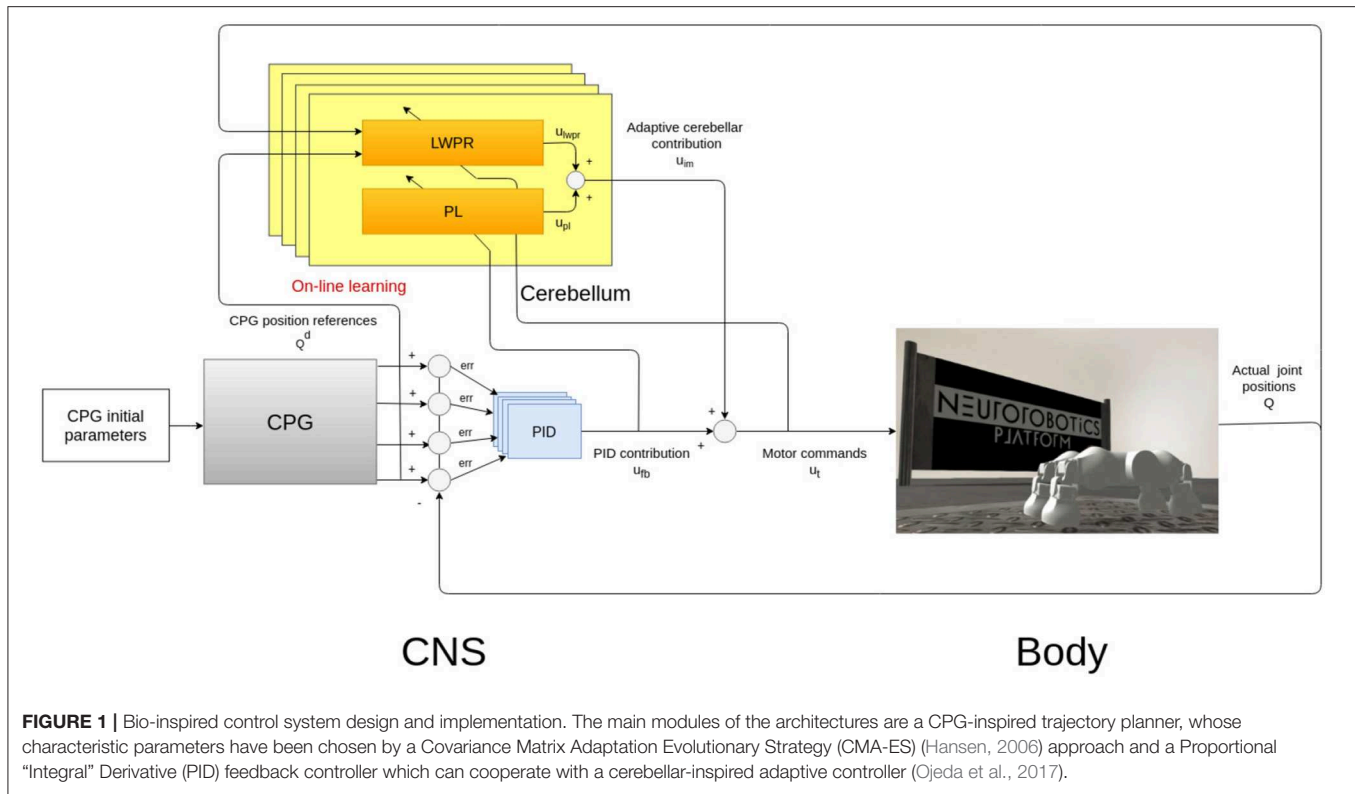
2.1. Central Pattern Generator (CPG)

In quadruped biological systems, simple locomotion can be generated as a low-level brain function, in the spinal cord, in the form of CPG. The term *central* indicates that there is no need for peripheral sensory feedback to generate the rhythms. From a control point of view, the CPG has also very interesting properties such as distributed control and modulation of locomotion by simple high-level commands (Ijspeert, 2008).

In our system, this biological neural function is mathematically modeled as a network of coupled non-linear oscillators and they are represented as the gray box in **Figure 1** (Gay et al., 2013). These oscillators are then used to plan the angular excursion in time of the hip joints of a quadruped robot (**Figure 2**). The benefits of using these oscillators lie in the fact that they are controlled by a low number of parameters that specifically affect certain aspects of the locomotion pattern. For instance, one of the most relevant parameters is the duty cycle (d in Equation 4) which controls the shape of a skewed sine wave modulating the protraction-retraction of the hip joint of the robot as shown in the systems of equations 1-4.

The CPG module is the main block involved in the evolutionary procedure (Sect. 2.3) and it is implemented in open-loop in the control architecture.

The initial parameters and the boundaries of the oscillators (**Table 1**), employed as a CPG, are selected to be a general starting point for the optimization algorithm. In defining the variables of the CPG oscillators, a difference between the front and hind legs is made to better characterize the morphology of the robot and to follow the default specifications of the work by Gay et al. (2013). These variables are the deterministic specifications which induce a certain type of locomotion for the Fable robot. Indeed, the locomotion patterns represent the *phenotype* for the evolutionary process, which means that they are the observable characteristics resulting from the interaction of the genotype of the robot with the environment. Equally, the CPG parameters (**Table 1**) represent the *genotype* which is evolved and mutated through multiple generations, whose expression are de facto the locomotion patterns (phenotype). In fact, to not steer the evolution toward a limited area in the space of the possible genetic outcomes, the generalizability and unbiasedness of the



starting values of the genotype are fundamental.

The selected parameters are listed in **Table 1**, where their initial values, boundaries and final optimal results are presented.

Here below, the equations of the unit oscillators model for the i -th robotic hip, with $\phi_{2\pi} = \phi_i(\text{mod } 2\pi)$:

$$\dot{r}_i = \gamma (\mu_i - r_i^2) r_i \quad (1)$$

$$\dot{\phi}_i = \omega_i + \sum_{j=1}^4 w_{ij} \sin(\phi_j - \phi_i - \psi_{ij}) \quad (2)$$

$$\theta_i = r_i \cos(\phi_{L_i}) + o_i \quad (3)$$

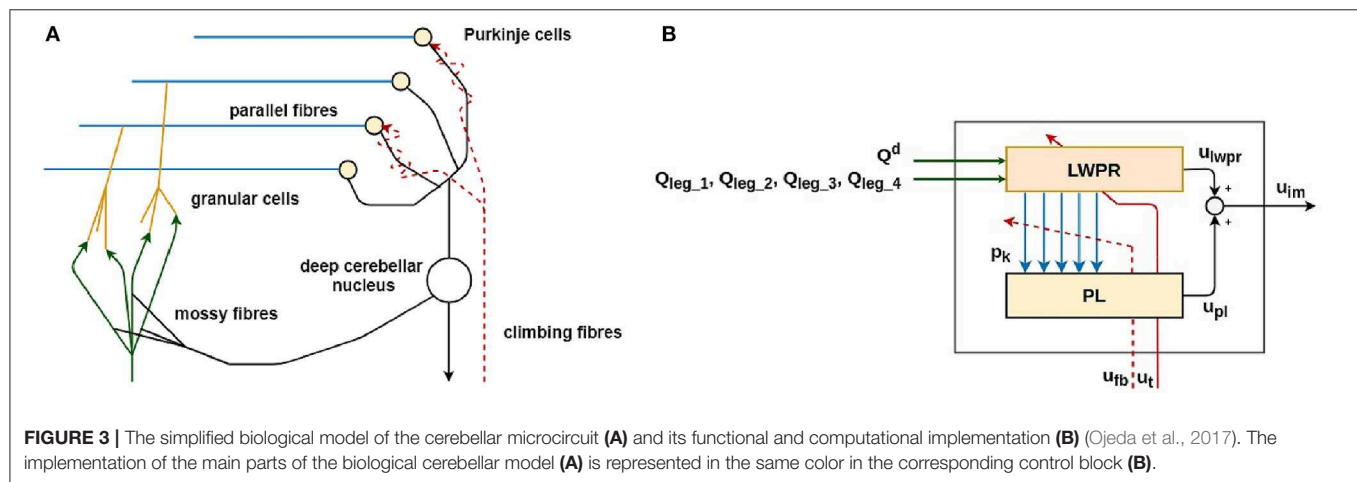
$$\phi_{L_i} = \begin{cases} \frac{\phi_{2\pi}}{2d_i} & \text{if } \phi_{2\pi} < 2\pi d_i \\ \frac{\phi_{2\pi} + 2\pi(1-2d)}{2(1-d_i)} & \text{otherwise} \end{cases} \quad (4)$$

r is the radius of the hip oscillator, μ is its hip target amplitude, ω its frequency, ϕ its phase, o its offset and θ its output angular excursion in radians. γ is a positive gain defining the speed of convergence of the radius to the target amplitudes μ . d is the virtual duty factor since the actual duty factor depending on the robot dynamics and on parameters of the gait. The four hips of the robot are also phase-coupled to synchronize them, to achieve different gaits. More in details, the coupling between hip oscillators i and j is obtained by adding the term $w_{ij}\sin(\phi_j -$

TABLE 1 | Distinctive parameters of the coupled oscillators which define the four joint trajectories for the robot.

Parameters	Initial values	Boundaries		Results	
		Min	Max	<i>adapt-after-evo</i>	<i>adapt-in-evo</i>
CPG EVOLVED PARAMETERS					
Front legs amplitude (μ)	1.58	0.5	1.56	1.04	1.46
Hind legs amplitude (μ)	0.88	0.5	1.56	0.69	0.71
Frequency (ω)	5	1	10	4.9	8.57
Phase shift leg 1-2 (ϕ)	0.001	0	6	1.19	0.38
Phase shift leg 2-3 (ϕ)	1.14	0	6	5.9	3.42
Phase shift leg 3-4 (ϕ)	4.35	0	6	1.4	3.32
Duty cycle leg 1 (d)	0.12	0	0.9	0.88	0.29
Duty cycle leg 2 (d)	0.75	0	0.9	0.57	0.73
Duty cycle leg 3 (d)	0.40	0	0.9	0.84	0.28
Duty cycle leg 4 (d)	0.85	0	0.9	0.9	0.69
Offset left front leg (o)	−20.8	−60	60	−12.73	22.3
Offset right front leg (o)	18.53	−60	60	36.58	−10.9
Offset left hind leg (o)	−17.96	−60	60	57.24	52.47
Offset right hind leg (o)	52.72	−60	60	8.64	26.79

These parameters define the four outputs of the Central Pattern Generator described in Gay et al. (2013) and their values are evolved during the CMA-ES search for the optimal solutions (Hansen, 2006) either in the *adapt-after-evo* and in the *adapt-in-evo*.



$\phi_i - \psi_{ij}$) in Equation (2), where ψ_{ij} is the desired phase difference between the oscillators controlling hips i and j and w_{ij} is a positive gain. Eventually, ϕ_L (Equation 4) is a filter applied on the phase ϕ and $\cos(\phi_L)$ is used to compute the output angle θ of the hip oscillator.

The described CPG oscillator acts as a trajectory planner in the control architecture since coordinates the robotic motion, defining the locomotion characteristics. In quadrupeds, the neural signal which descends from the spinal cord along the motoneurons regulates the contraction of the peripheral muscle fibers (Takakusaki, 2013). To obtain a consistent motor control signal, the final signals sent to the robotic legs are joint efforts. In the case of the Fable robot, these efforts are motor torques, computed by a PID feedback controller, after the CPG planning (Figure 1).

2.2. Bio-inspired Adaptive Controller

The proposed bio-inspired controller (in light blue and yellow in Figure 1) mimics one of the cerebellar roles in locomotion: the computation of the feedback-error-learning model. The body, or a part of the body as a leg, is a physical entity whose movements are controlled by the CNS. The controlled entity can be considered as a cascade of transformations between motor command (e.g., muscle activations in the biological case and joint torques in the robotic one) and links motion (e.g., joint angular position). This cascade of transformations defines the system dynamics. The neural description, which models the transformation from the desired movement trajectory to the motor commands needed to obtain it, is called the *inverse model*. This concept explains that if the inverse model is accurate, it can be used as a feedforward controller,

making the actual trajectory be reasonably comparable to its reference (Wolpert et al., 1998).

The proposed controller is then composed by a feedback part and a bio-inspired part (Tolu et al., 2012). The feedback part element is a PID controller (in light blue in **Figure 1**), often used in engineering for torque control, while the bio-inspired one is a simplified model of a cerebellar circuit (in yellow in **Figure 1**).

The cerebellar-inspired model has the role of computing a corrective torque contribution based on the inverse model of the system. As in the biological cerebellum, a specific circuit is dedicated to the inverse model of each one of the legs, but still merging information concerning the global body/robot state. Each circuit works as a Unit Learning Machine (ULM) which encodes the internal model of a body part to more precisely perform more precise motion control (Ito, 2008).

In **Figure 3**, the simplified model of one of the four biological cerebellar microcircuits and its mathematical implementation is shown.

The main functional biological sub-parts in the cerebellar microcircuit are:

- the **Mossy fibers (MF)**: they transfer the sensory inputs to the cerebellum (green in **Figure 3**);
- the **Granular cells (GC)**: they expand the sensory information from the mossy fiber to abstract the inverse model of the body movement corresponding to the specific body part (orange in **Figure 3**);
- the **Parallel fibers (PF)**: they transmit the information from the granular cells to the Purkinje cells. This layer is shared among all the cerebellar microcircuits and represents where the information is shared among the four cerebellar modules (light blue in **Figure 3**);
- the **Purkinje cells (PC)**: they modulate the input from the granular cells, which is carrying information about the actual state of the robot. The modulation is performed thanks to teaching information coming from the inferior olives through the climbing fiber (yellow in **Figure 3**);

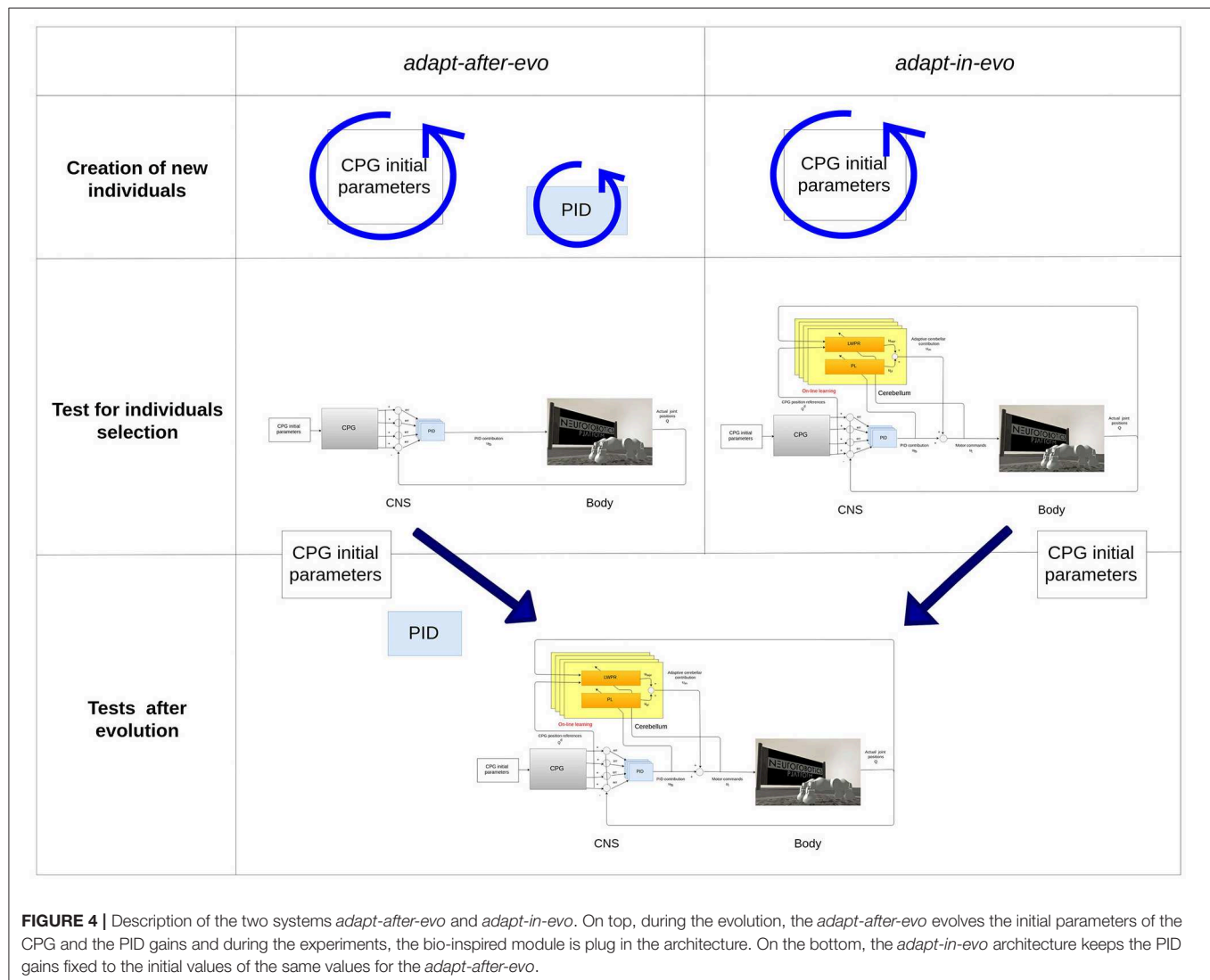


TABLE 2 | PID gains and hyper-parameters of the Cerebellum-inspired controller.

PID parameters	Adapt-after-evo		Adapt-in-evo	
	Boundaries		Result	values (fixed)
	Min	Max		
Kp	0.5	1	0.86	0.81
Ki	0.001	0.009	0.005	0.005
Kd	0.02	0.06	0.022	0.040

In the adapt-after-evo, the K_p , K_i , and K_d are evolved in the CMA-ES, as the CPG parameters in **Table 1**, while in the adapt-in-evo they are fixed as the initial values of the adapt-after-evo. Concerning the remaining four parameters, they are specifications for the learning modules of the architecture and for that, they are used just in the adapt-in-evo.

- the **Climbing fibers**: they carry the teaching signal to the Purkinje cells to modulate their activity (red dashed line in **Figure 3**);
- the **Deep nuclear cell (DCN)**: it gathers and integrates inputs from the information elaborated by the Purkinje and Granular cells. It generates the final cerebellar output (white in **Figure 3**).

The cerebellar inspired control module contains a total of 4 ULMs, one for each leg (**Figure 1**). Each ULM is considered as a single cerebellar microcircuit and the communication and synchronization through the different circuits are provided by the PFs layer and encoded as the information p_k in the Equation (3). p_k is also transferred between two sub-modules of the learning machine (in light blue in **Figures 3A,B**). Each microcircuit consists of 3 modules: a module for the *cortical layer* of the cerebellum (in orange in **Figure 3**), a module for its *molecular layer*, mainly constituted by the *Purkinje Cells Layer (PL)* (in yellow in **Figure 3**), and eventually, a model of the *Cerebellar Nuclei (DCN)* (the white circle in **Figure 3B**). All modules contribute to computing the final corrective command which constitutes the inverse model effort contribution u_{im} to the robot.

More in detail, the *cortical layer* module is implemented through the Locally Weighted Projection Regression (LWPR) algorithm. The LWPR is an algorithm for incremental nonlinear function approximation in high-dimensional spaces with redundant and irrelevant input dimensions (Vijayakumar and Schaal, 2000). This machine learning technique is computationally efficient and numerically robust thanks to its regression algorithm; it creates and combines N linear local models which perform the regression analysis in selected directions of the input space, taking inspiration from the partial least squares regression. The main advantages of using the described learning algorithm are listed in the following:

- it optimizes the role of the GC in the cerebellum, which exploit their particular plasticity to learn the dynamic model of the body for motor control (orange in **Figure 3**);
- it acts as a radial basis function filter which implies the processing of the sensory information input from the MF to the DCN (p_k in Equation 7 and in black in **Figure 3**);
- it allows rapid learning based on incremental training which perfectly fit in the specification of the designed system which

should be able to perform online learning, based on the dynamical environmental constraints;

- its learning is extremely fast and accurate since the weights of each kernel is based only on local information and its computational complexity is linear for each input information.

Each LWPR model is fed with the sensory inputs which are the reference position for the specific leg hip joint (Q^d) and the actual positions (Q_{leg_y} for y in ULMs) of all the 4 controlled joints. Then, the algorithm performs an optimal function approximation and divides the sensorimotor input space into a set of receptive fields (RFs), which represent the neurons of the cerebellar GCs layer. The RFs geometry is described by Equation (5), which describes a Gaussian weighting kernel. For each multidimensional input data point x_i , a RF activation p_k is computed, based on its distance to the center of the Gaussian kernel C_k .

$$p_k(x_i) = e^{-\frac{1}{2}((x_i - c_k)^T \cdot D_k(x_i - c_k))} \quad (5)$$

Basically, each RF activation p_k is an indicator of how often an input happens to be in the validity region of each RF linear model. The validity region is defined by a positive definite distance matrix D_k . The distance matrix is updated at each iteration according to a stochastic leave-one-out cross-validation technique to allow stable on-line learning. At each iteration, the LWPR weights p_k are sent to the cerebellar *molecular layer* model and once that the optimal centers and widths are found for each RF, the accuracy and the learning speed increase. Equation (3) has been proved to lead to a sparse code of the input data x_i and this facilitates the persistence of remaining sites of plasticity for the incremental learning process, as in the biological cerebellar circuit (Dean et al., 2010).

The output of the k_{th} RF is shown in Equation (4), where w_k is the weight vector of the RF and ϵ_k is the bias.

$$y_k(x_i) = w_k x_i + \epsilon_k \quad (6)$$

Moreover, the LWPR acts as a radial basis function filter which elaborates the sensory information and returns it as u_{lwpr} (Equation 7), that is the contribution from the cortical layer of the cerebellar microcircuit model. This contribution is modeled as a weighted linear combination of the kernels outputs $y_k(x_i)$.

$$u_{lwpr}(x_i) = \frac{\sum_{k=1}^N p_k(x_i) y_k(x_i)}{\sum_{k=1}^N p_k(x_i)} \quad (7)$$

p_k (Equation 3) also represents the contribution which is transmitted through the parallel fiber to the Purkinje Layer (PL). The parallel fibers gather all the information from the different GCs kernels. This information is multiplied by a set of weight r_k and thus, we obtain u_{pl} , the Purkinje Cell Layer (PL) output (Equation 6).

$$u_{pl}(x_i) = \sum_k r_k p_k(x_i) \quad (8)$$

The learning rule used for updating the weights in the Purkinje Cells Layer is explained in Equation (7), where the update gain δ_{r_k}

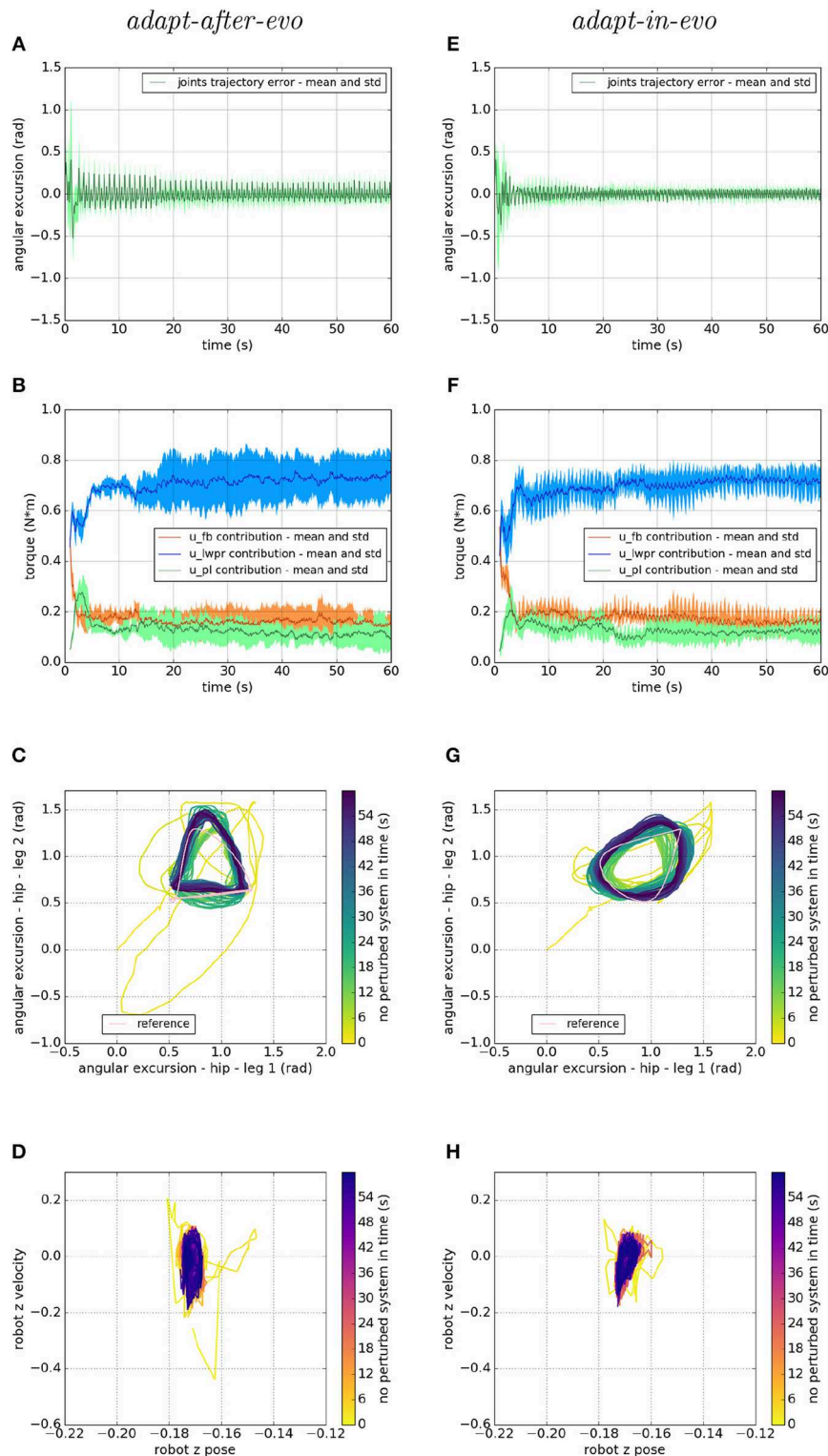


FIGURE 5 | Locomotion performance and characterization of the two systems; on the left the *adapt-after-evo* system and, on the right, the *adapt-in-evo* one.

(A,E) Represent the mean and the standard deviation of the position error of the four legs joints and (B,F) instead describe the mean and the standard deviation of the contribution ratio of the different modules of the control architecture. (C,G) Describe the periodic behavior relation between the actual joint trajectories of leg 1 and leg 2 compared to their reference values, in pink (among the other pairs of legs, the relation is periodic in a comparable way). Eventually, (D,H) represent the dynamics of the CoM of the robot, on the vertical axis to the ground. By plotting the CoM velocity against its position on the vertical axis, we can extract relevant information about the stability of the locomotion.

is computed. β is a small learning rate (usually 0.07) and $u_{fb}(x_i)$ is the motor command from the feedback part of the controller, used as teaching signal.

$$\delta_{r_k} = \beta u_{fb}(x_i) p_k(x_i) \quad (9)$$

Taking inspiration from the biological cerebellar micro-structure, the final output of the entire cerebellar circuit is the neural command coming from the Deep Cerebellar Nucleus (DCN) or Deep Nuclear Cell which represents the inverse model corrective torque u_{im} (Equation 8).

At each simulation iteration, the total effort command u_t to be sent to the robot is computed as in the Equation (8).

$$u_t(x_i) = u_{fb}(x_i) + u_{im}(x_i) = u_{fb}(x_i) + u_{lwr}(x_i) + u_{pl}(x_i) \quad (10)$$

2.3. Evolutionary Algorithm

In evolutionary robotics, the desired robotic behaviors emerge automatically through evolution due to the optimization and interactions between the robot and its surrounding environment. As a specification for the evolutionary procedure, a fitness function, which measures the ability of a robotic individual to perform the desired task, is defined based on this optimization procedure, the algorithm identifies the optimal robotic configuration (Pratihari, 2003).

In this research, an evolutionary algorithm to optimize the initial parameters of the CPG is applied using a *covariance matrix adaptation evolutionary strategy* (CMA-ES) (Hansen, 2006). It is a stochastic optimization algorithm which, compared to other evolutionary procedures, has the advantage of converging rapidly in a landscape with several local minima and requires few initialization parameters (Hansen, 2006). In an iterative fashion, the algorithm changes the initial CPG parameters (Table 1) and simulates the resulting locomotion patterns on the simulated

robotic platform for 2 min. At the end of the simulation, a fitness function computes a score to give to the different individuals, based on the distance each robot has covered during the locomotion simulation. The initial parameters for the CMA-ES are implemented as described by Hansen (2006).

2.4. Experimental Design

To assess the advantages of exploiting adaptability in employing evolution strategies for robotic locomotion tasks, two different configurations of the system are evolved (Figure 4):

- **adapt-after-evo:** Co-evolution of the CPG parameters and PID gains (Tables 1, 2)
 - *genotype*: CPG parameters + PID gains
 - *phenotype*: locomotion patterns
- **adapt-in-evo:** Evolution of the CPG parameters + learning phase of the cerebellar circuit (fixed PID gains, Tables 1, 2)
 - *genotype*: CPG parameters
 - *phenotype*: locomotion patterns + RFs in the cerebellar circuit

The PID gains are part of the evolved parameters in the *adapt-after-evo* in order to have a fair comparative study of the performance of the two systems. The classic controller (the *adapt-after-evo*) should be also optimized by the evolutionary exploration. Their initial conditions and the boundaries for the CPG parameters are the same, as in Table 1.

As a starting point for the evolution, the PID gains are the same for both robotic configurations: *adapt-after-evo* and *adapt-in-evo*. In the *adapt-after-evo* configuration, the PID gains are part of the evolutionary process and their boundaries are defined according to empirical evaluations on the stability of the system, while in the *adapt-in-evo* system configuration when

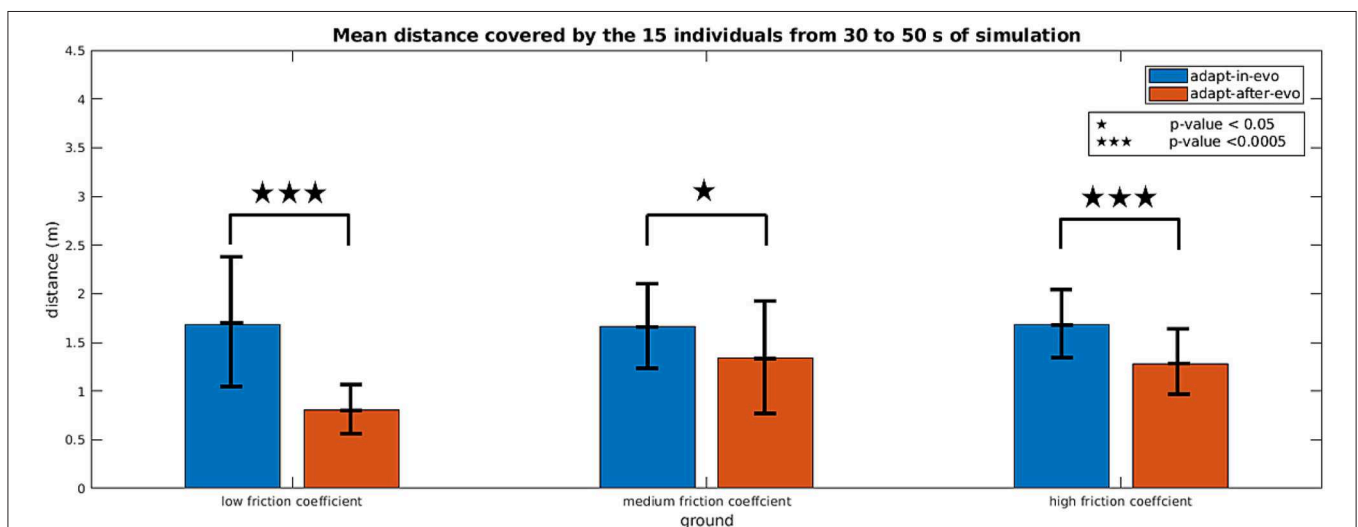


FIGURE 6 | Histograms which summarize the mean and standard deviation of the distance covered by the 15 individuals with the two control strategies *adapt-in-evo* and *adapt-after-evo* in the three different levels of robot-ground friction. The p -values, regarding the statistical significance of the performance of the two system *adapt-in-evo* and *adapt-after-evo*, are also shown in the figure.

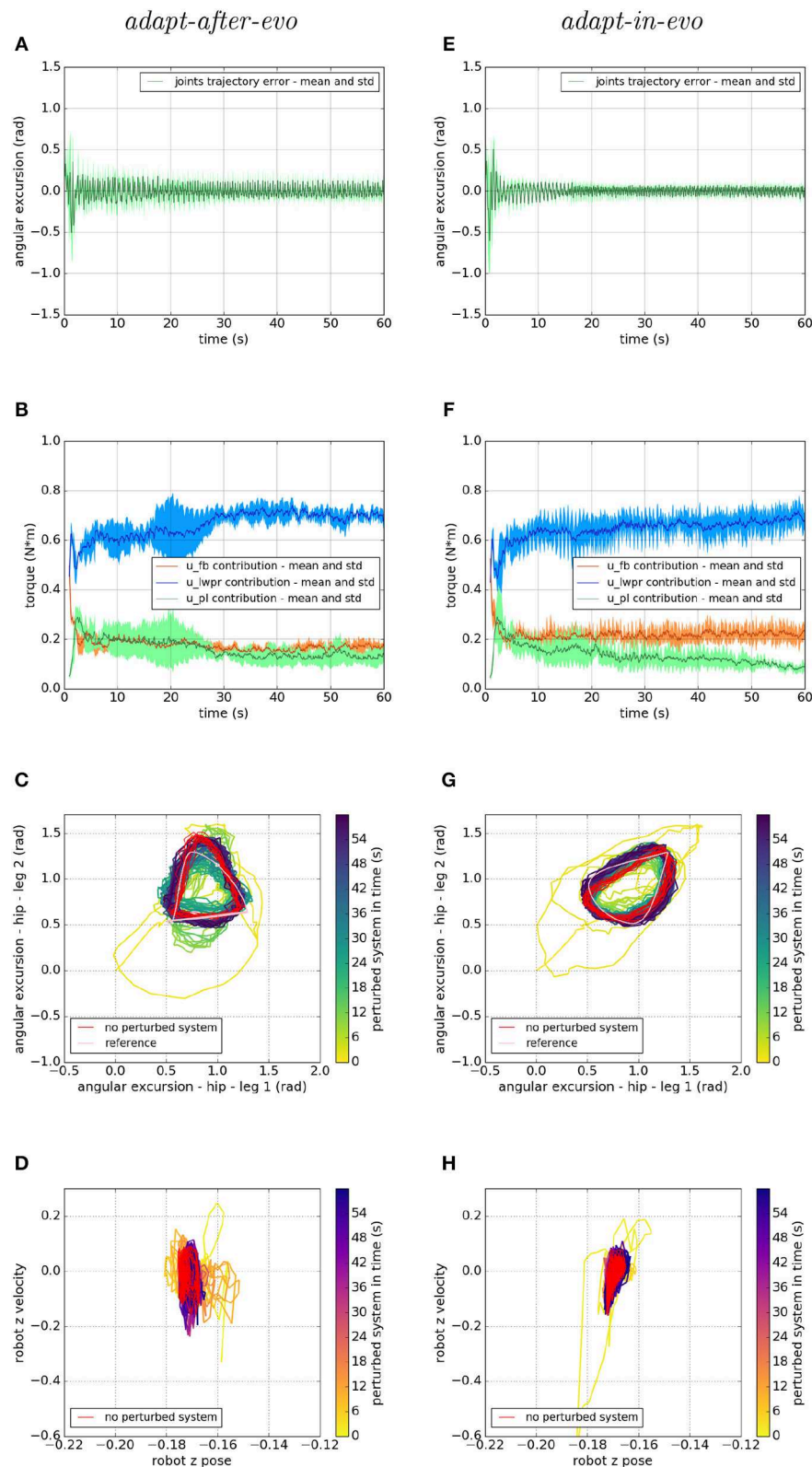


FIGURE 7 | Locomotion performance and characterization of the two systems; on the left the *adapt-after-evo* system and, on the right, the *adapt-in-evo* one, with a friction coefficient of 0.95 between robot and terrain. **(A,E)** Represent the mean and the standard deviation of the position error of the four legs joints and **(B,F)** instead *(Continued)*

FIGURE 7 | describe the mean and the standard deviation of the contribution ratio of the different modules of the control architecture. **(C,G)** Describe the periodic behavior relation between the actual joint trajectories of leg 1 and leg 2 compared to their reference values, in pink, and to the behavior of the no perturbed system, in red (among the other pairs of legs, the relation is periodic in a comparable way). Eventually, **(D,H)** represent the dynamics of the CoM of the robot, on the vertical axis to the ground, compared the same CoM dynamics when the system is not perturbed (in red).

the cerebellar circuit is plugged in the system, they are fixed (**Figure 4, Table 2**).

Concerning the specification of the cerebellar circuit, an experimental tuning has been performed on four of the most significant hyper-parameters of the LWPR algorithm (Vijayakumar and Schaal, 2000) (*init_D*, *init_α*, *w_gen* and *add_threshold* in **Table 2**), to obtain a stable and corrective system behavior for the frequency range of the locomotion trajectories (ω in **Table 1**), used as starting point of the evolutionary algorithm. This is an important constraint for the experiments because the response of the system needs to be stable for all the possible solutions found by the evolutionary algorithm. Ensuring stability in the system allows inspecting an unbiased comparison even if the adaptive part of the controller is included afterwards.

The first two hyper-parameters considered (*init_D* and *init_α*) are related to the creation of new Receptive Fields, while the last two (*w_gen* and *add_threshold*) directly influence the local regression algorithm. All the hyper-parameters are the same for the 4 Unit Learning Machines and they are described as follows:

- **init_D** = 0.7, it represents the initial distance metric which is assigned to each new created Receptive Fields (RFs);
- **w_gen** = 0.6, it is critical for the creation of new RFs. If no local model shows an activation greater than this value, a new RF is generated;
- **init_α** = 500, it is the initialization value for the learning rate in the gradient descent algorithm which minimizes the error in the different regressions of the input space;
- **add_threshold** = 0.95, it operates as a threshold value to stand when a new regression direction should be added to the algorithm. If the ratio between the mean squared error of the current regression dimension and the same mean squared error, at the previous time iteration, is lower than this value, thus, a new regression direction can be exploited in the robot modeling process.

All the simulations were run on the Neurorobotics Platform and implemented through its utilities, which has been shown capable of implementing robotic control loops (Vannucci et al., 2015). The controller was implemented using a domain-specific language that eases the development of robotic controllers, and that is part of the Neurorobotics Platform simulation engine (Hinkel et al., 2017). Another tool, called *Virtual Coach* and also included in the platform and employed to implement the evolutionary algorithm. It was used because capable of launching batch simulations with different parameters and gathering and storing results from these.

3. EXPERIMENTAL RESULTS

In both evolutionary configurations, each of the 16 generations consists of 10 individuals. Every simulation lasted for 2 min, which is enough time for the LWPR to converge. After the simulation, the fitness function has been computed.

In **Table 1**, the resultant characteristic parameters of the final CPG configurations for the best individuals in the *adapt-after-evo* and *adapt-in-evo* configuration, are shown.

In **Table 2**, for the *adapt-after-evo*, the PID gains are part of the genotype and their initial conditions represent the same fixed controller parameters used for the *adapt-in-evo*. Thus, in the *adapt-after-evo* case, the PID gains are changed by the evolutionary process, within the experimentally found boundary conditions for the starting locomotion robotic patterns to be stable and tolerable. Differently, the *adapt-in-evo* profits from the contribution of the cerebellar-inspired controller (**Figure 3B**), whose hyper-parameters (*init_D*, *init_α*, *w_gen* and *add_threshold*) are set as shown in **Table 2** and explained in section 2.4.

After the evolutionary process, experiments that compare the behavior of the two systems have been performed. To perform this comparison, the same cerebellar circuit, that was used in the *adapt-in-evo*, was plug in the *adapt-after-evo*. Thus, both systems are now adaptive thank to the contribution of the cerebellar control module and it is possible to test and compare the benefits of control adaptability *during* or *after* the optimization of the planning of the locomotion trajectories. The two resultant control architectures are then representative for:

- control adaptability during the evolutionary optimization of the CPG locomotion patterns (*adapt-in-evo*)
- control adaptability after the evolutionary optimization of the CPG locomotion patterns (*adapt-after-evo*)

While the individual representative for the *adapt-in-evo* architecture can safely be chosen as the winner of the evolutionary algorithm, the effect of adding the adaptive component to create the *adapt-after-evo* cannot be easily predicted. Thus, in order to better choose the individual for the *adapt-after-evo* architecture, the cerebellar circuit was added to the best three individuals resulting from the evolutionary process. After evaluating again, the fitness with the adaptive component, the one individual with better performances was chosen as the representative one.

In general, to provide a fair comparison between the two systems, the distance is computed only after the cerebellar algorithm has converged, as in the initial phase, where learning occurs, we can observe some instability. After this initial phase, that lasts for around 20 s, we can notice no significant improvements in the position error on the joint

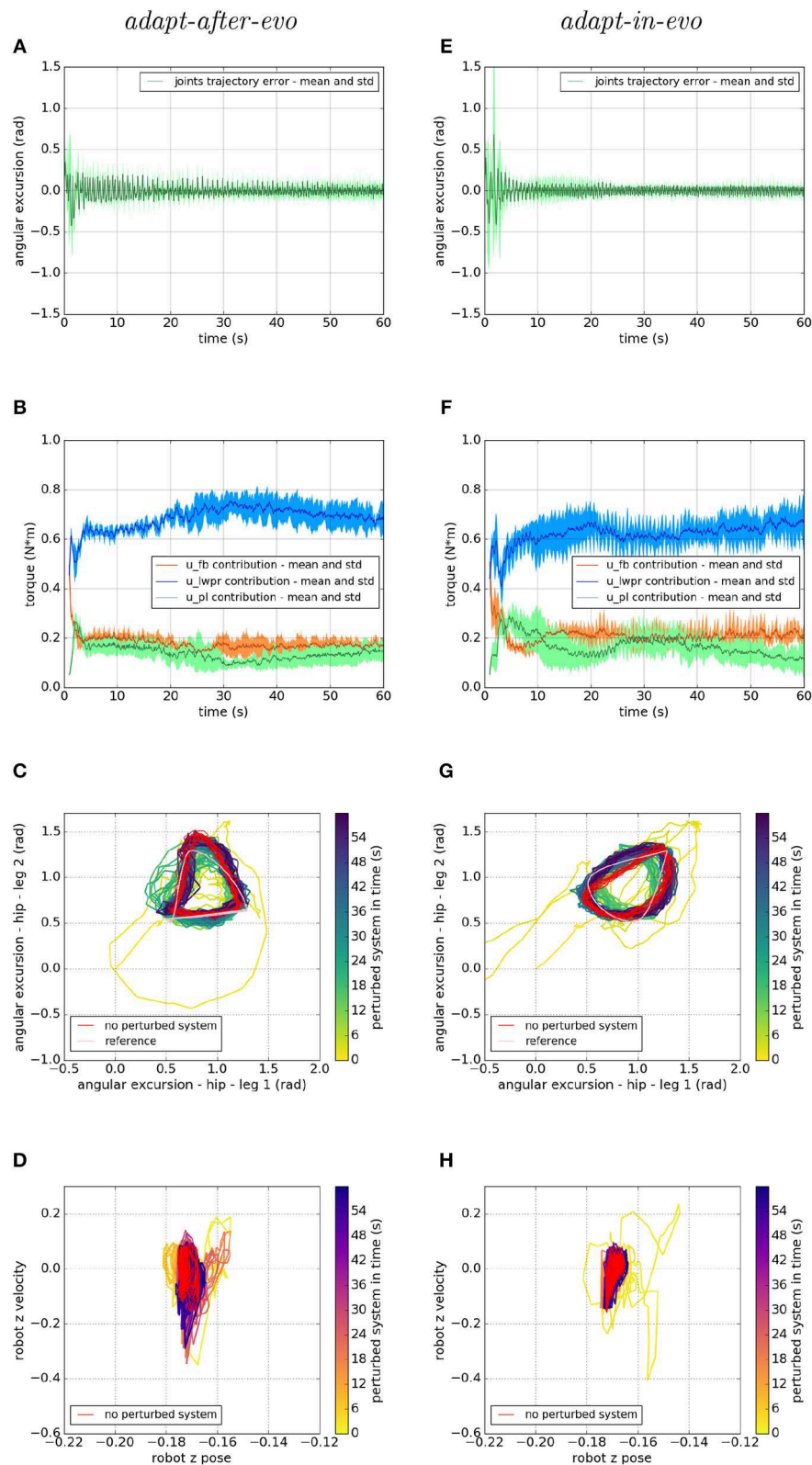


FIGURE 8 | Locomotion performance and characterization of the two systems; on the left the *adapt-after-evo* system and, on the right, the *adapt-in-evo* one, with a friction coefficient of 0.5 between robot and terrain. **(A,E)** Represent the mean and the standard deviation of the position error of the four legs joints and **(B,F)** instead *(Continued)*

FIGURE 8 | describe the mean and the standard deviation of the contribution ratio of the different modules of the control architecture. **(C,G)** Describe the periodic behavior relation between the actual joint trajectories of leg 1 and leg 2 compared to their reference values, in pink, and to the behavior of the no perturbed system, in red (among the other pairs of legs, the relation is periodic in a comparable way). Eventually, **(D,H)** represent the dynamics of the CoM of the robot, on the vertical axis to the ground, compared the same CoM dynamics when the system is not perturbed (in red).

trajectories, which could indicate that most of the learning has been done. This can also be observed by looking at the number of receptive fields created by the LWPR algorithm, that is not increasing anymore. Therefore, to avoid having the learning phase affecting the computation of the distance covered by the robot, a time window of 20 s is considered, from 30 to 50 s, during which the distance covered by the robot is recorded and compared between the two different cases (*adapt-after-evo* and *adapt-in-evo*).

3.1. Base Comparison

After simulating the best *adapt-after-evo* and *adapt-in-evo* individuals 10 times for 1 min, the results show that the winner robot walks for 1.72 m on average with the *adapt-in-evo* controller while it walks for 1.48 m with the *adapt-after-evo*. The respective standard deviations are 0.2 m for the *adapt-in-evo* controller and 0.11 m for the *adapt-after-evo*. This shows that, in the task space, there are benefits in using the adaptive controller during the search for the best locomotion patterns, rather than connecting it to the control architecture afterwards. The superiority of the *adapt-in-evo* approach is raised also by the fact that PID gains are not evolved and they keep the values, presented in **Table 2**, while the same gains are optimized in the *adapt-after-evo* approach.

Regarding the behavior of the two systems in the joint space, we analyze the differences in their performances as shown in **Figure 5**. On the left column, the *adapt-after-evo*-related plots are shown and on the right column, the plots related to the *adapt-in-evo*-system are presented.

Figures 5A,E represent the mean and the standard deviation of the position error of all the robotic legs. In both pictures, after an overshoot at the beginning of the simulation, which represents the transient where the cerebellar controller is calibrating its corrective contribution, the error decreases along with the simulation. Comparing the two plots, it is appreciable that in the *adapt-in-evo* trial (e) the error in following the reference positions is almost half compared to the other case *adapt-after-evo* (a). Their Root Mean Square Error (RMSE) are, respectively, 0.035 radians and 0.056 radians.

Then, in **Figures 5B,F**, the mean and the standard deviation of the ratio of the contributions of the different parts of the bio-inspired cerebellar controller are highlighted. It is evident that, in both cases, the contribution of the LWPR, whose teaching signal is the global motor command to the robot u_t , becomes predominant compared to the feedback controller contribution (PID). Furthermore, the PL contribution, whose teaching signal is the feedback controller u_{fb} , follows the trend of the output of the PID controller, which decreases along with the simulation, meaning that the final motor commands to the robot are mostly relying on the u_{im} output.

On the third line, **Figures 5C,G** stress the periodic and stable locomotion which characterizes the system after the first seconds of simulation. In the **Figures 5C,F**, just the cyclic behavior of two robotic legs (leg 1, one of the front legs, and leg 2, one of the hind legs) has been reported. The remaining two legs present comparable performances. It is appreciable from **Figures 5C,G** that the relation among the angular excursions of the two legs becomes more periodic along with the simulation time and closer to the pink limit cycle, shown to mark the reference trajectories of leg 1 and leg 2.

Ultimately, at the level of the task space, a dynamic analysis of the robotic locomotion is exhibited in **Figures 5D,H** when the robot vertical position is plotted against its vertical speed. In these images (**Figures 5D,H**), the dynamics of the system become more defined and constrained over time. It is relevant to point out that, in the *adapt-in-evo* case (**Figure 5H**) the winner locomotion patterns grant more robust locomotion, which is represented by a more confined stability region in the phase space with respect to the *adapt-after-evo* system (**Figure 5D**).

3.2. Statistical Analysis on Different Experimental Conditions

After discussing the results concerning the advantages of using control adaptability during the optimization of the locomotion trajectories (*adapt-in-evo*) rather than employing it afterwards (*adapt-after-evo*), we investigated on the effects of altering the experimental conditions with respect to the simulation circumstances where the locomotion patterns have been found. These experiments are also useful for testing the system in more realistic scenarios, which goes toward overcoming the reality gap. The adaptation to the changes in the experimental scenario is possible since the weights of the LWPR are never locked to certain values, but they are always updating based on the experimental circumstances.

The changes in the experimental constraints have been applied in the following order:

- variability in the robotic dynamics;
- variability in the interaction with the environment.

First, to verify the abstraction potential of the previous results, a population of 15 slightly different Fable robots is generated. After checking the consistency of the simulation in a certain range of variation of the robotic model dynamic parameters, we decided to generate 15 robots with the following features:

- additive white Gaussian noise (AWGN) fed in the encoder of the motors and randomly selected from a uniform distribution in the range of [0–10] % of the motor signal;
- damping coefficient, randomly taken from a uniform distribution in the range of [0.08–0.25] $\frac{Ns}{m}$, to define the dynamic model of all the hip joints of the robot.

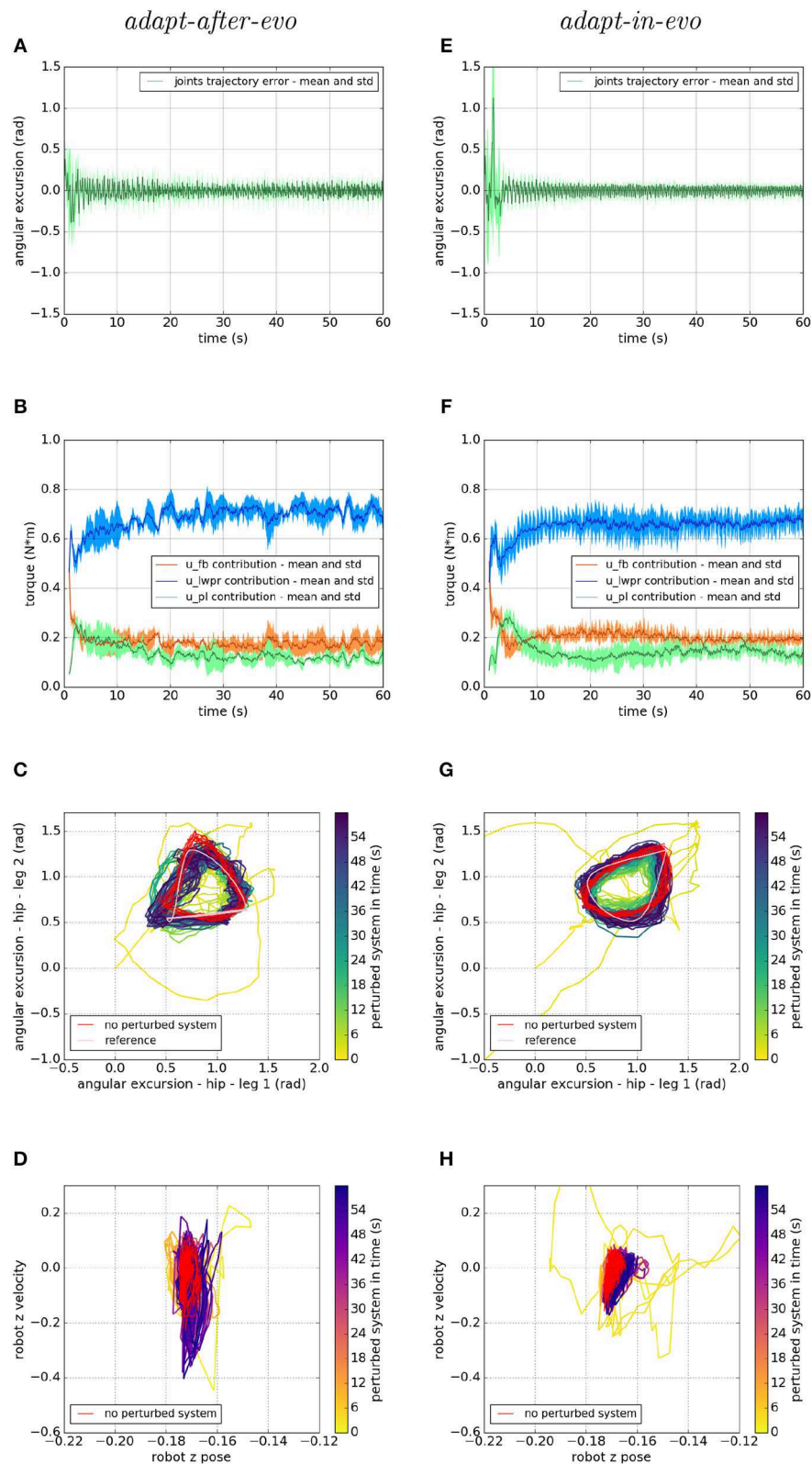


FIGURE 9 | Locomotion performance and characterization of the two systems; on the left the *adapt-after-evo* system and, on the right, the *adapt-in-evo* one, with a friction coefficient of 0.3 between robot and terrain. **(A,E)** Represent the mean and the standard deviation of the position error of the four legs joints and **(B,F)** instead *(Continued)*

FIGURE 9 | describe the mean and the standard deviation of the contribution ratio of the different modules of the control architecture. **(C,G)** Describe the periodic behavior relation between the actual joint trajectories of leg 1 and leg 2 compared to their reference values, in pink, and to the behavior of the no perturbed system, in red (among the other pairs of legs, the relation is periodic in a comparable way). Eventually, **(D,H)** represent the dynamics of the CoM of the robot, on the vertical axis to the ground, compared the same CoM dynamics when the system is not perturbed (in red).

Thus, the resulting 15 Fable robots have different dynamic characteristics and noisy signals injected in their motors' encoder. These modifications model the variability in the robotic population.

Subsequently, other experimental constraints have been modified. They represent the variability in the interaction robot-environment. Thus, to modulate this aspect of the simulation, the static friction coefficient is altered in the x – direction of the world reference frame. The default value of the simulator for this parameter is 1, meaning maximum static friction between robot and ground and we decided to affect the experiments by giving three different levels: 0.3, 0.5, 0.95 of static friction coefficient to the interaction robot-ground. Lower coefficients imply greater disturbances to the system. To have consistent results, the previously generated robotic individuals are simulated ten times for 1 min in each of the 3 different friction conditions explained above.

Figure 6 shows histograms with an error bar for the mean and standard deviation of the distance covered by all the combinations robot-terrain, simulated with the two different control architectures *adapt-after-evo* and *adapt-in-evo*, 10 times per individual.

A two-way repeated measures ANOVA (Potvin and Schutz, 2000) was run to determine the effect of the two systems (*adapt-in-evo*, and *adapt-after-evo*), i.e., factor *controller* over three different ground-robot interactions (low, medium and high friction), i.e., factor *ground* on the explanatory variable walked distance (D), expressed in meters. Data are mean \pm standard deviation. Analysis of the studentized residuals showed that there was normality, as assessed by the Shapiro-Wilk test of normality (Razali and Wah, 2011) and no outliers, as assessed by no studentized residuals greater than ± 3 standard deviations. The assumption of sphericity was violated for the interaction term, as assessed by Mauchly's test of sphericity ($X^2(2) = 7.003$, $p = 0.03$) (Gleser, 1966). There was a statistically significant interaction between *controller* and *ground* on D , $F_{(1.412, 19.767)} = 4.288$, $p = 0.04$, $\epsilon = 0.706$ (Greenhouse-Geisser correction Abdi, 2010), partial $\eta^2 = 0.234$.

Simple main effects were run for the factor *controller* (**Figure 6**). D of *adapt-in-evo* controller was always higher than that of *adapt-after-evo*:

- data for low-friction *ground* were (1.69 ± 0.71) m and (0.80 ± 0.27) m, respectively, p -value < 0.0005 (3 stars in **Figure 6**);
- data for medium-friction *ground*, (1.66 ± 0.45) m and (1.34 ± 0.55) m, respectively, p -value < 0.05 (1 star in **Figure 6**);
- data for high-friction *ground*, (1.68 ± 0.38) m and (1.28 ± 0.31) m, respectively, p -value < 0.0005 (3 stars in **Figure 6**);

Figures 7–9 describe the behavior of the two systems *adapt-after-evo* (on the left column) and *adapt-in-evo* (on the right one) in

the three different friction conditions with the terrain (**Figure 7** is high friction, **Figure 8** is medium friction and **Figure 9** is low friction). To analyze data from a representative experiment, the plots (**Figures 7–9**) include the behavior of one of the ten reiterations of the robotic individual whose performance, in covered distance D , is the closest to the average behavior among all the individuals in the two control cases *adapt-after-evo* and *adapt-in-evo*, for all the 3 levels of friction. This selected agent has a noise injected in the encoder which is 2% of its total motor signal, while its joints damping coefficient is $0.19 \frac{Ns}{m}$.

In all three cases (**Figures 7–9**), subplots (a) (*adapt-after-evo*) and (e) (*adapt-in-evo*) highlight that during the first minute of simulation, the position errors at the joint level are decreasing, even if the experimental conditions (robotic model and robot-ground friction coefficient) are changed compared to the initial simulation constraints, where the locomotion patterns have been found. The error for the system *adapt-in-evo* (right column) is always smaller than for the other system *adapt-after-evo* (left column), observing both its mean and standard deviation across the four legs. In the three different robot-ground interactions (**Figures 7–9**), the Root Mean Square Error (RMSE) in the following of the desired joint trajectories is shown in **Table 3**.

The contributions of the different modules of the controller architecture (subplots b and f) show the same trend as in **Figure 5**; after a few seconds after the beginning of the simulation, *u-lwpr* becomes predominant and *u-pl* learns the *u-fb* and they together decrease their contributions along the simulation.

The most significant differences between the behavior of two compared systems *adapt-after-evo* and *adapt-in-evo* without disturbances (**Figure 5**) and that when the dynamics of the experiments have been changed (**Figures 7–9**), can be observed in subplots (c, d, g, h). At joints level (**Figures 7C,G, 8C,G, 9C,G**), the performances of the two systems *adapt-after-evo* and *adapt-in-evo* demonstrate a less stable behavior if compared to the same subplots (c) and (g) in **Figure 5**. The trend of the joints trajectories still converges to the limit cycle obtained by the position references, which is indicated in pink, and to the periodic shape got in the last 10 s of simulation for the same system without disturbances. However, lower the friction coefficient value, longer the time the systems take to converge to the desired periodic behavior (**Figures 7–9**). It is also relevant to point out that the entropy of the joint trajectories increases in inverse proportion to the static friction coefficient of friction with the ground (the minimum tested static friction coefficient is showed in **Figure 9**).

Eventually, a meaningful index of the difference in the stability response of the two systems *adapt-after-evo* and *adapt-in-evo* is the plot showing the dynamics of the Center of

TABLE 3 | Root Mean Square Error in following the desired locomotion trajectories for the representative Fable robot individuals.

Friction coefficient	Adapt-after-evo	Adapt-in-evo
High	0.056 rad	0.035 rad
Medium	0.039 rad	0.033 rad
Low	0.044 rad	0.041 rad

These values are related to **Figures 7A,E, 8A,E, 9A,E**.

Mass (CoM) of the robot (d, h). Here, the stability region in the no disturbances case is represented in red, while the behavior for the affected systems is in the remaining color gradient timeline (**Figures 7, 8, 9D,H**). In all the three figures (**Figures 7, 8**), the behavior of the *adapt-in-evo* agent (on the right) is confined in a region of the phase space which is very close to region covered by the dynamics of the same system without disturbances (in red in subplots d and h). Instead, the dynamics of the center of mass of the *adapt-after-evo* experiments (on the left column in **Figures 7–9**) are always more unstable than its equivalent *adapt-in-evo* (**Figures 7–9**), meaning that the adaptability, brought by the cerebellar inspired module, as a control feature during the evolutionary exploration for effective locomotion trajectories, contributes to discovery more flexible robotic locomotion patterns.

3.3. Dynamically Changing Experimental Set-Up

After testing the control architecture with a set of simulated Fable robots with different dynamical characteristics and friction interactions with the environment, further experiments are performed. This set of tests has been carried out to compare the performances of the two systems with respect to scenarios in which the interaction with the environment changes dynamically. In this case, the static friction coefficient is changed during the experiment, respectively, at 50 and 100 s from the beginning of the simulation and the simulation lasts 2 min in total.

For these experiments, the same representative individual we choose for designing the previous plots (2% of the motor signal as noise in the encoders and $0.19 \frac{Ns}{m}$ of joints damping coefficient) is tested for the dynamically changing set-up, and the simulations are run 5 times per type of controller (*adapt-after-evo* and *adapt-in-evo*).

Concerning the task space, the average, among the 5 trials, of the distance covered by the robot, from 50 to 120 s of simulation, is 6.18 m for the *adapt-after-evo* and 10.28 m for the *adapt-in-evo*, respectively, with standard deviation 2.25 and 2.40 m.

In **Figure 10**, we show the response of the two systems *adapt-after-evo*, on the left, and *adapt-in-evo*, on the right, when the friction coefficient is dynamically changed during the simulation. As explained in section 3.2, the initial static friction coefficient is 1, the maximum value allowed in the Gazebo simulator and then it is decreased to 0, its minimum, around 50 s from the beginning of the simulation, and increase

again to 0.5 at 100 s. In **Figure 10** the same graphs, as for the previous experiments, are shown. In subplots (a) and (e) the mean and standard deviation of the legs are shown. A fast spike is visible around 50 s of simulation when the interaction with the environment is changed, but then the position error decreases again and a slight change in the graph is also visible around 100 s when the friction is changed again. Both systems *adapt-after-evo* and *adapt-in-evo* reject the disturbance given by changing the static friction coefficient. Also, in this case, the assessment of the advantage brought by the *adapt-in-evo* controller is quantitatively proved by the RMSE which is 0.05 radian in the *adapt-after-evo* and 0.04 radian in the *adapt-in-evo* one.

In **Figures 10B,F**, it is clear that around 50 s of the simulation, an unexpected change perturbs the system and the *u-lwpr* and *u-pl* need to learn again the model of the interaction among robot and ground. The second change in the static friction coefficient is lightly visible around 100 s from the beginning of the simulation.

In conclusion, in the **Figures 10C,D,G,H**, the difference in the rejection of the disturbances among the two systems *adapt-after-evo* and *adapt-in-evo*, is more evident. In fact, after the second 50 of simulation, the *adapt-after-evo* is not able to completely recover from the disturbance. In fact, the last seconds of simulation (in dark blue) are slightly different from the behavior of the no-perturbed system (in red). This happens both at joint level in **Figure 10C** and at the task level in **Figure 10D**. On the contrary, the *adapt-in-evo* system feels the change in the interaction with the environment, but it can return to a state of the system which is closer to the initial one whose response is highlighted in red. The temporary divergence of the behavior of the system is visible around second 50 either in **Figure 10G**, in light green, and in **Figure 10H**, in pink. In these final subplots (c, e, g, h), the second change in the static friction coefficient does not have an evident impact, either in the *adapt-after-evo* and in the *adapt-in-evo* case. A significant divergence in the locomotion stability of the system is visible just in the dynamics of the CoM of the *adapt-after-evo* system in **Figure 10D**.

4. DISCUSSION

For the first time, taking inspiration from nature, the proposed research uses robotics to suggest the advantages and benefits of employing adaptive controllers in conjunction with optimization strategies, such as evolutionary algorithms. For this purpose, a new bio-inspired approach to control robotic locomotion is presented. The control design is based on neurophysiological evidences concerning a simplified model of the neural control in the locomotion of quadruped animals. In the proposed control architecture, the trajectory planner is a CPG-inspired system of equations and the motion controller is composed of a PID and a bio-inspired algorithm, whose weights are changing on-line with the simulation time. This latter part of the architecture models the adaptive role of the Cerebellar-inspired circuit in the locomotion of vertebrates which encodes information about the inverse dynamic model of the quadruped.

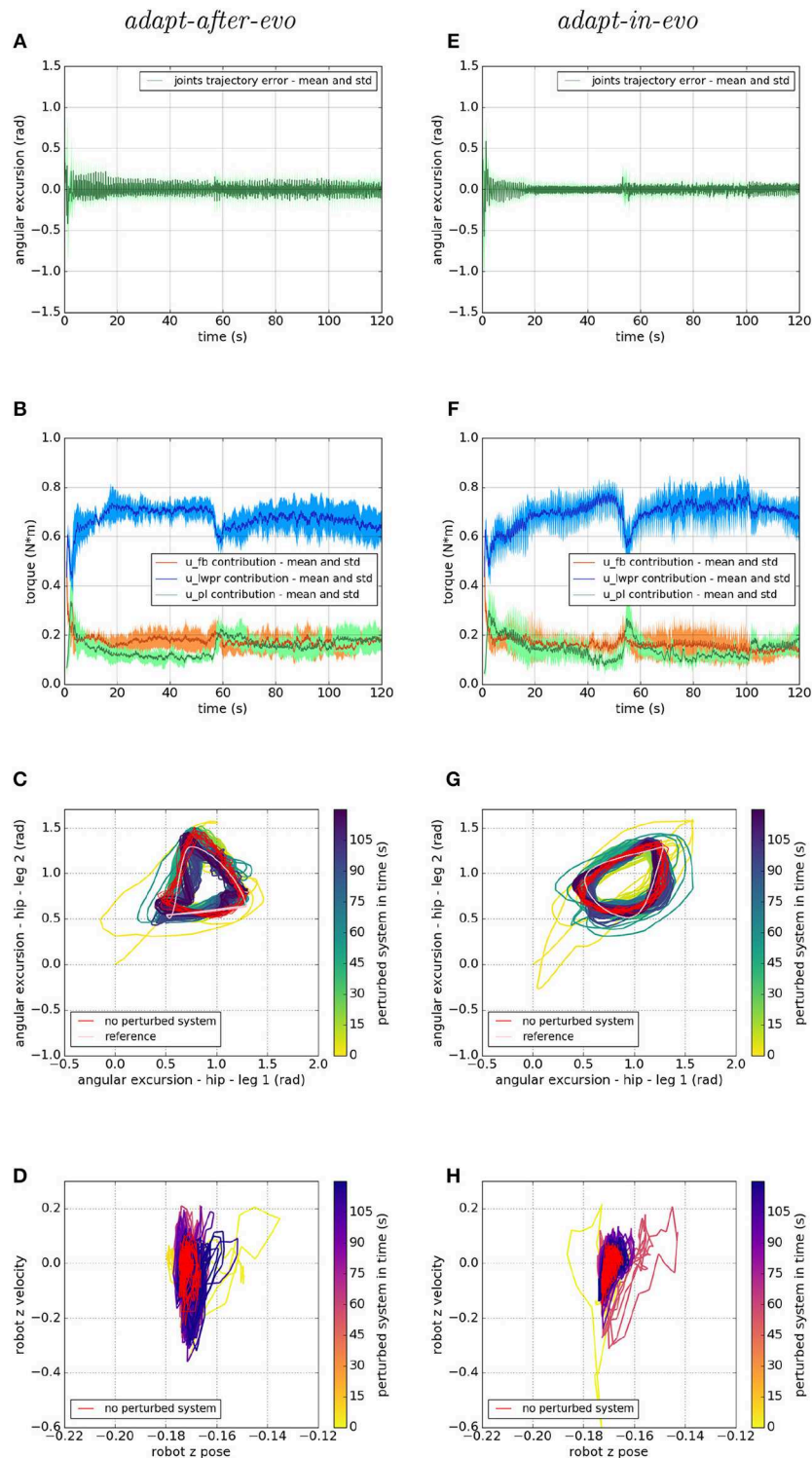


FIGURE 10 | Locomotion performance and characterization of the two systems; on the left the *adapt-after-evo* system and, on the right, the *adapt-in-evo* one, with a friction coefficient that has been changed from 1 to 0 at 50 s and from 0 to 0.5 at 100 s. **(A,E)** Represent the mean and the standard deviation of the position error of the four legs joints and **(B,F)** instead describe the mean and the standard deviation of the contribution ratio of the different modules of the control architecture. **(C,G)** Describe the periodic behavior relation between the actual joint trajectories of leg 1 and leg 2 compared to their reference values, in pink, and to the behavior of the no perturbed system, in red (among the other pairs of legs, the relation is periodic in a comparable way). Eventually, **(D,H)** represent the dynamics of the CoM of the robot, on the vertical axis to the ground, compared the same CoM dynamics when the system is not perturbed (in red).

The main contribution of the paper is investigating the advantages of using a learning control module during the optimization of the locomotion patterns for a quadruped robot rather than employ it when the optimal locomotion patterns have already been found (as it is usually done in already existing approaches, Urbain et al., 2018; Vandesompele et al., 2019). This idea comes from nature since evolution has always been acting on plastic and learning systems. The research aims to investigate if the solutions found out by the evolution-inspired algorithm are statistically better when a learning module is included in the controller, during the evolution. The presented approach shows the advantages of this optimization procedure for quadruped robotic locomotion both in the task and in the joint space. The distance covered by the robot is greater when the learning module is involved in the genetic optimization process and, the position error of the joints is smaller.

These results are also reflected in new experiments when the robot dynamic characteristics are changed, and some noise is injected in the robot encoders. The preponderance of the *adapt-in-evo* solution has been generalized by running other experiments with a different robot-environment interaction, which allows to infer the crossing of the reality gap. Further, the robot-ground interaction has also been dynamically changed during the experiments, assessing the potential of the *adapt-in-evo* approach in readjusting to different experimental constraints even though learning stability has already been reached by the cerebellar inspired module. The results show that the inclusion of the cerebellar-inspired control in the process of optimization of the locomotion trajectories allow a maximization of the synergy between the CPG-inspired trajectory planner and the adaptive cerebellar controller. The best patterns, which emerge during the previously explained synergy, are more robust. Even when the experimental conditions change, in the dynamics of the robot and in its interaction with the environment, before or during

the experiments, the locomotion preserves more stability both at joint and task level.

In conclusion, further investigations can be done by testing the architecture on the real Fable robot since the conducted experiments aimed at proving the suitability of employing the same controller in real scenarios. In fact, the results show that both control strategies, *adapt-after-evo* and *adapt-in-evo*, are robust enough to work, without changing parameters, in unexpected conditions such as noisy sensors or slippery terrains (also applied in the same experiment).

DATA AVAILABILITY

The datasets generated for this study are available on request to the corresponding author.

AUTHOR CONTRIBUTIONS

The bio-inspired control architecture was primarily developed by EM and ST. The use of the evolution-based approach was mainly handled by EM, GU, AV, and JD. EM, LV, UA, and MC worked on the implementation of the experiment. EM, AS, and EF statistically analyzed and interpreted the data. EM, LV, ST, EF, and CL wrote and reviewed the manuscript. All authors read and approved the final manuscript.

FUNDING

This project/research has received funding from the European Union's Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreement No. 785907 (Human Brain Project SGA2) and from the Marie Skłodowska-Curie Project No. 705100 (Biomodular).

REFERENCES

- Abdi, H. (2010). The greenhouse-geisser correction. *Encyclop. Res. Design* 1, 544–548.
- Corucci, F., Cheney, N., Lipson, H., Laschi, C., and Bongard, J. (2016). "Evolving swimming soft-bodied creatures," in *ALIFE XV, The Fifteenth International Conference on the Synthesis and Simulation of Living Systems, Late Breaking Proceedings* (Cambridge, MA), 6.
- Dean, P., Porrill, J., Ekerot, C.-F., and Jörntell, H. (2010). The cerebellar microcircuit as an adaptive filter: experimental and computational evidence. *Nat. Rev. Neurosci.* 11:30. doi: 10.1038/nrn2756
- Falotico, E., Vannucci, L., Ambrosano, A., Albanese, U., Ulbrich, S., Vasquez Tieck, J. C., et al. (2017). Connecting artificial brains to robots in a comprehensive simulation framework: The neurorobotics platform. *Front. Neurobotics* 11:2. doi: 10.3389/fnbot.2017.00002
- Floreano, D., Dürr, P., and Mattiussi, C. (2008). Neuroevolution: from architectures to learning. *Evol. Intell.* 1, 47–62. doi: 10.1007/s12065-007-0002-4
- Fujiki, S., Aoi, S., Funato, T., Tomita, N., Senda, K., and Tsuchiya, K. (2015). Adaptation mechanism of interlimb coordination in human split-belt treadmill walking through learning of foot contact timing: a robotics study. *J. R. Soc. Interface* 12:20150542. doi: 10.1098/rsif.2015.0542
- Full, R., and Koditschek, D. (1999). Templates and anchors: neuromechanical hypotheses of legged locomotion on land. *J. Exp. Biol.* 202, 3325–3332.
- Garrido Alcazar, J. A., Luque, N. R., D'Angelo, E., and Ros, E. (2013). Distributed cerebellar plasticity implements adaptable gain control in a manipulation task: a closed-loop robotic simulation. *Front. Neural Circuits* 7:159. doi: 10.3389/fncir.2013.00159
- Gay, S., Santos-Victor, J., and Ijspeert, A. (2013). "Learning robot gait stability using neural networks as sensory feedback function for central pattern generators," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, number EPFL-CONF-187784 (Piscataway, NJ), 194–201. doi: 10.1109/IROS.2013.6696353
- Gleser, L. J. (1966). A note on the sphericity test. *Ann. Math. Stat.* 37, 464–467. doi: 10.1214/aoms/1177699529
- Hansen, N. (2006). "The cma evolution strategy: a comparing review," in *Towards a New Evolutionary Computation*, eds J. A. Lozano, P. Larrañaga, I. Inza, and E. Bengoetxea (Berlin: Springer), 75–102.
- Harvey, I., Paolo, E. D., Wood, R., Quinn, M., and Tuci, E. (2005). Evolutionary robotics: a new scientific tool for studying cognition. *Artif. Life* 11, 79–98. doi: 10.1162/1064546053278991
- Hinkel, G., Groenda, H., Krach, S., Vannucci, L., Denninger, O., Cauli, N., et al. (2017). A framework for coupled simulations of robots and spiking neuronal networks. *J. Intell. Robot. Syst.* 85, 71–91. doi: 10.1007/s10846-016-0412-6
- Hughes, G., and Wiersma, C. (1960). The co-ordination of swimmeret movements in the crayfish, *procambarus clarkii* (girard). *J. Exp. Biol.* 37, 657–670.

- Ijspeert, A. J. (2008). Central pattern generators for locomotion control in animals and robots: a review. *Neural Netw.* 21, 642–653. doi: 10.1016/j.neunet.2008.03.014
- Ijspeert, A. J., Crespi, A., Ryczko, D., and Cabelguen, J.-M. (2007). From swimming to walking with a salamander robot driven by a spinal cord model. *Science* 315, 1416–1420. doi: 10.1126/science.1138353
- Ito, M. (2000). Mechanisms of motor learning in the cerebellum. *Brain Res.* 886, 237–245. doi: 10.1016/S0006-8993(00)03142-5
- Ito, M. (2008). Control of mental activities by internal models in the cerebellum. *Nat. Rev. Neurosci.* 9:304. doi: 10.1038/nrn2332
- Kawato, M., and Gomi, H. (1992). A computational model of four regions of the cerebellum based on feedback-error learning. *Biol. Cybern.* 68, 95–103. doi: 10.1007/BF00201431
- Kousuke, I., Takaaki, S., and Shugen, M. (2007). “Cpg-based control of a simulated snake-like robot adaptable to changing ground friction,” in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Piscataway, NJ), 1957–1962.
- Lipson, H., and Pollack, J. B. (2000). Automatic design and manufacture of robotic lifeforms. *Nature* 406:974. doi: 10.1038/35023115
- Nolfi, S., Floreano, D., and Floreano, D. D. (2000). *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*. Cambridge, MA: MIT Press.
- Ojeda, I. B., Tolu, S., and Lund, H. H. (2017). “A scalable neuro-inspired robot controller integrating a machine learning algorithm and a spiking cerebellar-like network,” in *Conference on Biomimetic and Biohybrid Systems* (Berlin: Springer), 375–386.
- Pacheco, M., Fogh, R., Lund, H. H., and Christensen, D. J. (2014). “Fable: a modular robot for students, makers and researchers,” in *Proceedings of the IROS Workshop on Modular and Swarm Systems: From Nature to Robotics* (Piscataway, NJ).
- Potvin, P. J., and Schutz, R. W. (2000). Statistical power for the two-factor repeated measures anova. *Behav. Res. Methods Instrum. Comput.* 32, 347–356. doi: 10.3758/BF03207805
- Pratihari, D. K. (2003). Evolutionary robotics: a review. *Sadhana* 28, 999–1009. doi: 10.1007/BF02703810
- Razali, N. M., and Wah Y. B. (2011). Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. *J. Stat. Model. Anal.* 2, 21–33.
- Ryu, J.-K., Chong, N. Y., You, B. J., and Christensen, H. I. (2010). Locomotion of snake-like robots using adaptive neural oscillators. *Intell. Serv. Robot.* 3:1. doi: 10.1007/s11370-009-0049-4
- Starzyk, J. A. (2008). “Motivation in embodied intelligence,” in *Frontiers in Robotics, Automation and Control*, ed A. Zemliak (London, UK: InTech), 83–110.
- Takakusaki, K. (2013). Neurophysiology of gait: from the spinal cord to the frontal lobe. *Movem. Disord.* 28, 1483–1491. doi: 10.1002/mds.25669
- Tolu, S., Vanegas, M., Garrido, J. A., Luque, N. R., and Ros, E. (2013). Adaptive and predictive control of a simulated robot arm. *Int. J. Neural Syst.* 23:1350010. doi: 10.1142/S012906571350010X
- Tolu, S., Vanegas, M., Luque, N. R., Garrido, J. A., and Ros, E. (2012). Bio-inspired adaptive feedback error learning architecture for motor control. *Biol. Cybern.* 106, 507–522. doi: 10.1007/s00422-012-0515-5
- Urbain, G., Vandesompele, A., Wyffels, F., and Dambre, J. (2018). “Calibration method to improve transfer from simulation to quadruped robots,” in *International Conference on Simulation of Adaptive Behavior* (Berlin: Springer), 102–113.
- Vandesompele, A., Urbain, G., Mahmud, H., wyffels, F., and Dambre, J. (2019). Body randomization reduces the sim-to-real gap for compliant quadruped locomotion. *Front. Neurobotics* 13:9. doi: 10.3389/fnbot.2019.00009
- Vannucci, L., Ambrosano, A., Cauli, N., Albanese, U., Falotico, E., Ulbrich, S., et al. (2015). “A visual tracking model implemented on the icub robot as a use case for a novel neurobotic toolkit integrating brain and physics simulation,” in *IEEE-RAS International Conference on Humanoid Robots* (Piscataway, NJ), 1179–1184. doi: 10.1109/HUMANOID.2015.7363512
- Vannucci, L., Falotico, E., Tolu, S., Cacucciolo, V., Dario, P., Lund, H. H., et al. (2017). A comprehensive gaze stabilization controller based on cerebellar internal models. *Bioinspir. Biomimet.* 12:065001. doi: 10.1088/1748-3190/aa8581
- Vannucci, L., Tolu, S., Falotico, E., Dario, P., Lund, H. H., and Laschi, C. (2016). “Adaptive gaze stabilization through cerebellar internal models in a humanoid robot,” in *2016 6th IEEE International Conference on Biomedical Robotics and Biomechanics (BioRob)* (Piscataway, NJ: IEEE), 25–30. doi: 10.1109/BIOROB.2016.7523593
- Vijayakumar, S., and Schaal, S. (2000). “Locally weighted projection regression: an o(n) algorithm for incremental real time learning in high dimensional space,” in *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000)* (San Francisco, CA), Vol. 1, 288–293.
- Wolpert, D. M., Miall, R. C., and Kawato, M. (1998). Internal models in the cerebellum. *Trends Cogn. Sci.* 2, 338–347. doi: 10.1016/S1364-6613(98)01221-2

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Massi, Vannucci, Albanese, Capolei, Vandesompele, Urbain, Sabatini, Dambre, Laschi, Tolu and Falotico. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Generating Pointing Motions for a Humanoid Robot by Combining Motor Primitives

J. Camilo Vasquez Tieck^{1*}, Tristan Schnell¹, Jacques Kaiser¹, Felix Mauch¹, Arne Roennau¹ and Rüdiger Dillmann^{1,2}

¹ FZI Research Center for Information Technology, Karlsruhe, Germany, ² Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

OPEN ACCESS

Edited by:

Rainer Goebel,
Maastricht University, Netherlands

Reviewed by:

Fernando Perez-Pea,
University of Cádiz, Spain
Huaping Wang,
Beijing Institute of Technology, China

*Correspondence:

J. Camilo Vasquez Tieck
tieck@fzi.de

Received: 15 April 2019

Accepted: 02 September 2019

Published: 18 September 2019

Citation:

Tieck JCV, Schnell T, Kaiser J, Mauch F, Roennau A and Dillmann R (2019) Generating Pointing Motions for a Humanoid Robot by Combining Motor Primitives. *Front. Neurobot.* 13:77. doi: 10.3389/fnbot.2019.00077

The human motor system is robust, adaptive and very flexible. The underlying principles of human motion provide inspiration for robotics. Pointing at different targets is a common robotics task, where insights about human motion can be applied. Traditionally in robotics, when a motion is generated it has to be validated so that the robot configurations involved are appropriate. The human brain, in contrast, uses the motor cortex to generate new motions reusing and combining existing knowledge before executing the motion. We propose a method to generate and control pointing motions for a robot using a biological inspired architecture implemented with spiking neural networks. We outline a simplified model of the human motor cortex that generates motions using motor primitives. The network learns a base motor primitive for pointing at a target in the center, and four correction primitives to point at targets up, down, left and right from the base primitive, respectively. The primitives are combined to reach different targets. We evaluate the performance of the network with a humanoid robot pointing at different targets marked on a plane. The network was able to combine one, two or three motor primitives at the same time to control the robot in real-time to reach a specific target. We work on extending this work from pointing to a given target to performing a grasping or tool manipulation task. This has many applications for engineering and industry involving real robots.

Keywords: neurorobotics, motion generation, spiking neural networks (SNN), pointing a target, motor primitives, humanoid robot (HR), closed-loop

1. INTRODUCTION

The human motor system has been studied for a considerable period of time. Yet, robots lack robust, flexible and adaptive controllers comparable to the human motor system (Pfeifer and Bongard, 2006). One specific example is the capability to generate or pre-shape motions before execution (Shenoy et al., 2013).

Recent studies provide insights into the mechanisms for motion generation in the motor cortex. During reaching, activity in the motor cortex as a whole shows a brief but strong rotational component (Churchland et al., 2012; Russo et al., 2018). Instead of encoding parameters of movement in single neurons, the motor cortex as a whole can be understood as a dynamical system that drives motion. An initial state is produced externally and the system naturally relaxes while producing motor activity, which is then projected down the spinal cord to inter-neurons and

motor-neurons (Churchland et al., 2012; Russo et al., 2018). Neural activity in the motor cortex shows a strong and amplified but stable response to initial activation (Hennequin et al., 2014). There is no broad consensus on the role of the motor cortex in voluntary movement. Nevertheless, neural correlates of many different types of parameters of arm movements have been found in the motor cortex (Kalaska, 2009). This behavior can be replicated by artificial neurons with strong recurrent connections balanced by strong inhibitory connections (Hennequin et al., 2014). Activity in the resulting network closely resembles activity in the motor cortex and can be used as an engine for complex transient motions (Hennequin et al., 2014). For example, in Ayaso (2001) an architecture detailing how to generate motor commands for arm motions is proposed, which also includes how learning and adaptation can be achieved by changing the gain.

A broadly accepted hypothesis is that the central nervous system uses linear combinations of a small number of muscle synergies to produce diverse motor outputs (Bizzi et al., 2008). The activation of the synergies can change based on sensor feedback to produce adaptive motions. The neuron activity in the intermediate zone of the spinal cord resembles motor primitives rather than individual muscles (Hart and Giszter, 2010). These neurons could act as building blocks for more complex voluntary movements. Different approaches have used the concepts of motor primitives to represent and model motions (Schaal, 2006; Tieck et al., 2018b,c). The *dynamic movement primitives* introduces a representation of movement as a spring-damping system in which the goal state is an attractor that allows for easily adaptable complex motor behaviors, both rhythmic and discrete (Schaal, 2006).

A set of approaches implemented with spiking neural networks (SNN) (Maass, 1997; Vreeken, 2003; Walter et al., 2016), represent motion using motor primitives to model target reaching (Tieck et al., 2018c, 2019) and different activation modalities (Tieck et al., 2018b). An SNN that autonomously learns to control a robotic arm through motor babbling and STDP was proposed in Bouganis and Shanahan (2010). In Chadderdon et al. (2012) an SNN is implemented that learns to rotate a single joint to a target and the learning is based on dopamine inspired reinforcement learning with a global reward and punishment signal. In Tieck et al. (2018a) a combination of reinforcement learning with a liquid state machine was used to learn continuous muscle activation of a musculo-skeletal arm.

To control robots in a way closer to biology we can use SNNs to implement models from neuroscience. Using the principles outlined in our previous work on motor primitives (Tieck et al., 2018b,c, 2019) and using the mechanisms for motion generation from the motor cortex (Ayaso, 2001; Hennequin et al., 2014), we can model pointing motions for a humanoid robot.

We propose an SNN that combines a simplified model of the motor cortex to generate motions combining motor primitives to control pointing motions with a humanoid robot arm. Our approach for motion generation (pre-shaping) before execution has three main components (see **Figure 1**): a motion generation layer, a motor control layer with motor primitives and a target representation layer. The motion generation layer produces circular activity that creates the activation patterns

for the primitives. The motor control layer has one base primitive for the pointing motion, and four correction primitives that point to targets left, right, up and down from the base motion target point. The target representation layer takes the target position and based on the relative distance to the base motion target point uses selective disinhibition to activate the correction primitives. We evaluated our approach with a humanoid robot, HoLLiE in Hermann et al. (2013), by defining different targets on a plane and having the robot point to them (see **Supplementary Video 1**).

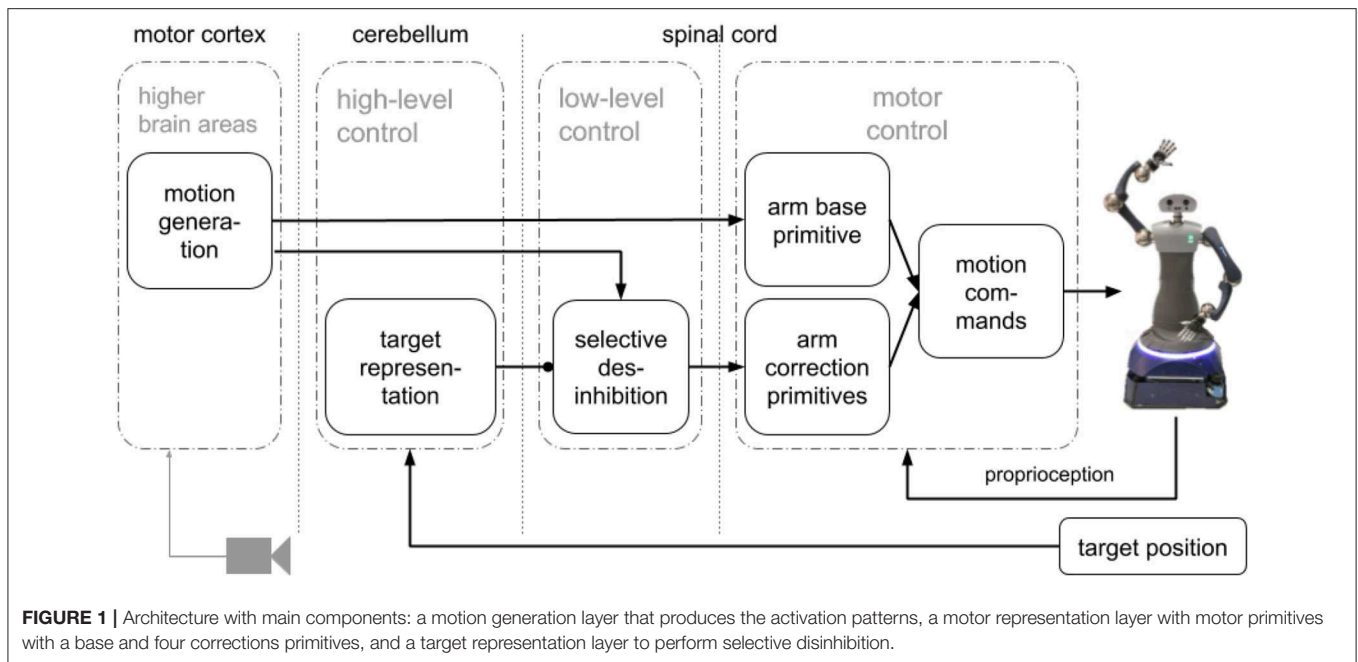
2. APPROACH

Our SNN combines a simplified model of the motor cortex to generate motions combining motor primitives to control pointing motions with a humanoid robot arm. And here we present the details. In the work presented in Tieck et al. (2018c, 2019) we show how to perform online combination of primitives to achieve perception driven target reaching. In this work, the SNN performs motion generation (pre-planning) before execution using a bio inspired architecture.

We formalize the problem as follows: given an initial state of the robot and a set of primitives, move it to a target point on a plane. In classical robotics a system calculates the inverse kinematics (IK) and then validates the configuration to finally generate a motion trajectory. In contrast, our approach can do this without calculating the IK and without validating the resulting configurations. We define motor primitives for the arm as valid possible motions in the working space. The way new motions are generated is by using a base primitive that is activated, combined with a full or partial activation of the correction primitives. By using motor primitives to represent motions, we solve the trajectory generation in the “motor primitive space.” The resulting motions are combination of the primitives, which have no invalid configurations. In this work, we do not consider obstacles.

A go-cue in one neuron initiates circular activity in the motor generation layer that represents the motor cortex (Ayaso, 2001; Kalaska, 2009; Russo et al., 2018). The activity of this layer is used to activate the base and correction motor primitives (Tieck et al., 2018b,c, 2019). Based on an error signal representing the target, the correction primitives are disinhibited and combined with the base (Richter et al., 2012; Sridharan and Knudsen, 2015). The resulting spike activation is decoded to motor commands for the robot joints. The learned weights are the distance based inhibitory connections in motion generation layer, the connections to the base motor primitive, and the connections to the correction primitives. The architecture with the main components is presented in **Figure 1**. It has three main components: a motion generation layer, a motor control layer with motor primitives and a target representation layer.

The motion generation layer produces circular activity that creates the activation patterns for the primitives. A population generates neural activity over a certain period of time. The first step is to normalize spike activation by changing the weights of active neurons to get a similar amount of spikes from the whole



population. Then, to obtain heterogeneity we add an inhibitory population with random connections.

The motor control layer provides the low level motor representation using motor primitives. There is one base motion primitive for pointing to the center, and four correction primitives that point to targets left, right, up and down from the base motion target point. The base primitive is activated and, depending on the target representation signal, the correction primitives are disinhibited.

The target representation layer takes the target position and, based on the relative distance to the base motion target point, uses selective disinhibition to activate the correction primitives. The target signal is the relative position to the base primitive final position, and it is used to regulate the activation of the correction primitives.

2.1. Motion Generation, M1

In the motion generation layer MG there is a group of two recurrent populations representing the motor cortex, one is a 2D grid MG^G and the other is an inhibitory MG^I to obtain heterogeneity (see **Figure 2**). This layer generates circular neural activity over a period of time (Churchland et al., 2012; Russo et al., 2018).

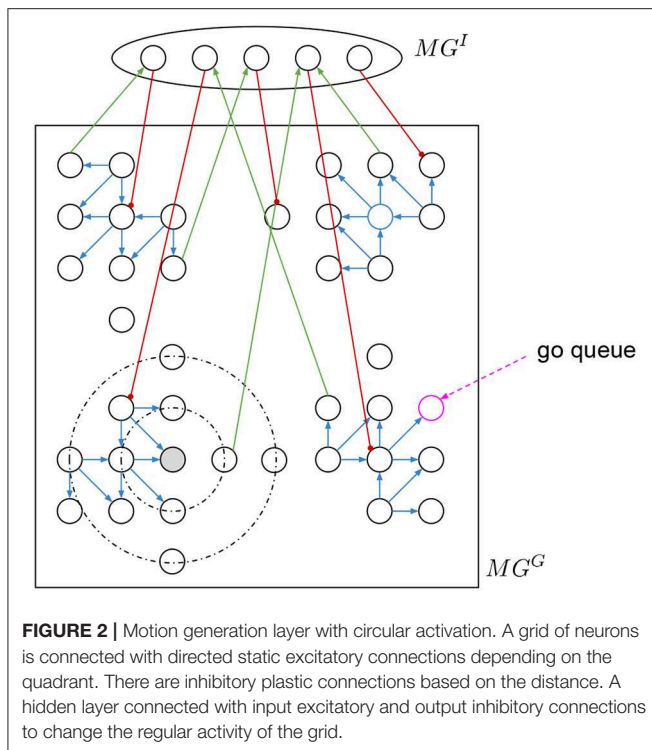
To initialize the motion generation layer there are two steps. First, we stabilize the spike activation in MG^G and second we add the inhibitory connections from and to MG^I . Then we go over all neurons giving a go-cue to each one and we record how long the activity propagates. The go-cue is a continuous input of spikes to the respective neuron during 10 ms. For each motion we select the “go-neuron” as the neuron that produces activity with similar time to the desired motion.

MG^G is square grid of 20×20 neurons with recurrent connections (see **Figure 2**). There are two types of connections,

the directed excitatory to create the circular activity and the local inhibitory to stabilize the activity. The excitatory connections (blue connections in **Figure 2**) are static and have specific directed connectivity depending on the quadrant the neurons area to amplify the activity and force the rotational activation. The distance based local inhibitory connections (black dotted circular lines in **Figure 2**) stabilize the activity.

To normalize the spike activity of MG^G , the inhibitory weights are changed to achieve a specific total activity MG_{norm}^G with the following learning procedure. We add a spike recorder to all MG^G neurons. A go-cue (pink dotted arrow in **Figure 2**) is given as short burst of 10 ms of spikes into one single neuron at a time. This initial neuron is chosen randomly every time, so that there are no “dark” spots in MG^G without spike activity. Every 100 ms Δt (nest.sim(100 ms)) the simulation is stopped. The total spikes of MG^G in that δt are counted as MG_{spikes}^G . If $MG_{spikes}^G > MG_{norm}^G$, then increase the weights by Δw of the inhibitory connections coming out of all active neurons. Else if $MG_{spikes}^G < MG_{norm}^G$, then decrease them. The Δw must be small, so that a weight update does not kill the activity. In other words, we want to regulate the global total activity of the MG^G population, if it is too high then propagate less, if it is too low then propagate more.

After training, once the circular activity propagation of MG^G is stable, we add a small population MG^I with random input and output connections to and from the 2D grid MG^G to obtain heterogeneity. Both, input and output connections are static and random. The output connections—from MG^I to MG^G —are strong inhibitory (red connections in **Figure 2**), and the input connections are excitatory (green connections in **Figure 2**). To set the connections, we set fix numbers of input and output connections, then we sample random neurons from both populations and connect them.



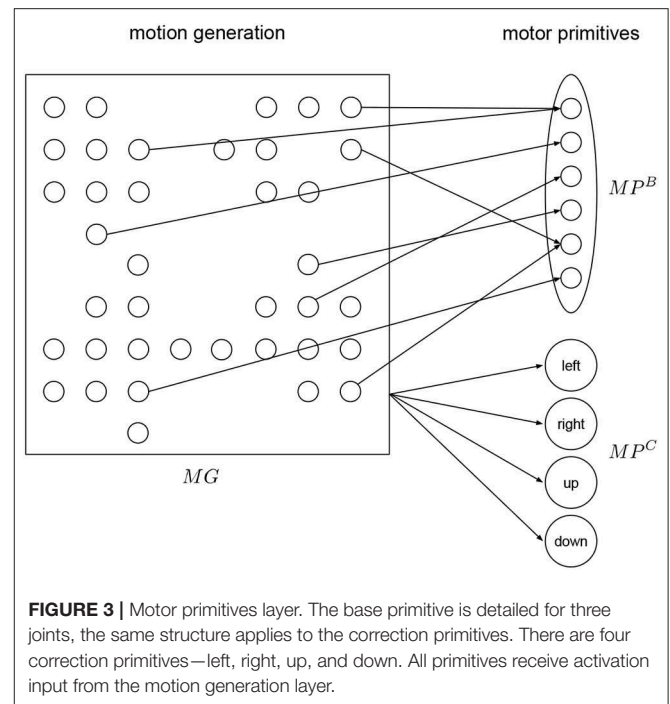
With MG^I and MG^G connected, we then go over all neurons in MG^G to assess the resulting activity. We give again a “go-cue” as short burst of spikes for 10 ms into each single neuron “go-neuron” (pink circle in **Figure 2**), and then measure how long does the activity propagates. The time is measured either until no more spikes occur, or interrupted after a maximum time limit in simulation steps. The activity duration for each “go-neuron” is stored in a table. Then we pick those with similar time to the desired motions, and this will be the “go-neuron” for the primitives.

2.2. Base and Correction Motor Primitives

The motor primitive layer MP is a layer for low level motor representation using motor primitives (Tieck et al., 2018b,c, 2019) (see **Figure 3**). The primitives combine to generate a specific motion activated by the motion generation layer MG . In MP there are populations, one for the base primitive and one for each of the correction primitives. During execution of a motion, the base primitive is activated, and depending on the target representation signal, the correction primitives are activated.

To generate pointing motions in a certain working space, we define the following motion representation. We define first a base primitive MP^B (see **Figure 3**), which is a motion to point at the center of the working space. Then we define four correction primitives MP^C to point at points to the left, right, up and down of the center (see **Figure 3**). This four points define an ellipsoid as the boundary of the working space in the plane.

For each primitive, a different population is connected to MG . Each primitive has two motor neurons per joint in the robot. Each output spike causes small change in the corresponding

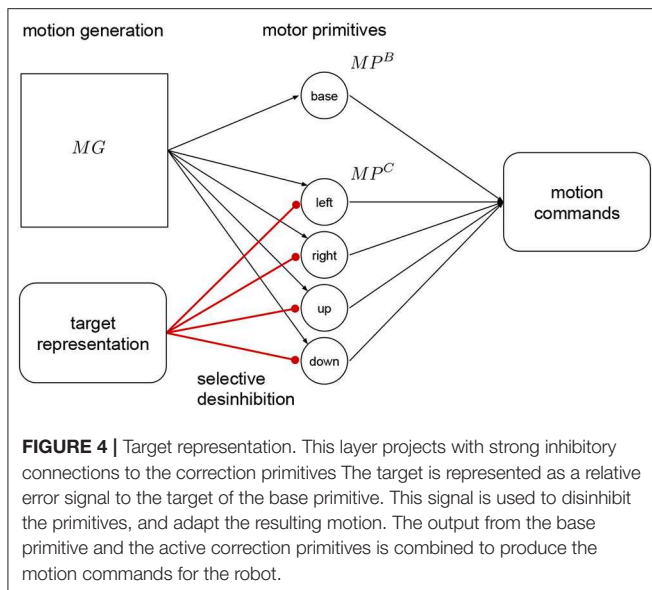


robot joint, it is defined as a fixed gain factor that regulates the speed. There is a detailed view of the primitive population for the base motion in **Figure 3**. The training is done one by one to resemble the exemplary motion. We use supervised learning to minimize the error and adapt the weights and produce a specific motion (Tieck et al., 2018b).

2.3. Target Representation

The target representation layer is connected to the correction primitives with inhibitory synapses as shown in **Figure 4**. The correction primitives are inhibited by default, and they are disinhibited according to required adaptation provided by this layer. This mechanism is called selective disinhibition and it is used for attention mechanisms, decisions and mechanisms for target selection (Richter et al., 2012; Sridharan and Knudsen, 2015). For example, if no correction to the right is necessary, then the right primitive remains fully inhibited. In Kawato (1999) and Wolpert et al. (1998), they see the cerebellum as an internal model that can predict how the end result of a known motion will be like. This prediction can be compared to a desired target to make the respective corrections before execution.

In our approach we use a relative target representation, with the target's relative position to the base primitive final position. That signal is used to regulate the activation of the neurons in this layer, by decreasing the input current proportionally this layer activates the correction primitives using selective disinhibition. This signal translates to the amount or percentage of activation, between 0 and 1, of the respective correction primitives, with 1 being full inhibition and 0 full activation. This adaptation or pre-shaping happens before executing the motion.



3. RESULTS

In most modern and more complex robotic applications motions have to be dynamically generated according to flexible targets or constraints. A major component of many robot tasks is the reaching of specific, often dynamic targets. While this is usually followed by some form of manipulation of an object, the pure act of reaching a specified goal state with a robot manipulator can be understood as a pointing motion. Due to this, we use pointing toward different goal-points on a board plane (see **Figure 5**) to evaluate how well the robot generates adaptive motions.

3.1. Experiment Setup

Initially, base motor primitives have to be learned. A base motion of pointing toward a central target is manually created, but could easily be generated with motion capture or taught-in. The network is then trained to produce this specific pointing motion when all correction primitives are fully inhibited. Afterwards, the base motion is manually adapted toward 4 specified points in the target area, each with a distance of 25 cm from the center to the left, right, top and bottom (red points in **Figure 5**). The correction primitives in the network are trained to produce the difference from the base motion toward these adapted motions, so that as a whole the network produces them when their the corresponding correction primitive is uninhibited.

This allows the network to create different motions by partially inhibiting the corrections primitives. The quality of the generated motions is measured based on the network's ability to point at different targets. The reference points are used as a coordinate system, with positive x-axis representing the inverse inhibition of the right primitive and the negative axis the left primitive, respectively. In the same way, the y-axis represents the up and down primitives. This allows a mapping from every point on the board to specific inhibitions of the correction neurons. A motion is generated with these inhibitions set manually and the final position of the end-effector of the robot

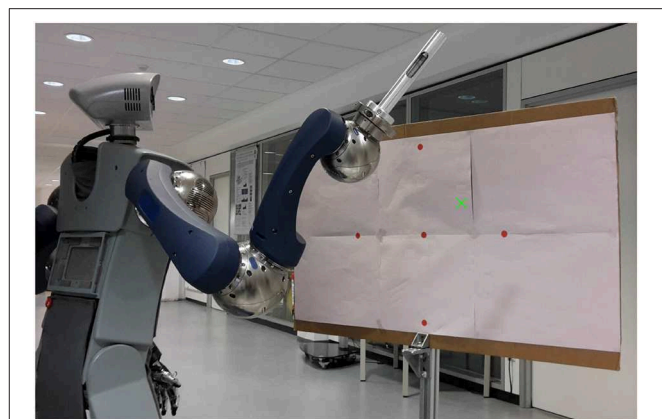


FIGURE 5 | Basic experiment setup. The robot is in a starting position in front of the board plane and will produce a motion toward a target point (green x). Red points show the targets for the base and correction primitives that are already learned.

is then compared with the intended goal. The distance between actual and target position is used as a measure of error in the following experiments.

3.2. Humanoid Robot HoLLiE

HoLLiE, Hermann et al. (2013) is a mobile service robot with two functional arms and humanoid hands (see **Figure 5**). The robot was developed at the FZI Research Center for Information Technology for different tasks, such as accompanying visitors and mobile manipulation (see¹). With a range of different sensors and a highly articulated body HoLLiE can handle everyday objects, interact with humans in multiple ways and therefore be employed in various service robotic scenarios. For these characteristics HoLLiE was chosen to achieve human-like pointing motions, as the arms are mounted on an upper body in a similar kinematics to a human arm.

3.3. Implementation Details

Motions are generated by an SNN using the PyNN API implemented in NEST, Diesmann and Gewaltig (2001) running on a laptop computer. We use *Robot Operating System (ROS)*² as a communication layer to connect NEST with the robot.

The SNN was simulated in steps of 100 ms and the spikes in this time frame were accumulated before being sent to the robot. This frequency is enough to generate smooth real-time robot movements, and a complete pointing motion takes about 10s. The generated spikes in the output of the motor-neurons were directly decoded into changes in joint values for the robot. The neuron activity is decoded by changing joint position by a fixed value for each spike. The resulting joint values were then used as goals for the joint trajectory controller in ROS.

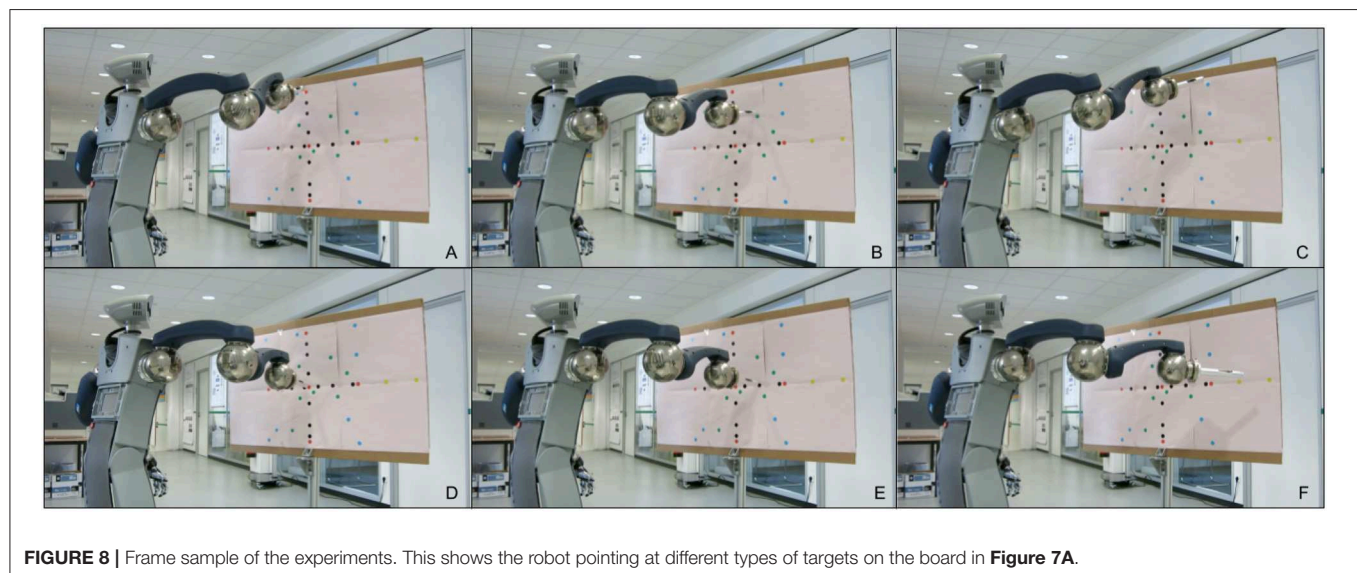
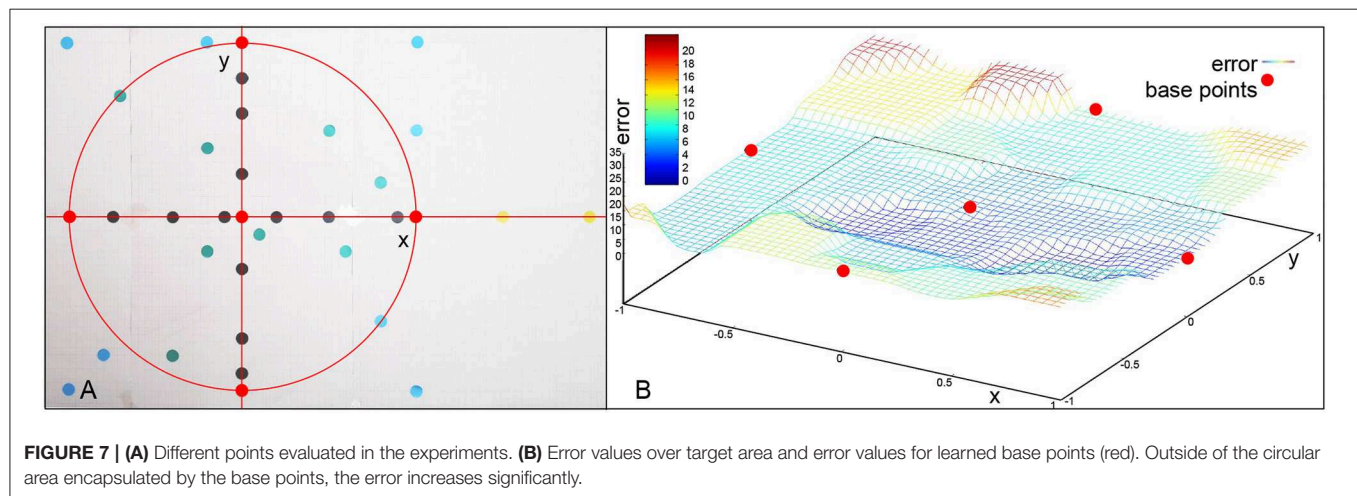
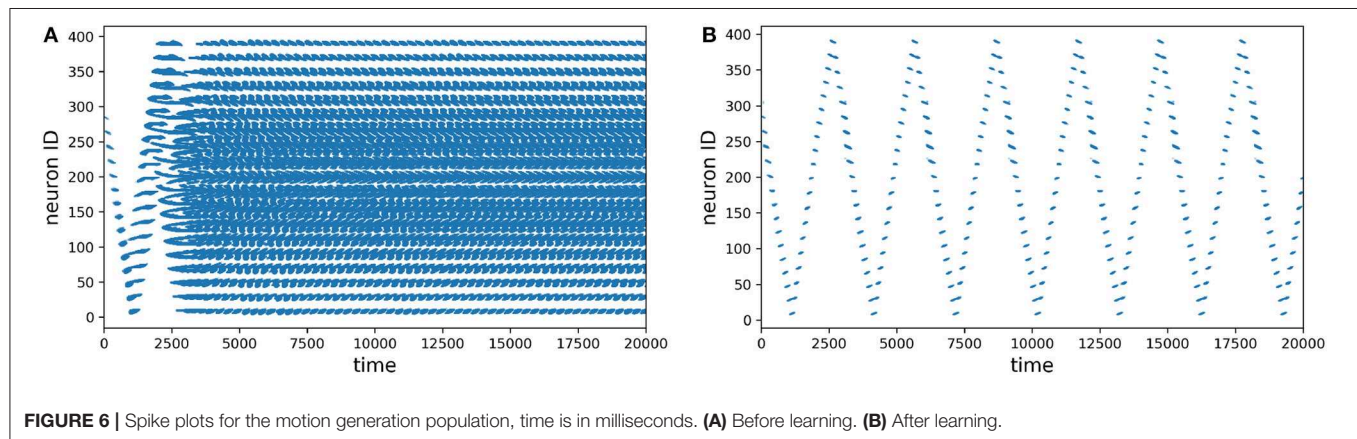
During training of MG^G , the weights of one iteration are stored in a dictionary data-structure where all the required weight updates are performed. Only after all updates have been calculated, the “set weights” function in NEST is called, as

¹<https://www.fzi.de/en/research/projekt-details/hollie/>

²<http://www.ros.org/>

constant weight changes are greatly reduce the simulation time for little gain. Using this, the total training time could be reduced to about 1 h on a single processor.

The network is implemented with basic leaky integrate and fire neurons LIF. The layer MG is built as a population organized in a grid of 20×20 neurons MG^G and an inhibitory population



of 20 neurons MG^I . For each of the 5 motor primitives MP (one base and four correction) 2 neurons are used per joint, with three active joints being used for the evaluated motions, for a total of 30 neurons. The total SNN contains 450 neurons and about 20,000 synapses.

3.4. Experiment

The first thing we evaluated was how does the learning in the network work, specially in the motion generation layer. In **Figure 6** we recorded the spike activity of all the neurons before and after learning. Without learning, you can see on the left how the go-cue propagates in the neurons and then saturates, producing chaotic activation. After learning, you can see on the right how the activation of the population is periodic (circular) and is stable.

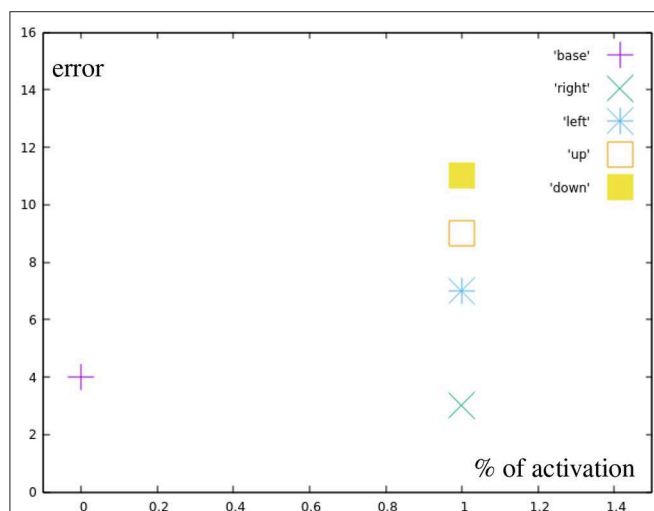


FIGURE 9 | Error by distance to center for reaching all the base points (red in **Figure 7A**).

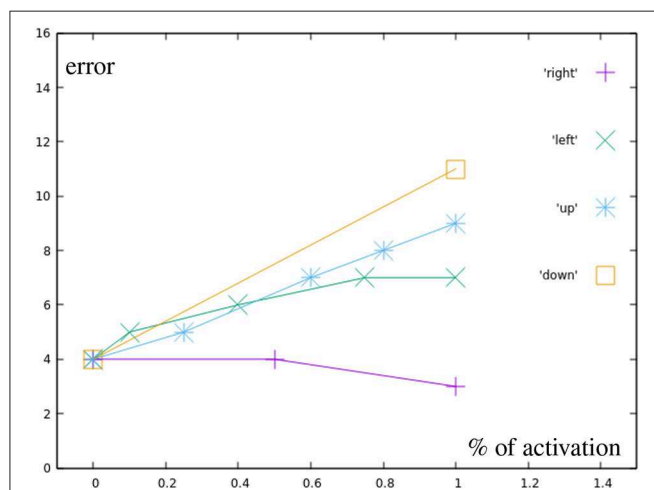


FIGURE 10 | Error by distance to center for different correction primitives (black in **Figure 7A**). Right is the most accurately learned primitive, down the least accurate one.

Throughout the experiments different types of targets were attempted to be reached based on the board displayed in **Figure 7**. The distance from the target in millimeters is used as an error for evaluation. The base motion is the center red dot. The correction primitives are the red dots on the circle. If we only use one of the correction primitives at a time, we obtain black dots. A combination of multiple correction primitives are the green dots. The blue dots are outside of the working space, but still in the primitive space. The yellow dots on the right are extrapolations. The frame sample in **Figure 8** shows the robot pointing at different types of targets on the board.

Red points represent the targets for the manually designed base motions that can be reached by fully inhibiting all or all but one correction primitives. **Figure 9** shows the errors for the different base motions. It can be seen, that they are not hit completely accurately, which results from the relatively high impact single spike inaccuracies have on the end position.

The black points represent motions using only a single, partially inhibited correction primitive. **Figure 10** shows that there is no additional error created by partially inhibiting the primitives, other than the already existing inaccuracy in the learned motions themselves. Green points display motions combining two correction primitives, but with a total distance from the base motion not greater than one full correction primitive.

While **Figure 11** can not show as easily how the in these targets results purely from the base primitives, with the exception of one point directly on the circular test area all motions produced a smaller error than the most incorrect base motion. This again suggests, that no additional error is added through the combination of two correction primitives. The light blue points are also created by combining two correction primitives, in this case, though, their distance to the base is greater than one the distance of one primitive.

These results (**Figure 12**) show errors that do not seem to simply happen from inaccuracies in the learned motions. The

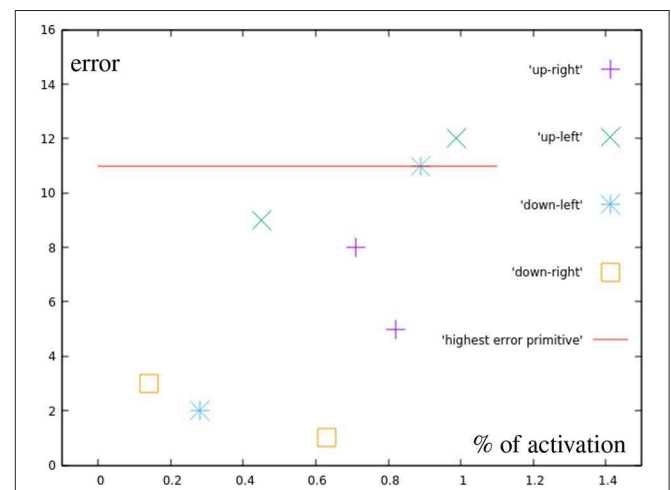
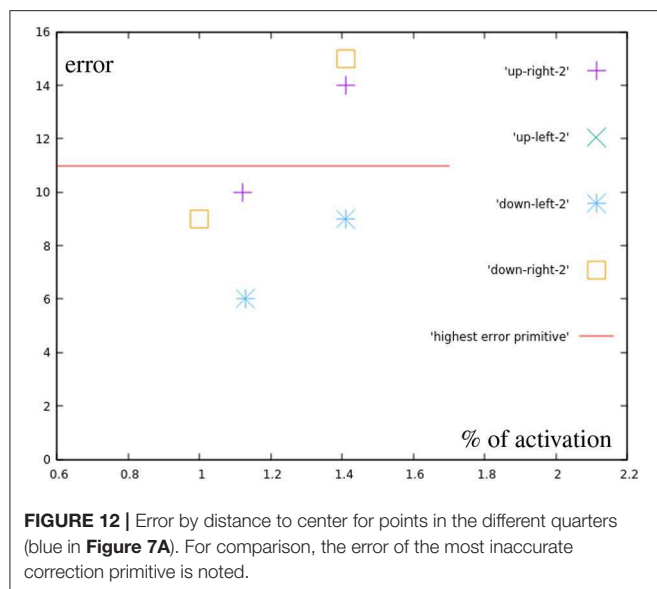


FIGURE 11 | Error by distance to center for points in the different quarters (green in **Figure 7A**). For comparison, the error of the most inaccurate correction primitive is noted.



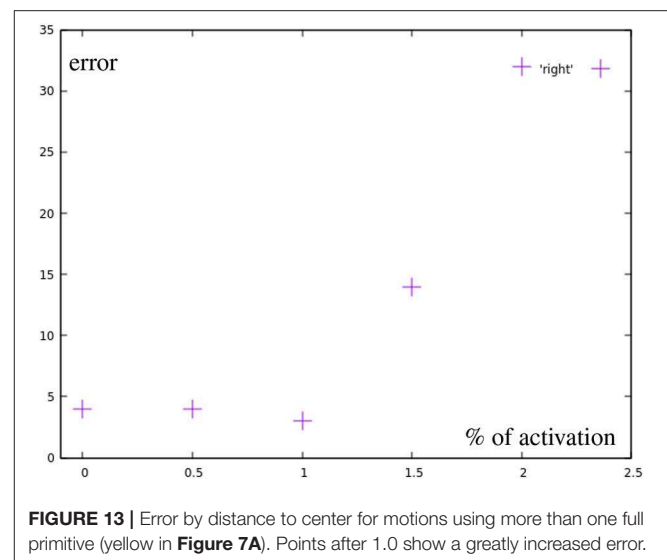
upper right point using both primitives fully, for example, generates an error of 14 millimeters, while the sum of the errors of both used correction primitives is only 12 mm.

All marked points are well within the workspace of the robot. But yellow points are not reachable with the defined primitives, meaning an activation of 1 (100) of any combination of the motions will not go outside of the bounds defined by the primitives (red dots). Moreover, outside of the circular area used in the previous experiment the method of combining primitives loses precision, as a consequence. Finally, the yellow points are actually unreachable by combining primitives with total activation. An extrapolation from the right primitive would be necessary. To accomplish this, the right primitive is not only uninhibited, but additional spikes are added to generate more activity. So, there is a correlation of the error with the positions of the learned base motions. As **Figure 13** shows, while direction of the adaptation is correlated, the error is greatly increased and a precise correction does not occur. The total errors over the target area can be seen in **Figure 7B**. In a circular area between the base motions, the error can be reduced to inaccuracies in learning, while outside of this area additional errors can be measured.

4. DISCUSSION

Based on the results and the evaluation from the experiments we can highlight certain aspects. If the target distance is of one correction primitive or less, then there is no significant added error through adaptation. If there is a higher distance, then the error increases. The error gets a relatively high impact from single spikes and a reduction by using larger populations and different encoding techniques would allow for more precision. This is a low level control problem, and we currently work on a spike based controller for ROS to achieve smooth control.

We successfully implemented and tested an SNN for voluntary adaptive motions using an architecture based on recent theories



about motion generation in the central nervous system. The network was able to pre-shape motions and generate new trajectories before the execution by combining primitives using selective disinhibition. The SNN was able to control a real humanoid robot in real-time in a closed-loop scenario. This approach can be used with different robot arms, and is not dependent on a specific kinematic structure.

In the future we want to benchmark the technical aspects, and increase the precision and speed of the motions. With the recent advances in backpropagation-like learning rules for SNN as in Kaiser et al. (2019), we can learn different motion types for different tasks in same network, and start them with different go-cues. We also want to integrate event-based vision to this system to get the target and drive the adaptation as in Kaiser et al. (2016), and to explore learning by demonstration as in Kaiser et al. (2018). We work on extending this work from pointing to a given target to perform there a grasping or tool manipulation task. This has many applications for engineering and industry with real robots.

DATA AVAILABILITY

All datasets analyzed for this study are included in the manuscript and the **Supplementary Files**.

AUTHOR CONTRIBUTIONS

All authors participated in writing the paper. JCVT, TS, JK, and FM conceived the experiments and analyzed the data.

FUNDING

This research has received funding from the European Union's Horizon 2020 Framework Programme for Research and

Innovation under the Specific Grant Agreement No. 785907 (Human Brain Project SGA2).

ACKNOWLEDGMENTS

We would like to thank the team at FZI Forschungszentrum Informatik for the discussions and support during the

development of this work, specially those involved in the development of HoLLiE.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fnbot.2019.00077/full#supplementary-material>

REFERENCES

- Ayaso, O. (2001). *A model for command generation in motor cortex* (Ph.D. thesis). Cambridge, MA: Massachusetts Institute of Technology.
- Bizzi, E., Cheung, V. C. K., Avella, A., Saltiel, P., and Tresch, M. (2008). Combining modules for movement. *Brain Res. Rev.* 57, 125–33. doi: 10.1016/j.brainresrev.2007.08.004
- Bouganis, A., and Shanahan, M. (2010). “Training a spiking neural network to control a 4-DoF robotic arm based on spike timing-dependent plasticity,” in *The 2010 International Joint Conference on Neural Networks (IJCNN)* (Barcelona: IEEE), 1–8. doi: 10.1109/IJCNN.2010.5596525
- Chadderdon, G. L., Neymotin, S. A., Kerr, C. C., and Lytton, W. W. (2012). Reinforcement learning of targeted movement in a spiking neuronal model of motor cortex. *PLoS ONE* 7:e47251. doi: 10.1371/journal.pone.0047251
- Churchland, M. M., Cunningham, J. P., Kaufman, M. T., Foster, J. D., Nuyujukian, P., Ryu, S. I., et al. (2012). Neural population dynamics during reaching. *Nature* 487, 1–20. doi: 10.1038/nature11129
- Diesmann, M., and Gewaltig, M.-O. (2001). NEST: an environment for neural systems simulations. *Forschung und wissenschaftliches Rechnen, Beiträge zum Heinz-Billing-Preis* 58, 43–70.
- Hart, C. B., and Giszter, S. F. (2010). A neural basis for motor primitives in the spinal cord. *J. Neurosci.* 30, 1322–1336. doi: 10.1523/JNEUROSCI.5894-08.2010
- Hennequin, G., Vogels, T. P., and Gerstner, W. (2014). Optimal control of transient dynamics in balanced networks supports generation of complex movements. *Neuron* 82, 1394–1406. doi: 10.1016/j.neuron.2014.04.045
- Hermann, A., Sun, J., Xue, Z., Ruehl, S. W., Oberländer, J., Rönnau, A., et al. (2013). “Hardware and software architecture of the bimanual mobile manipulation robot HoLLiE and its actuated upper body,” in *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (Wollongong, NSW: IEEE), 286–292.
- Kaiser, J., Friedrich, A., Tieck, J. C. V., Reichard, D., Roennau, A., Neftci, E., et al. (2019). Embodied event-driven random backpropagation. *arXiv:1904.04805*.
- Kaiser, J., Melbaum, S., Tieck, J. C. V., Roennau, A., Butz, M. V., and Dillmann, R. (2018). “Learning to reproduce visually similar movements by minimizing event-based prediction error,” in *2018 7th IEEE International Conference on Biomedical Robotics and Biomechanics (Biorob)* (Enschede: IEEE), 260–267.
- Kaiser, J., Tieck, J. C. V., Hubschneider, C., Wolf, P., Weber, M., Hoff, M., et al. (2016). “Towards a framework for end-to-end control of a simulated vehicle with spiking neural networks,” in *IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots, SIMPAR 2016* (San Francisco, CA: IEEE), 127–134. doi: 10.1109/SIMPAR.2016.7862386
- Kalaska, J. F. (2009). “From intention to action: motor cortex and the control of reaching movements,” in *Progress in Motor Control* (Springer), 139–178. doi: 10.1007/978-0-387-77064-2
- Kawato, M. (1999). Internal models for motor control and trajectory planning. *Curr. Opin. Neurobiol.* 9, 718–727. doi: 10.1016/S0959-4388(99)00028-8
- Maass, W. (1997). Networks of spiking neurons: the third generation of neural network models. *Neural Networks* 10, 1659–1671. doi: 10.1016/S0893-6080(97)00011-7
- Pfeifer, R., and Bongard, J. (2006). *How the Body Shapes the Way We Think: A New View of Intelligence*. Cambridge, MA: MIT Press.
- Richter, M., Sandamirskaya, Y., and Schöner, G. (2012). “A robotic architecture for action selection and behavioral organization inspired by human cognition,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Vilamoura: IEEE), 2457–2464.
- Russo, A. A., Bittner, S. R., Perkins, S. M., Seely, J. S., London, B. M., Lara, A. H., et al. (2018). Motor cortex embeds muscle-like commands in an untangled population response. *Neuron* 97, 953–966. doi: 10.1016/j.neuron.2018.01.004
- Schaal, S. (2006). “Dynamic movement primitives—a framework for motor control in humans and humanoid robotics,” in *Adaptive motion of Animals and Machines* (Tokyo: Springer), 261–280.
- Shenoy, K. V., Sahani, M., and Churchland, M. M. (2013). Cortical control of arm movements: a dynamical systems perspective. *Annu. Rev. Neurosci.* 36, 337–359. doi: 10.1146/annurev-neuro-062111-150509
- Sridharan, D., and Knudsen, E. I. (2015). Selective disinhibition: a unified neural mechanism for predictive and *post hoc* attentional selection. *Vis. Res.* 116, 194–209. doi: 10.1016/j.visres.2014.12.010
- Tieck, J. C. V., Pogančić, M. V., Kaiser, J., Roennau, A., Gewaltig, M.-O., and Dillmann, R. (2018a). “Learning continuous muscle control for a multi-joint arm by extending proximal policy optimization with a liquid state machine,” in *International Conference on Artificial Neural Networks ICANN 2018* (Rhodes: Springer), 211–221. doi: 10.1007/978-3-030-01418-6_21
- Tieck, J. C. V., Steffen, L., Kaiser, J., Arne, R., and Dillmann, R. (2018b). “Multi-modal motion activation for robot control using spiking neurons,” in *2018 IEEE International Conference on Biomedical Robotics and Biomechanics (BioRob)* (Enschede).
- Tieck, J. C. V., Steffen, L., Kaiser, J., Reichard, D., Roennau, A., and Dillmann, R. (2019). Combining motor primitives for perception driven target reaching with spiking neurons. *Int. J. Cogn. Inform. Natural Intell.* 13, 1–12. doi: 10.4018/IJCINI.2019010101
- Tieck, J. C. V., Steffen, L., Kaiser, J., Roennau, A., and Dillmann, R. (2018c). “Controlling a robot arm for target reaching without planning using spiking neurons,” in *2018 IEEE 17th International Conference on Cognitive Informatics & Cognitive Computing (ICCI*CC)* (Berkeley, CA: IEEE), 111–116.
- Vreeken, J. (2003). *Spiking Neural Networks, An Introduction*. Utrecht University: Information and Computing Sciences.
- Walter, F., Röhrbein, F., and Knoll, A. (2016). Computation by time. *Neural Process. Lett.* 44, 103–124. doi: 10.1007/s11063-015-9478-6
- Wolpert, D. M., Miall, R. C., and Kawato, M. (1998). Internal models in the cerebellum. *Trends Cogn. Sci.* 2, 338–347. doi: 10.1016/S1364-6613(98)01221-2

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Tieck, Schnell, Kaiser, Mauch, Roennau and Dillmann. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Response Dynamics in an Olivocerebellar Spiking Neural Network With Non-linear Neuron Properties

Alice Geminiani^{1,2*}, Alessandra Pedrocchi², Egidio D'Angelo^{1,3} and Claudia Casellato¹

¹ Department of Brain and Behavioral Sciences, University of Pavia, Pavia, Italy, ² NEARLab, Department of Electronics, Information and Bioengineering, Politecnico di Milano, Milan, Italy, ³ IRCCS Mondino Foundation, Pavia, Italy

Sensorimotor signals are integrated and processed by the cerebellar circuit to predict accurate control of actions. In order to investigate how single neuron dynamics and geometrical modular connectivity affect cerebellar processing, we have built an olivocerebellar Spiking Neural Network (SNN) based on a novel simplification algorithm for single point models (Extended Generalized Leaky Integrate and Fire, EGLIF) capturing essential non-linear neuronal dynamics (e.g., pacemaking, bursting, adaptation, oscillation and resonance). EGLIF models specifically tuned for each neuron type were embedded into an olivocerebellar scaffold reproducing realistic spatial organization and physiological convergence and divergence ratios of connections. In order to emulate the circuit involved in an eye blink response to two associated stimuli, we modeled two adjacent olivocerebellar microcomplexes with a common mossy fiber input but different climbing fiber inputs (either on or off). EGLIF-SNN model simulations revealed the emergence of fundamental response properties in Purkinje cells (burst-pause) and deep nuclei cells (pause-burst) similar to those reported *in vivo*. The expression of these properties depended on the specific activation of climbing fibers in the microcomplexes and did not emerge with scaffold models using simplified point neurons. This result supports the importance of embedding SNNs with realistic neuronal dynamics and appropriate connectivity and anticipates the scale-up of EGLIF-SNN and the embedding of plasticity rules required to investigate cerebellar functioning at multiple scales.

Keywords: olivocerebellar circuit, spiking neural network (SNN), point neuron, non-linear neuronal dynamics, eyeblink response

OPEN ACCESS

Edited by:

Mario Senden,
Maastricht University, Netherlands

Reviewed by:

Christian Hansel,
The University of Chicago,
United States
Pablo Varona,
Autonomous University of Madrid,
Spain

*Correspondence:

Alice Geminiani
alice.geminiani@unipv.it

Received: 28 June 2019

Accepted: 10 September 2019

Published: 01 October 2019

Citation:

Geminiani A, Pedrocchi A,
D'Angelo E and Casellato C (2019)
Response Dynamics in an
Olivocerebellar Spiking Neural
Network With Non-linear Neuron
Properties.
Front. Comput. Neurosci. 13:68.
doi: 10.3389/fncom.2019.00068

INTRODUCTION

A broad set of experimental observations has suggested that cerebellar circuit functioning relies on a number of detailed features distributed over multiple scales. Single neuron properties along with an organized modular connectivity shape population-specific spiking patterns and spatio-temporal network dynamics, which in turn determine the relationship between input stimuli and responses. The precise encoding of spatio-temporal features into the output (which is in motor domain) corresponds to the cerebellar contribution in sensorimotor tasks (Llinas and Negrello, 2011;

Llinás, 2014; D'Angelo, 2018). Indeed, together with synaptic plasticity, single neuron electroresponsiveness and network connectivity affect motor learning and alterations of these elements can significantly compromise movement adaptation (Peter et al., 2016).

At the cerebellar input, the Granular layer is thought to act as a spatio-temporal filter of sensory inputs (Marr, 1969). This operation has been related to specific properties of Golgi cells (GoCs) and Granule cells (GrCs), such as oscillatory and resonant dynamics, along with the arrangement of microcircuit connectivity, which includes recurrent GoC-GrC inhibitory loops and GoC local networks (D'Angelo et al., 2013; Gandolfi et al., 2013). The GoCs contribute to process sensory signals coming from Mossy Fibers (MFs) by shaping the activity of GrCs. GrC signals converge to the Molecular and Purkinje cell layers through Ascending Axons (AAs) and Parallel Fibers (PFs), with a very precise geometrical organization. Purkinje cells (PCs) are the final integrators of the cerebellar cortex, inhibiting the cerebellar output that drives motor responses (Heiney et al., 2014). *In vivo*, intrinsic simple spikes of PCs are modulated by excitation from GrCs and inhibition from Molecular Layer Interneurons (MLIs). Moreover, inputs from Inferior Olive (IO), through Climbing Fibers (CFs), elicit PC complex spikes (Davie et al., 2008). Deep Cerebellar Nuclei cells (DCNs) are the only output of the cerebellar circuit, projecting centrally to multiple brain areas, and peripherally to the motor pathways. Integrating the inputs from the cerebellar cortex and MFs, DCNs can modify their spontaneous firing and generate pauses and bursts. Burst-pauses in PCs and pause-bursts in DCN cells are thought to be essential to finely tune the motor responses (Shadmehr, 2017). DCNs also continuously control learning processes through inhibitory feedback loops to the IO (De Zeeuw et al., 2011). The PC-DCN-IO loop connections are organized to form microcomplexes: CFs from IO sub-regions project to different sagittal stripes of PCs, which in turn receive signals from subvolumes of the granular layer and of the molecular layer (i.e., microzones); then, PCs of a microcomplex target the corresponding nuclear regions reached by the same CFs (Llinás and Negrello, 2011; Ruigrok, 2011; D'Angelo et al., 2013). On the other hand, GrCs project in the medio-lateral direction by PFs (Uusisaari and de Schutter, 2011), carrying the same signals transversally to multiple microcomplexes. The result is a modular geometrically-organized architecture, where each microcomplex integrates sensorimotor information from different sources and emits spike patterns that, in turn, correlate with specific aspects of behavior (Zhou et al., 2014; Powell et al., 2015).

In this scenario, single neuron properties and cerebellar connectivity are sufficiently well characterized and can be simplified to simulate behavioral tasks using bioinspired cerebellar models (Yamazaki and Igarashi, 2013; Casellato et al., 2014; Antonietti et al., 2016). However, the key causal relationships across scales, i.e., from neuron properties to network dynamics and finally to behavior, are still unclear. To what extent do intrinsic excitability and synaptic inputs contribute to the spiking patterns of PCs and DCN cells during

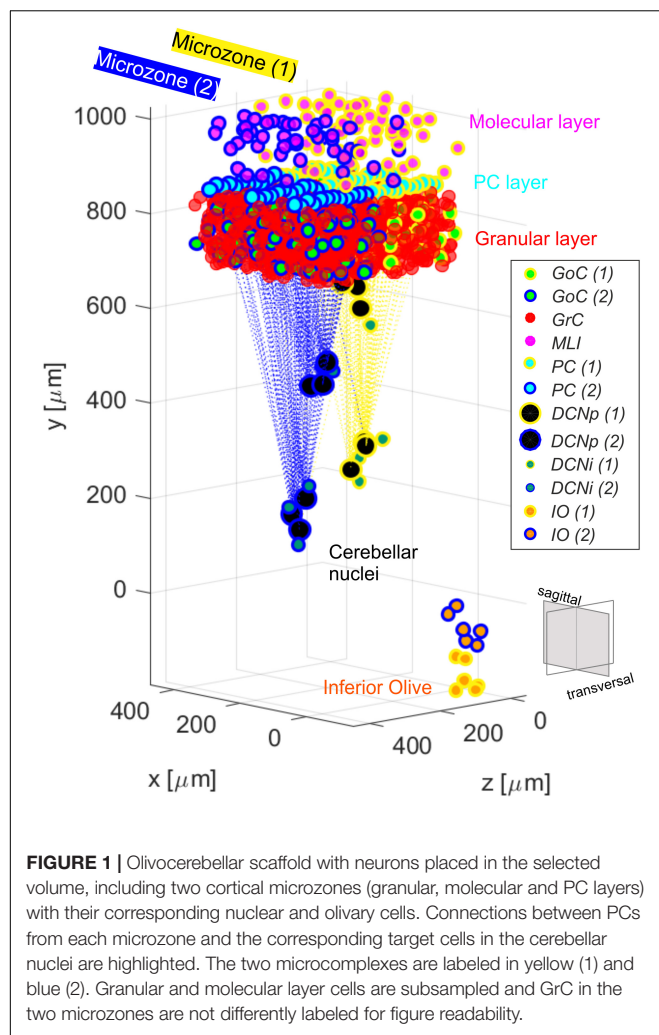
a behavioral task? How do complex firing patterns emerge in cascade within the network?

Here, we have reconstructed and simulated an olivocerebellar microcircuit by integrating monocompartmental neurons with complex electroresponsiveness into the geometrically-organized connectivity of a spiking neural network (SNN). The simulations provide the network with sensory-like stimulation patterns and monitor the microcircuit responses. Such a computational tool compromises between biological plausibility and computational load, allowing a multiscale investigation of the cerebellar network. This is achieved by integrating two main aspects. The first one is the Extended-Generalized Leaky Integrate and Fire (EGLIF) point neuron that maintains salient electrophysiological features – autorhythm, bursts, adaptation, oscillations and resonance – by using just a few state variables (Geminiani et al., 2018b). The EGLIF proved capable to reproduce the rich set of firing patterns of the main olivocerebellar neurons: GoCs, GrCs, PCs, MLIs, DCNs, and IO (Geminiani et al., 2019). The second aspect is network geometry derived from a cerebellar scaffold model, which reproduces the physiological convergence and divergence ratios of connections with a realistic spatial organization (Casali et al., 2019). Here, EGLIF neurons are here evaluated within the whole SNN, where positioning and connectivity of each neuron type are based on their morphology and density within the cerebellar microcircuit (Casali et al., 2019). Therefore, the EGLIF-SNN is exploited to investigate how single neuron properties and network architecture allow the emergence of spatio-temporal dynamic properties, such as burst-pause in PCs and pause-burst in DCN cells. In particular, the EGLIF-SNN is tested by using input patterns encoding two types of sensory signals, whose timing association elicits an eyeblink motor response with multiple afferent pathways specifically activating interconnected microcomplexes (De Zeeuw et al., 2011). The simulations using EGLIF-SNN have been compared to others using simple LIF neurons, in order to understand the impact of single neuron dynamics on network functioning and signal encoding. These results provide a critical assessment of the role of microcircuit properties needed for future closed-loop simulations of cerebellum-driven learning tasks (D'Angelo et al., 2016).

MATERIALS AND METHODS

Reconstruction of the Olivocerebellar Network

To evaluate the role of single neuron electrophysiology and, at the same time, of geometrical and statistical connectivity, a SNN was developed, reproducing an olivocerebellar volume. The reconstructed volume included 96'767 neurons and 4'151'182 total synapses and represented a portion of two cerebellar microcomplexes with the corresponding olivary nuclei (Figure 1). The SNN was built based on the cerebellar scaffold developed in Casali et al. (2019). In this scaffold, neurons were placed in the selected volume based on known cell densities from neurophysiology and geometric features. Then, they were connected according to connectivity rules based



on proximity of neuronal processes (pre-synaptic axon span extension and post-synaptic dendritic field extension) and on statistical convergence/divergence ratios (Casali et al., 2019). The starting network version was made up of cells distributed in a multi-layered volume including the Molecular, Purkinje and Granular layers of the cerebellar cortex – $400 \times 330 \times 400 \mu\text{m}^3$, and the underlying cerebellar nuclei – $200 \times 600 \times 200 \mu\text{m}^3$ (Table 1). The thickness (along y-direction) was fixed based on neurophysiology (330 μm for cortex + 600 μm for nuclei), while the other two sizes (x and z) were flexible, and there defined in order to have a complete exemplificative reconstruction, able to include all the elements in a functional representative module. Here, we subdivided the scaffold cortex into two sub-volumes, by a parasagittal plane, so obtaining two microzones with a transversal length of 200 μm each (along z-axis). Consequently, we reorganized the PC-DCN connections to be confined within the same subvolumes, with a neurophysiological crosstalk. This way, two adjacent microcomplex volumes were reconstructed (Uusisaari and de Schutter, 2011). Then, we added an olivary volume of $100 \times 200 \times 40 \mu\text{m}^3$ chosen to maintain the ratio between the cerebellar cortical volume and the olivary one

measured in mice, i.e., $\sim 66\text{--}68:1$ (Lein et al., 2007). Based on IO neuron density (i.e., $\sim 15'172 \text{ cells}/\text{mm}^3$), we positioned 12 cells in the olivary scaffold volume (Zanjani et al., 2004). The neurons were placed using self-avoiding bounded random walk procedure. For each olivocerebellar microcomplex, six IO neurons were included.

In the cerebellar nuclei, we considered two types of neurons: non-GABAergic DCNs, which are the principal large neurons projecting outside the cerebellum in an excitatory way (DCNp), and GABAergic interneurons (DCNi), which send inhibitory feedback signals to IO. For each DCNp, already present in the previous scaffold release (Casali et al., 2019), we added one DCNi, positioned around the corresponding DCNp, at a random distance d in the range between d_1 (minimum to avoid somata overlap) and d_2 (maximum in order to have a DCNi as a satellite of a specific DCNp, i.e., closer to that DCNp than to the other DCNp neurons):

$$d_1 = r_{\text{DCNp}} + r_{\text{DCNi}}$$

$$d_2 = \text{mean_dist}/4 - r_{\text{DCNp}} - r_{\text{DCNi}}$$

where:

$r_{\text{DCNp}}, r_{\text{DCNi}}$ = radius of DCN neurons' somata;

mean_dist = mean pairwise distance between DCNp in the scaffold (Casali et al., 2019).

Connections to and from IO were organized to mimic the geometry of microcomplexes. IO and DCN neurons were divided into two clusters based on their position and connected to PCs in homologous microzones. This topological segregation was maintained also in connecting IO to DCNp, and DCNi to IO cells.

Furthermore, also the connections from IO to MLIs were introduced following the microcomplex correspondence (Szapiro and Barbour, 2007; Jörntell et al., 2010). The resulting convergence/divergence values of the connections within the entire olivocerebellar scaffold are reported in Table 2.

Single neurons in the SNN were modeled as EGLIF, able to reproduce the full set of spiking patterns of cerebellar neurons (Geminiani et al., 2018b). In details, a cell-specific parameter set was applied to meet the electroresponsive phenotype of each olivocerebellar neuron (e.g., GoC: autorhythm, adaptation, rebound bursting, phase reset, subthreshold oscillations, resonance; GrC: subthreshold oscillations and

TABLE 1 | Neuron types and numbers in the olivocerebellar scaffold.

Neuron type	Number of neurons
MF	7073
GoC	219
GrC	88164
MLI	1206
PC	69
DCNp	12
DCNi	12
IO	12

TABLE 2 | Olivocerebellar scaffold connections with convergence/divergence ratios (reported as mean \pm Standard Deviation, SD) and corresponding synaptic parameters.

	Convergence (mean \pm SD)	Divergence (mean \pm SD)	Weight [nS]	Delay [ms]	τ_{α} [ms]	Connection type
MF-GrC	4	50 \pm 22	0.15	4.0	5.8 (Prestori et al., 2008)	exc
MF-GoC	65 \pm 27	2 \pm 1	1.5	4.0	0.23 (Kanichay and Silver, 2008)	exc
GoC-GrC	2 \pm 1	624 \pm 267	0.6	2.0	13.6 (Mapelli et al., 2009)	inh
GoC-GoC	34 \pm 8	34 \pm 9	0.3	1.0	10	inh
AA-GoC	360 \pm 81	1	1.2	2.0	0.5 (Kanichay and Silver, 2008)	exc
PF-GoC	1600	4 \pm 2	0.05	5.0	0.5 (Kanichay and Silver, 2008)	exc
MLI-MLI	4 \pm 2	4	0.2	1.0	2	inh
PF-MLI	1004 \pm 221 (BC) 1021 \pm 221 (SC)	12 \pm 4 (BC) 12 \pm 5 (SC)	0.015	5.0	0.64	exc
MLI-PC	20	3 \pm 1	0.3	4.0 (BC) 5.0 (SC)	2.8 (He et al., 2015)	inh
AA-PC	249 \pm 13	1	0.7	2.0	1.1 (Miyazaki et al., 2004)	exc
PF-PC	28401 \pm 776	23 \pm 3	0.007	5.0	1.1 (Miyazaki et al., 2004)	exc
PC-DCNp	26 \pm 2	5 \pm 1	0.4	4.0	0.7 (Uusisaari and Knöpfel, 2008)	inh
PC-DCNi	26 \pm 4	5 \pm 1	0.12	4.0	1.14 (Uusisaari and Knöpfel, 2008)	inh
MF-DCNp	147	1	0.05	4.0	1 (Wu and Raman, 2017)	exc
CF-PC	1	6 \pm 1 (min = 4; max = 8)	350.0	4.0 (Llinás, 2014)	0.4 (Miyazaki et al., 2004)	exc
CF-MLI	3 \pm 1	115 \pm 23	1.0	70.0 \pm 10.0 (De Zeeuw et al., 2011)	1.2 (Szapiro and Barbour, 2007)	exc
IO-DCNp	6	6	0.1	4.0 (Hoebeek et al., 2010)	1 (Wu and Raman, 2017)	exc
IO-DCNi	6	6	0.2	5.0	3.64 (Uusisaari and Knöpfel, 2008)	exc
DCNi-IO	6	6	3.0	20.0 (Best and Regehr, 2009)	60.0 (Best and Regehr, 2009)	inh

resonance; PC: autorhythm and bursting; DCN: autorhythm, adaptation and rebound bursting; IO: subthreshold oscillations, rebound spiking, phase reset), as optimized in Geminiani et al. (2019) (**Supplementary Material**). Only the firing irregularity parameters were modified with respect to Geminiani et al. (2019), to account during network simulations for higher noise components that are absent during *in vitro* experiments (**Supplementary Material**). As a result, we obtained physiological Coefficient of Variation of Inter-Spike Intervals (CV_{ISI}) and average firing frequency (f_{tonic}) observed *in vivo* (Ten Brinke et al., 2017; Boele et al., 2018). Specifically, PCs showed $f_{tonic} = 85$ Hz and $CV_{ISI} = 0.2$, and DCNp, $f_{tonic} = 65$ Hz and $CV_{ISI} = 0.2$.

Then, the same reconstructed circuit was populated by basic LIF neurons (LIF-SNN). The passive membrane parameters were set equal for EGLIF and LIF neurons, specific for each neuron type (**Supplementary Material**). The intrinsic current generating

spontaneous firing was tuned in the LIF neurons using trial and error, to obtain the same desired autorhythm rates.

Synaptic transmission was regulated by alpha-shaped conductance-based synapses, where reversal potentials were set to 0 mV for all excitatory synapses and -80 mV for inhibitory synapses (Cavallari et al., 2014). Multiple synapses on the same post-synaptic neuron were introduced in order to modulate the impact of different pre-synaptic populations, by using *ad hoc* synaptic parameters. The time constants of the conductance functions (τ_{α}) and the synaptic delays were defined based on scaffold values (Casali et al., 2019) and literature data (**Table 2**). Synaptic weights were set through trial and error in order to generate reference firing rates of each neural population, during baseline state of the network, i.e., without external stimuli. In setting those synaptic weights, qualitative and comparative information were taken as constraints, e.g., the robust connections from IOs to PCs through CFs, and the

stronger effect of GrCs on the post-synaptic neuron when the connection is through AAs than through PFs (Casali et al., 2019). Since the non-synaptic “spill-over” interaction between CFs and MLIs (Szapiro and Barbour, 2007; Jörntell et al., 2010), delay values of CF-MLI connections were set not all equal, but randomly chosen within a normal distribution to represent the slow and gradual neurotransmitter release. Short 1 ms delays (corresponding to the simulation resolution) were used in the interneuron inhibitory subnetworks (GoCs-GoCs and MLIs-MLIs) to mimic gap junctions (Hahne et al., 2015). The same synaptic delays and weights were used in both EGLIF-SNN and LIF-SNN, to ensure that the response differences between the two models could be ascribed unequivocally to different single neuron dynamics.

Network Stimulation Protocol and Data Analysis

The reconstructed olivocerebellar network with optimized cell-specific neuron models (Geminiani et al., 2019) was then simulated in PyNEST (Diesmann and Gewaltig, 2002; Eppler et al., 2009). The emergent spatio-temporal dynamics was analyzed, such as the responses of all neuron populations to sensory signals involving different input pathways. To understand the impact of single neuron dynamics in emerging properties at network and signal encoding level, the same simulation protocols were applied in the two network models, EGLIF-SNN and LIF-SNN.

The chosen input signals mimic those used in EyeBlink classical conditioning (EBCC), a well-known cerebellum-driven task, commonly used to investigate cerebellar learning and the underlying circuit mechanisms (Jirenhed et al., 2007). Recruiting different sensory pathways, the input signals during EBCC are usually a continuous light signal (a LED) and a time-locked short air puff stimulation on the eye. On the other hand, the motor response is an eye closure. Our model focused on the beginning of this task, when timing associative learning has not occurred yet, and only the second stimulus is supposed to generate an attention-triggered motor response. Within our SNN, the light stimulus was encoded as a 40 Hz Poisson process conveyed through a wide MF bundle investing both microcomplexes. Moreover, transversal PF projections from the Granular layer and MF collaterals to DCN cells allow the signals to travel across adjacent microcomplexes (Kalmbach et al., 2010). The air puff was a 500 Hz burst conveyed to CFs belonging specifically to one microcomplex (Ten Brinke et al., 2015, 2017). The output motor response was decoded from the net spiking activity of DCNp neurons.

The network testing protocol included a first 1-s baseline phase with a 4 Hz Poisson process to MFs. This baseline input simulated the typical *in vivo* background noise (Rancz et al., 2007). Afterward, a 40-Hz MF spike train (associated to LED light) started, lasting 260 ms. It co-terminated with the 500-Hz CF burst (associated to air puff) which lasted 10 ms. A final 500-ms phase was added after this stimulation pair, to evaluate the capability of the network to return to baseline rest condition (Ten Brinke et al., 2015, 2017).

The input spike train activated a MF bundle in the scaffold network, specifically a cylinder with a basis radius of 150 μm at the center of the transversal x - z plane, and a height of 150 μm thus including the whole granular layer thickness. This activation pattern was chosen based on the experimental observation that cerebellar activation is region specific and topographically organized, with MFs activating in bundles eliciting local responses (Morissette and Bower, 1996; Diwakar et al., 2011). In addition, this pattern allowed to avoid edge effects due to truncated connectivity close to the borders. As a result, about 80% of glomeruli received the afferent input.

To avoid unnatural synchronization of populations' initial spikes, the membrane potential of each neuron was initialized to a random value between the population-specific resting potential and threshold potential, in both EGLIF-SNN and LIF-SNN.

Raster plots of example neurons were used to visualize single neuron responses, while the network activity was represented as PeriStimulus time histograms (PSTH) with time bin = 5 ms, for each neural population at rest and during the imposed stimulation patterns.

PC and DCNp populations represented the convergence stages of both input stimuli pathways. Therefore, the instantaneous firing rates of PC and DCNp neurons in the first microcomplex (the one receiving both MF input and CF burst) were computed as the convolution between the neuron spiking patterns and a gaussian sliding window of 5 ms and 10 ms, respectively (Dayan and Abbott, 2001). To evaluate the difference in the responses between EGLIF-SNN and LIF-SNN, for each PC and DCNp neuron, we measured the activity change – response *speed* – following the second stimulus (i.e., CF burst):

$$speed_i = \frac{max_rate_i - min_rate_i}{\Delta t} \quad \text{for each neuron } i,$$

being max_rate_i and min_rate_i , the maximum and minimum firing rate of the i -th neuron within the 100-ms interval starting 5 ms after the CF burst onset, and Δt the time interval between them.

Finally, the resulting motor response was computed from DCNp activity: the spiking pattern of each microcomplex was first decoded using an update and decay rule (update constant: 1.0; decay time constant: 10 ms) and then filtered with a moving average filter using a 50-sample window. The final eyeblink response was computed from the net decoded activity of both microcomplexes.

RESULTS

The olivocerebellar SNN was organized into two cortical *microzones*, distinguished by their connections from CFs while sharing information from the granular layer (Voogd and Glickstein, 1998). The two microzones, differentially connected to DCN and IO, formed two distinct *microcomplexes* (Ito, 1984; **Figure 1**). The olivocerebellar SNN was able to encode different inputs into output spike patterns. We have analyzed in detail the

response to spike trains imitating EBCC-like sensory inputs. The comparison between the EGLIF-SNN and LIF-SNN allowed to identify the contribution of non-linear single neuron properties to ensemble network dynamics.

The basal activity of cerebellar neurons and their response to MF and CF inputs is illustrated in **Figures 2–7**. In both EGLIF-SNN and LIF-SNN models, during baseline MF activation with random noise at 4 Hz (Rancz et al., 2007), the GrCs were driven into low frequency firing, and the GoC, MLI, PC and DCN neurons slightly increased their firing rate compared to intrinsic pacemaking (Geminiani et al., 2019).

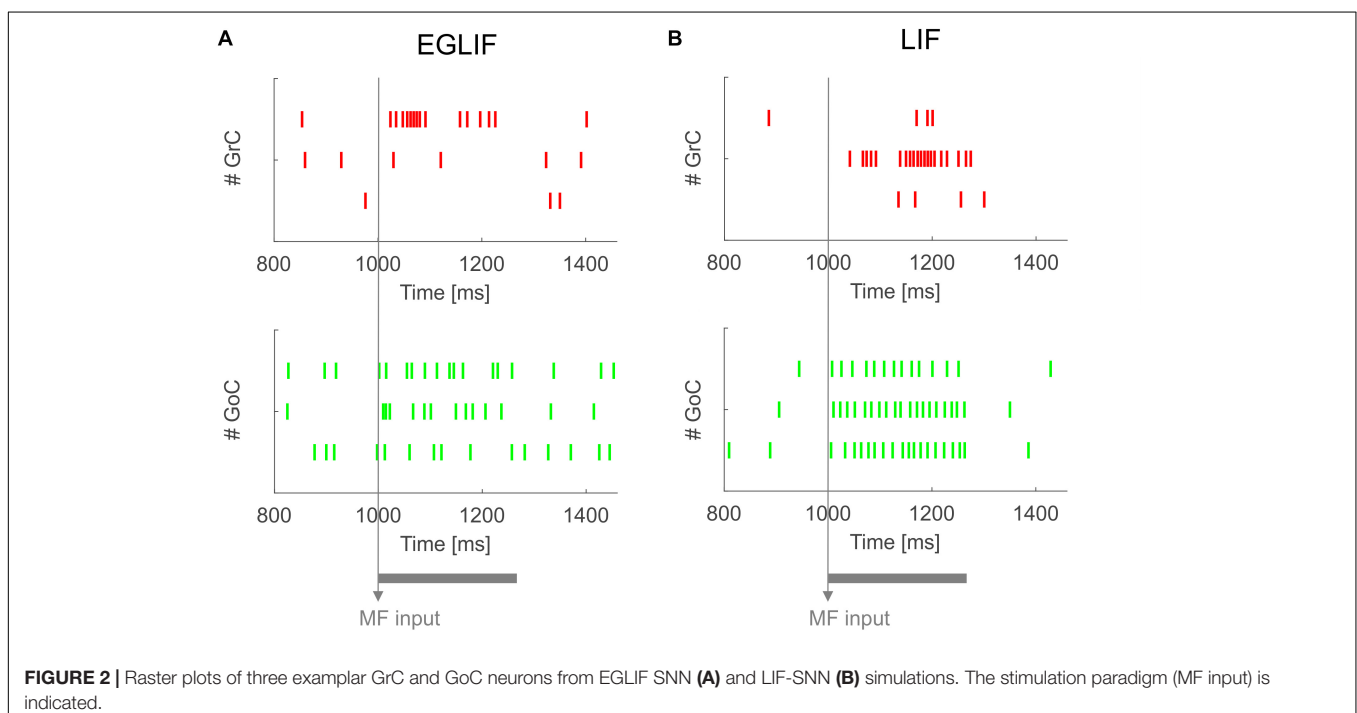
The activity of EGLIF-SNN and LIF-SNN changed during stimulation of the MFs (260 ms at 40 Hz on a MFs bundle, see section “Materials and Methods”) and when a burst was generated in CFs coming from the IO (10 ms at 500 Hz on one microcomplex, see section “Materials and Methods”). At the onset of stimulation, when only MFs were active, the firing rates for all neural populations of the cortical microzones increased with average frequency values within the physiological range. In particular, an increase of about 10 Hz in PC firing rate with respect to 85 Hz baseline emerged, consistent with experimental observations showing that PC activity is largely sustained by pacemaking (Cerminara and Rawson, 2004). The responses of DCN neurons demonstrated a reduction in DCNi, which received only inhibition from PCs, and almost no change in DCNp, which received balanced excitation from MFs and inhibition from PCs, revealing the regulatory power of the system on the cerebellar output. On the other hand, when also the CF burst was injected, complex dynamic spiking patterns were elicited, differentiated in the two microcomplexes; and here the superiority of EGLIF-SNN with respect to LIF-SNN to simulate non-linear responses emerged.

Granular Layer

Both in EGLIF-SNN and LIF-SNN, the GrCs showed a background low-frequency sparse activation that increased and then recovered to baseline without apparent rebounds. The GoCs also increased firing frequency during the MF stimulus, and then showed a rapid reduction at its end lasting about 30 ms. This was due to slow recovery of the pacemaker cycle reflecting a phase-reset mechanism (Solinas et al., 2007; Geminiani et al., 2018b). The GrCs did not show a corresponding remarkable rebound in their firing rate, probably because of the prolonged effect of GoC-GrC synaptic inhibition, which lasts for about 50 ms (Bengtsson et al., 2013).

Molecular Layer, PC, and DCN – Microcomplex 1

The activation of IO neurons connected to microcomplex 1 caused a characteristic spiking pattern. In the EGLIF-SNN, the IO input burst induced a typical response in connected PCs, consisting of synchronous complex spikes followed by a pause (*burst-pause*). Each complex spike included a first burst approximating dendritic spikelets, induced by the 10-ms IO input, and a subsequent pause/hyperpolarization, resulting from intrinsic neuron model mechanisms (De Zeeuw et al., 2011; Geminiani et al., 2019). After the burst-pause response, firing recovered but a second firing decrease occurred, caused by spillover-mediated inhibition from MLIs (about 70 ms after the IO burst onset). The PC complex spikes triggered by the IO silenced DCNp neurons (pause), which, after the hyperpolarization, generated a rebound burst. The DCNp *pause-burst* response matches neurophysiological observations (Pugh and Raman, 2006; Zheng and Raman, 2010). DCNi received



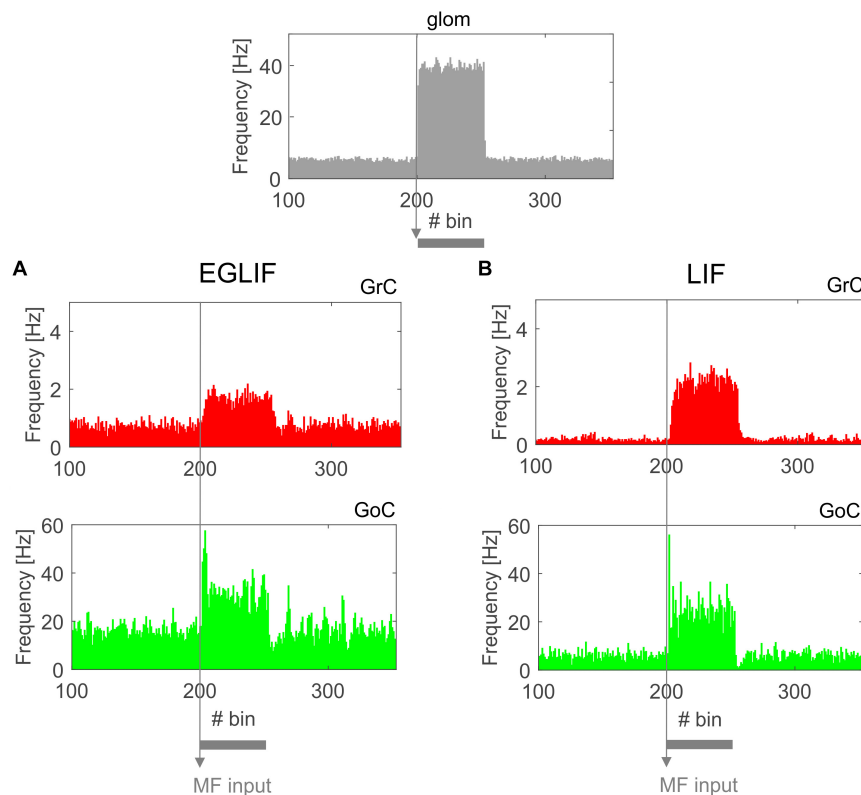


FIGURE 3 | PSTH of all Granular layer neurons in EGLIF-SNN (A) and LIF-SNN (B). In both cases, stimulation of a MF bundle common to both microcomplexes (top panel) causes the mean firing rate of granule cells and Golgi cells to increase. Note the similar patterns of neuronal activity in the two networks. The absolute values of firing rates are within physiological ranges *in vivo*. Each PSTH bin is 5 ms long.

only PC and IO inputs but not MF excitation, they generated a rebound spike after the strong inhibition from PC complex spikes. In the LIF-SNN, the burst-pause regime of PCs and pause-burst regime of DCN cells did not emerge.

Molecular Layer, PC, and DCN – Microcomplex 2

Neurons belonging to microcomplex 2 received only the MF stimulus causing a net increase of firing rates in MLI, PC and DCNp neurons, and a pause in DCNi cells not receiving MF excitation.

For PC and DCNp in the microcomplex 1, where PF and CF stimuli converged, the average firing rate response was sharper in the EGLIF-SNN (Figure 8A), impacting on the timing precision of the network output. Indeed, the dynamic modulation of spike patterns observed using EGLIF could not be reproduced with LIF network models, since the simplified dynamics of single neurons prevented from generating bursting, pause and rebound responses. Consequently, the response *speed* was significantly higher in PC and DCNp neural populations within EGLIF-SNN (PC *speed*: -23.82 ± 1.96 Hz/ms in EGLIF-SNN vs. -2.25 ± 0.91 Hz/ms in LIF-SNN, *t*-test: $p < 0.01$; DCNp *speed*: 1.72 ± 0.83 Hz/ms in EGLIF-SNN vs. 1 ± 0.06 Hz/ms in LIF-SNN, *t*-test: $p < 0.01$).

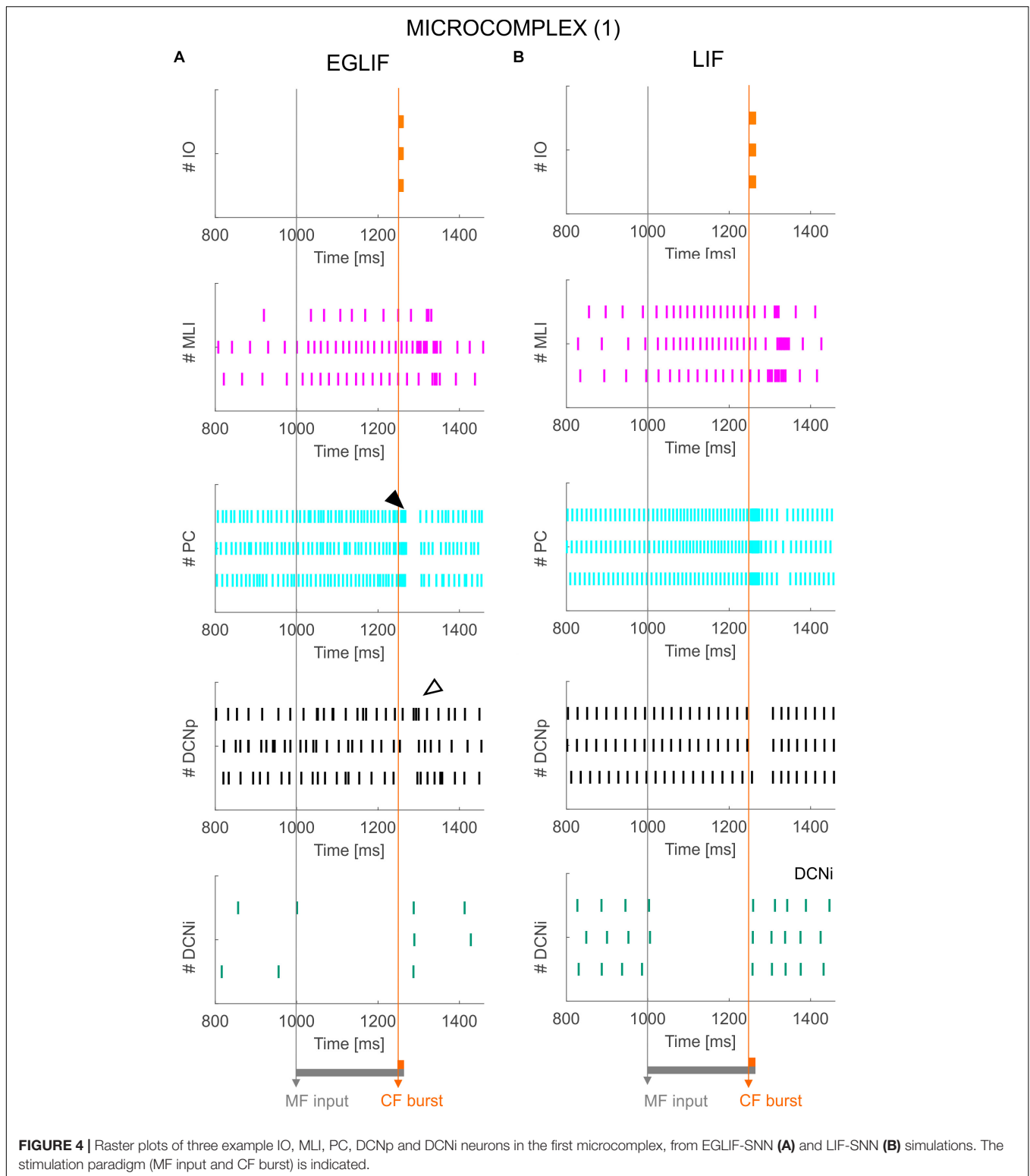
As a result, the eyeblink response computed from the net decoded activity of DCNp neurons was faster and sharper in the EGLIF-SNN simulations (Figure 8B).

DISCUSSION

The main observation in this study is that neuron models with realistic non-linear properties EGLIF (Geminiani et al., 2018b, 2019), once embedded into networks with realistic geometry and connectivity (Casali et al., 2019), have a significant impact on ensemble response dynamics compared to simpler models (LIF). The effectiveness of EGLIF emerged as a pattern of burst-pause and pause-burst responses in PC and DCNp neurons reproducing observations *in vivo* (Herzfeld et al., 2015; Moscato et al., 2019) and was most evident when the microcomplex received the CF stimuli. Since we used stimulus patterns emulating those occurring in the eye-blink reflex, it is anticipated that single neuron properties will reverberate on sensorimotor control in closed-loop.

Single Neuron Activity and SNN Responses to Stimuli

In EGLIF-SNN simulations, the integration of bursts on the CFs and spike trains on PFs proved fundamental for generating



a realistic PC output. These stimuli caused PCs to shift from spontaneous background activity to complex spikes and simple spike trains taking the form of a burst-pause response. The burst-pause was the consequence of intrinsic PC non-linear

electroresponsive dynamics engaged by patterned synaptic inputs from PFs, MLIs, and IO (Jirenhed et al., 2013). Always in EGLIF-SNN simulations, DCNp neurons showed pause-burst responses deriving from intrinsic DCNp neuron electroresponsiveness

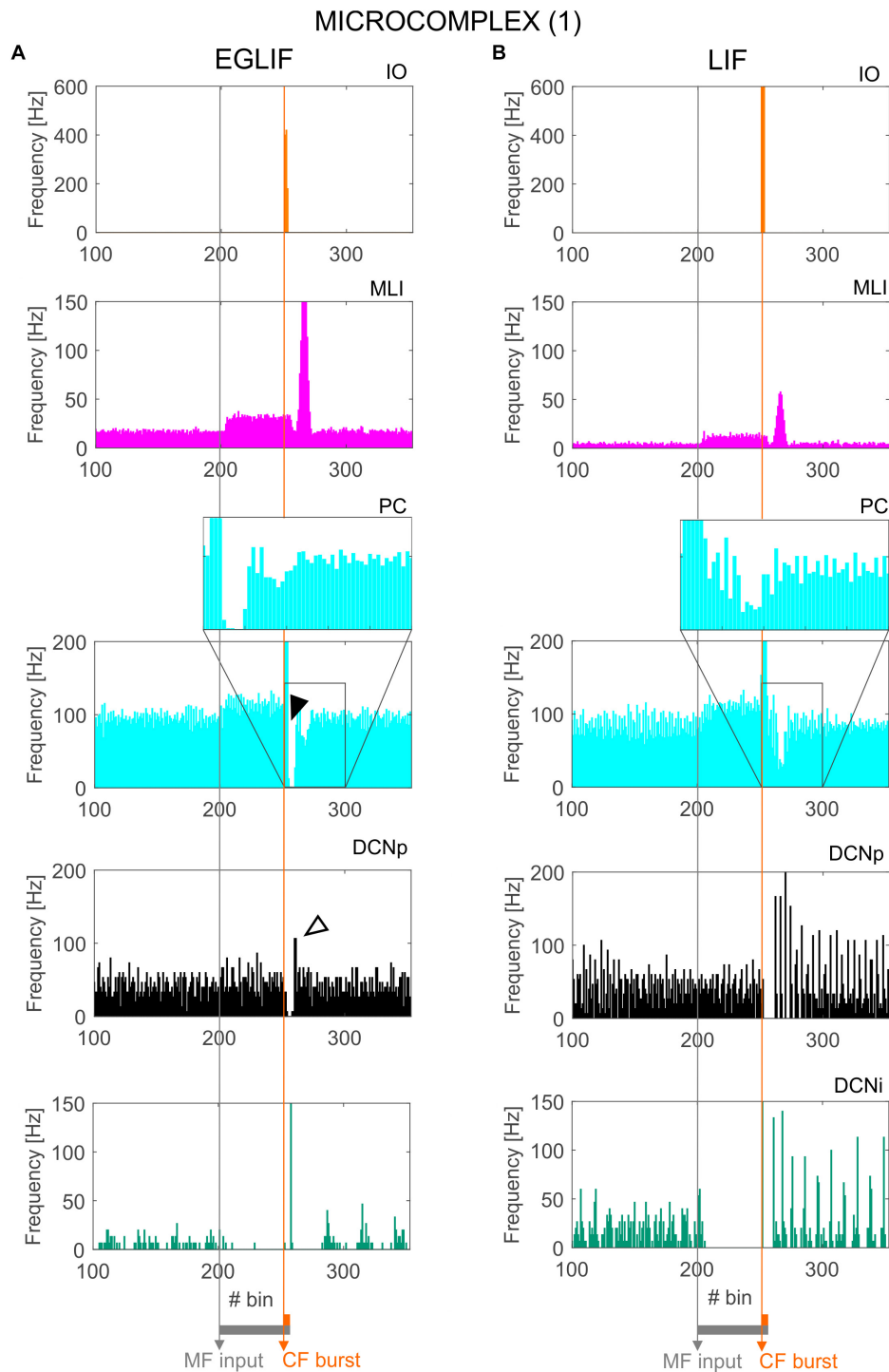
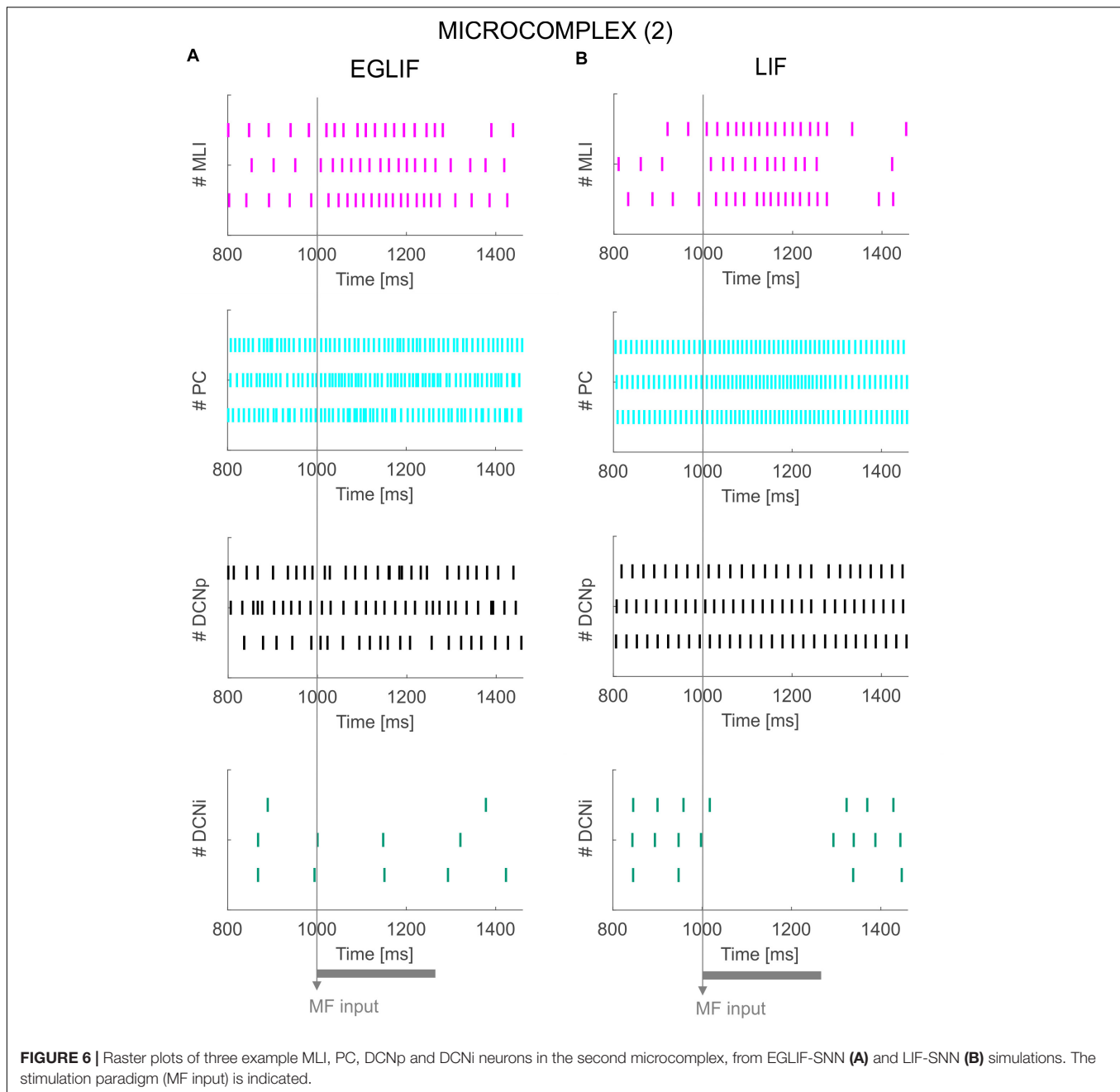


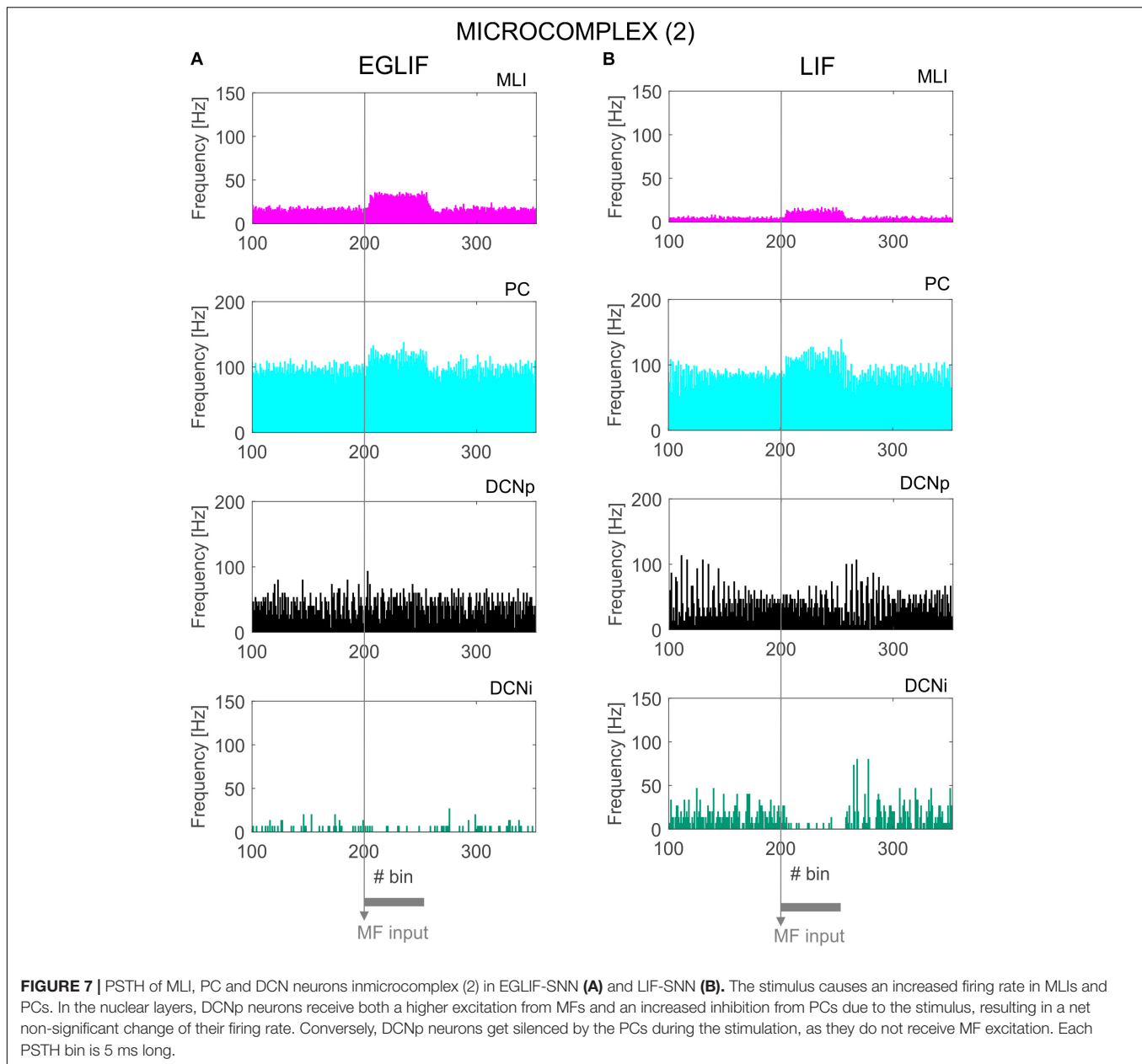
FIGURE 5 | PSTH of IO, MLI, PC and DCN neurons in microcomplex (1) in EGLIF-SNN (A) and LIF-SNN (B). The first stimulus (MF input) increases the firing rate in MLI, PC and DCNp neurons during the 260 ms interval, while DCNi cells that do not receive MF inputs, get inhibited by the increased PC firing. The air puff is encoded as a burst from CFs. MLIs receive the CF stimulus through the IO pathway causing a delayed protracted increase in firing rate about 70 ms after the stimulus, due to neurotransmitter spillover from CFs. At PC level, CF stimulation results in a complex spike (burst-pause, black arrow) causing a pause-burst in DCN neurons (white arrow). Note that these dynamic behaviors are observed only in the EGLIF-SNN due to the complex intrinsic dynamics of EGLIF neuron models. In LIF-SNN, the PC burst caused by CF input is not followed by the pause, while in DCNp neurons the pause due to PC complex spike inhibition is followed by a synchronous restart of firing (causing the increased instantaneous frequency) without any rebound burst. Note that the lower irregularity of firing in LIF-SNN simulations resulted in apparent higher firing rates, due to non-physiological synchronization of population spikes. Each PSTH bin is 5 ms long.



engaged by synaptic inputs from PCs, MF and CF collaterals (Herzfeld et al., 2015; Moscato et al., 2019). Indeed, these spiking patterns proved to have a crucial impact on response speed and time precision (**Figure 8**) providing a potential advantage for cerebellum-driven tasks, in which the cerebellum acts as a millisecond-precise controller (Bareš et al., 2019; Heck et al., 2013). The intrinsic bursting properties of the EGLIF model, already proved in simulations of single neuron responses to current steps (Geminiani et al., 2019), here proved fundamental to capture emergent network dynamics. It should be noted that, in LIF-SNN simulations, burst-pause and pause-burst responses did not emerge. These results therefore support the

adequacy of EGLIF neurons for realistic simulations of cerebellar SNNs in closed-loop.

The impact of EGLIF neurons on oscillatory network dynamics, that are expected to emerge from feedback circuit loops in the granular layer (D'Angelo et al., 2013; Maex and De Schutter, 2013), remains to be investigated. Indeed, the intrinsic membrane potential oscillations of EGLIF in single neuron stimulation protocols could impact on network oscillations, and should be further investigated (Geminiani et al., 2019). An open question is also how the EGLIF representation compromises with non-linear dendritic processing in PCs, in which the excitatory post-synaptic potentials are locally amplified

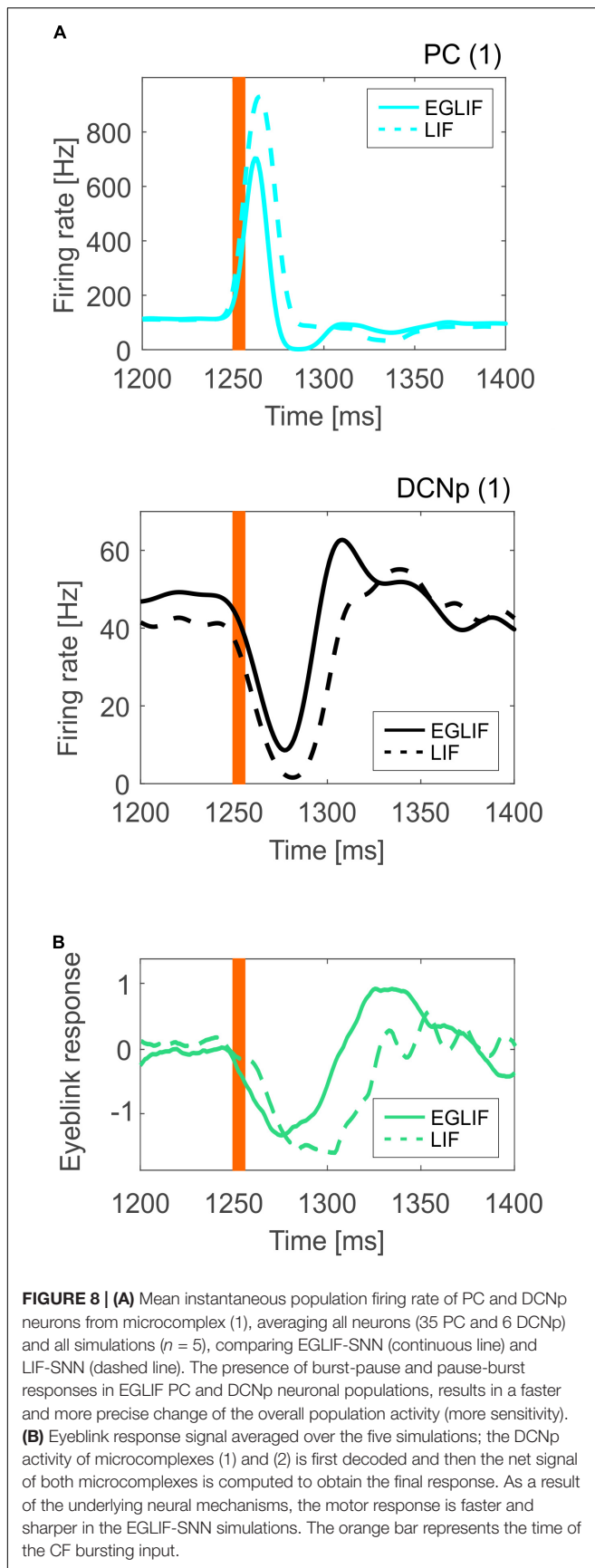


by Calcium spikes and integrated into complex spatio-temporal sequences (Masoli et al., 2015; Masoli and D'Angelo, 2017). A similar case applies to DCN cells too, in which the inhibitory post-synaptic potentials set up non-linear interactions with low-threshold calcium spikes (Si Feng et al., 2013). These aspects need to be further investigated by comparison with detailed multicompartmental neuron models.

Neuronal Wiring and Synaptic Transmission in the SNN

The importance of geometry and connectivity was recently addressed using LIF neurons in a scaffold cerebellar network (Casali et al., 2019). Here the network has been upgraded with

EGLIF neurons and extended to include the IO-DCN sub-circuit to form two different microcomplexes, demonstrating additional network properties. In the current configuration, as said, the network generated spiking patterns similar to those observed *in vivo*. A critical issue in this context is the definition of synaptic models (Cavallari et al., 2014). Here we have chosen conductance-based synaptic models implemented with alpha functions, which accounted in an accurate yet simplified form for neurotransmission kinetics (Table 2). A future improvement could be to define conductance changes using specific NMDA, AMPA and GABA kinetics in each neuron type [e.g., see (Wu and Raman, 2017)]. In addition, the more precise spiking patterns of the EGLIF-SNN make this network a better candidate also to investigate short-term plasticity mechanisms.



For example, it could be possible to evaluate whether short-term facilitation can further enhance the time precision of the response, amplifying bursting mechanism. In addition, EGLIF-SNN simulations with short-term plasticity could allow to clarify how single neuron and synaptic dynamics interact to generate proper network dynamics.

Finally, phenomena like neurotransmitter spillover and electrical transmission through gap-junctions were approximated here by tuning delay parameters, but could be better reproduced by customized models (Latorre et al., 2013). In GoC and IO neuronal populations, more realistic gap junctions would allow, for instance, to investigate more in detail circuit oscillation properties (Leznik and Llinás, 2005).

Implications for Eyeblink Conditioning and Other Cerebellum-Driven Paradigms

The stimulation patterns used here mimicked the typical input signals that are used in EBCC tasks including a prolonged and spatially distributed sensory stimulus (CS, light) and a short attentional signal [Unconditioned Stimulus (US), air puff]. The current study focused on the response before learning: CS excited the granular layer across microzones, consistent with the operation of signal analysis (through recombinatorial expansion) carried out by the granular layer (D'Angelo et al., 2013; Gilmer and Person, 2017). The granular layer output was then synthesized and further processed in the PC layer (Dean and Porrill, 2011). US influenced individual microcomplexes through specific IO projections, segregating the attention (or error) signal within the network. These modular activation patterns represent the most elementary instantiation of cerebellar functioning, i.e., the ability to correlate neural signals transmitted along different afferent pathways, the MFs and CFs. These signals, in a behavioral context, are needed to allow the cerebellum *to learn to predict the precise timing of correlated events*, setting the basis for cerebellar contribution to motor and cognitive control (Ivry, 2000; D'Angelo and Casali, 2013). It seems therefore highly relevant that the emerging burst-pause and pause-burst responses in PC and DCNp neurons are precisely reproduced using EGLIF-SNN. These activity patterns will be critical for generating the proper time-locked response in future simulations of EBCC (Rasmussen et al., 2008). This will require to endow the current SNN model with distributed long-term plasticity to simulate learning mechanisms (Antonietti et al., 2016). While the current work evaluated the impact of non-linear single neuron dynamics and network topology on stimulus-response spiking patterns, closed-loop simulations of a full cerebellum-driven learning task with the EGLIF-SNN will allow to evaluate the impact of long-term plasticity, mainly spike-timing dependent plasticity mechanisms, driven by IO and PC spikes.

As a result of modularity and specific connectivity to various brain regions, different cerebellar modules are engaged in different tasks (D'Angelo and Casali, 2013). The modules receive various kinds of input signals, which carry information about specific sensory modalities or specific body parts as well as about activity in motor and associative cortical areas. The modules can differ not only in terms of sources and pathways of the incoming

signals, but also in terms of specific electroresponsive properties of neurons. For example, differences in the autorhythm of PCs were observed between regions involved in EBCC and vestibulo-ocular reflexes (Zhou et al., 2014). Similarly, a modulation of oscillatory properties emerge in the IO neural population when encoding either somatosensory or visual stimuli (Llinás, 2014). The possibility to easily modify neuron models and connectivity in our olivocerebellar EGLIF-SNN would allow to fine tune specific features associated to sensorimotor loops and functional cerebellar regions (Casellato et al., 2014; Geminiani et al., 2017; Luque et al., 2019).

According to the modular organization of the cerebellum, these microcomplexes could be multiplied and reconnected to investigate how input signals are integrated and elaborated to control complex movements, for example in whisking and locomotion (Romano et al., 2018). Scaling-up the network modular architecture would require to re-organize connectivity among microcomplexes, which can determine fundamental properties of cerebellar functioning, such as somatotopic organization, fractured somatotopy mapping and multimodal sensory fusion.

CONCLUSION

Since the model satisfactorily captures fundamental properties of microcomplexes, it can help shedding light on the links between structure, function and dynamics in the cerebellum under physiological and pathological conditions and during learning (D'Angelo and Gandini Wheeler-Kingshott, 2017). These extended applications are warranted by the flexible structure of the scaffold (Casali et al., 2019) and the tunable nature of EGLIF neurons (Geminiani et al., 2018b, 2019). For example, in different species or in pathological conditions, EGLIF-SNN could account for variations in the number of neurons as well as in their connectivity and intrinsic electroresponsiveness, while maintaining high efficiency when running large-scale simulations in closed-loop. Future work will endow the EGLIF-SNN cerebellum models with mechanisms for synaptic plasticity in order to evaluate the impact of single neuron and network properties on motor learning (Hansel et al., 2001; Schonewille et al., 2010; Gao et al., 2012; D'Angelo, 2014; Boele et al., 2018). Eventually, the model may be exploited to

mimic pathological conditions at multiple scales (Geminiani et al., 2018a) providing new insights into the role of cerebellum in brain diseases (D'Angelo and Casali, 2013; D'Angelo, 2019; Schmahmann, 2019). It is also envisaged that the EGLIF scaffold strategy could be customized to model and simulate other brain regions (like the cerebral cortex, hippocampus or basal ganglia).

DATA AVAILABILITY STATEMENT

All datasets generated for this study are included in the manuscript/**Supplementary Files**.

AUTHOR CONTRIBUTIONS

AG and CC designed and carried out the simulations, performed data analysis and wrote the manuscript. AP, ED'A and CC coordinated the whole work and substantially contributed to the writing of the final manuscript.

FUNDING

This work has been developed within the *CerebNEST* HBP Partnering Project and has received funding from the European Union's Horizon 2020 Framework Programme for Research and Innovation under Grant Agreement No. 785907 (Human Brain Project SGA2). Supercomputing resources were provided by the Italian Supercomputing Center CINECA, within the ISCRA C(IsC67) project HP10CK7QLR (NEST-EBC).

ACKNOWLEDGMENTS

We thank Cristiano Padrin from CINECA for his technical support.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fncom.2019.00068/full#supplementary-material>

REFERENCES

- Antonietti, A., Casellato, C., Angelo, E. D., and Pedrocchi, A. (2016). Model-driven analysis of eyeblink classical conditioning reveals the underlying structure of cerebellar plasticity and neuronal activity. *IEEE Trans. Neural Networks Learn. Syst.* 28, 2748–2762. doi: 10.1109/TNNLS.2016.2598190
- Bareš, M., Apps, R., Avanzino, L., Breska, A., D'Angelo, E., Filip, P., et al. (2019). Consensus paper: decoding the contributions of the cerebellum as a time machine. From neurons to clinical applications. *Cerebellum* 18, 266–286. doi: 10.1007/s12311-018-0979-5
- Bengtsson, F., Geborek, P., and Jörntell, H. (2013). Cross-correlations between pairs of neurons in cerebellar cortex in vivo. *Neural Netw.* 47, 88–94. doi: 10.1016/j.neunet.2012.11.016
- Best, A. R., and Regehr, W. G. (2009). Inhibitory regulation of electrically coupled neurons in the inferior olive is mediated by asynchronous release of GABA. *Neuron* 62, 555–565. doi: 10.1016/j.neuron.2009.04.018
- Boele, H., Peter, S., Brinke, M. M., Ten Verdonck, L., Ijpelaar, A. C. H., Rizopoulos, D., et al. (2018). Impact of parallel fiber to Purkinje cell long-term depression is unmasked in absence of inhibitory input. *Sci. Adv.* 4, 1–8. doi: 10.1126/sciadv.aas9426
- Casali, S., Marenzi, E., Medini, C., Casellato, C., and D'Angelo, E. (2019). Reconstruction and simulation of a scaffold model of the cerebellar network. *Front. Neuroinform.* 13:37. doi: 10.3389/fninf.2019.00037
- Casellato, C., Antonietti, A., Garrido, J. A., Carrillo, R. R., Luque, N. R., Ros, E., et al. (2014). Adaptive robotic control driven by a versatile spiking cerebellar network. *PLoS One* 9:e0118518. doi: 10.1371/journal.pone.0112265

- Cavallari, S., Panzeri, S., and Mazzoni, A. (2014). Comparison of the dynamics of neural interactions between current-based and conductance-based integrate-and-fire recurrent networks. *Front. Neural Circuits* 8:12. doi: 10.3389/fncir.2014.00012
- Cerminara, N. L., and Rawson, J. A. (2004). Evidence that climbing fibers control an intrinsic spike generator in cerebellar Purkinje cells. *J. Neurosci.* 24, 4510–4517. doi: 10.1523/JNEUROSCI.4530-03.2004
- D'Angelo, E. (2014). The organization of plasticity in the cerebellar cortex: from synapses to control. *Prog. Brain Res.* 210, 31–58. doi: 10.1016/B978-0-444-63356-9.00002-9
- D'Angelo, E. (2018). "Physiology of the cerebellum," in *The Cerebellum: From Embryology to Diagnostic Investigations*, ed. A. Plaitakis, (Boston, MA: Springer), 85–108. doi: 10.1016/B978-0-444-63956-1.00006-0
- D'Angelo, E. (2019). The cerebellum gets social. *Science*. 363:229. doi: 10.1093/cercor/bhy322
- D'Angelo, E., Antonietti, A., Casali, S., Casellato, C., Garrido, J. A., Luque, N. R., et al. (2016). Modeling the cerebellar microcircuit: new strategies for a long-standing issue. *Front. Cell. Neurosci.* 10:176. doi: 10.3389/fncel.2016.00176
- D'Angelo, E., and Casali, S. (2013). Seeking a unified framework for cerebellar function and dysfunction: from circuit operations to cognition. *Front. Neural Circuits* 6:116. doi: 10.3389/fncir.2012.00116
- D'Angelo, E., and Gandini Wheeler-Kingshott, C. (2017). Modelling the brain: elementary components to explain ensemble functions. *Riv. del nuovo Cim.* 40, 273–333. doi: 10.1393/ncr/i2017-10137-5
- D'Angelo, E., Solinas, S., Mapelli, J., Gandolfi, D., Mapelli, L., and Prestori, F. (2013). The cerebellar Golgi cell and spatiotemporal organization of granular layer activity. *Front. Neural Circuits* 7:93. doi: 10.3389/fncir.2013.00093
- Davie, J. T., Clark, B. A., and Häusser, M. (2008). The origin of the complex spike in cerebellar Purkinje cells. *J. Neurosci.* 28, 7599–7609. doi: 10.1523/JNEUROSCI.0559-08.2008
- Dayan, P., and Abbott, L. F. (2001). *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. Cambridge, MA: Massachusetts Institute of Technology Press.
- De Zeeuw, C. I., Hoebeek, F. E., Bosman, L. W. J., Schonewille, M., Witter, L., and Koekkoek, S. K. (2011). Spatiotemporal firing patterns in the cerebellum. *Nat. Rev. Neurosci.* 12, 327–344. doi: 10.1038/nrn3011
- Dean, P., and Porrill, J. (2011). Evaluating the adaptive-filter model of the cerebellum. *J. Physiol.* 589, 3459–3470. doi: 10.1113/jphysiol.2010.201574
- Diesmann, M., and Gewaltig, M.-O. (2002). "NEST: an environment for neural systems simulations," in *Forsch. und Wissenschaftliches Rechn. Beiträge zum Heinz-billing-pr. 2001, GWDG-Bericht*, eds T. Plesser, and V. Macho, (Göttingen: Ges. für Wiss. Datenverarbeitung), 43–70.
- Diwakar, S., Lombardo, P., Solinas, S., Naldi, G., and D'Angelo, E. (2011). Local field potential modeling predicts dense activation in cerebellar granule cells clusters under LTP and LTD Control. *PLoS One* 6:e21928. doi: 10.1371/journal.pone.0021928
- Eppler, J. M., Helias, M., Muller, E., Diesmann, M., Gewaltig, M., and Stewart, T. C. (2009). PyNEST?: a convenient interface to the NEST simulator. *Front. Neuroinform.* 2:12. doi: 10.3389/neuro.11.012.2008
- Gandolfi, D., Lombardo, P., Mapelli, J., Solinas, S., and D'Angelo, E. (2013). Theta-frequency resonance at the cerebellum input stage improves spike timing on the millisecond time-scale. *Front. Neural Circuits* 7:64. doi: 10.3389/fncir.2013.00064
- Gao, Z., van Beugen, B. J., and De Zeeuw, C. I. (2012). Distributed synergistic plasticity and cerebellar learning. *Nat. Rev. Neurosci.* 13, 619–635. doi: 10.1038/nrn3312
- Geminiani, A., Casellato, C., Antonietti, A., D'Angelo, E., and Pedrocchi, A. (2018a). A multiple-plasticity spiking neural network embedded in a closed-loop control system to model cerebellar pathologies. *Int. J. Neural Syst.* 28:1750017. doi: 10.1142/S0129065717500174
- Geminiani, A., Casellato, C., Locatelli, F., Prestori, F., Pedrocchi, A., and D'Angelo, E. (2018b). Complex dynamics in simplified neuronal models: reproducing Golgi cell electroresponsiveness. *Front. Neuroinform.* 12:88. doi: 10.3389/fninf.2018.00088
- Geminiani, A., Casellato, C., D'Angelo, E., and Pedrocchi, A. (2019). Complex electroresponsive dynamics in olivocerebellar neurons represented with extended-generalized leaky integrate and fire models. *Front. Comput. Neurosci.* 13:35. doi: 10.3389/FNCOM.2019.00035
- Geminiani, A., Pedrocchi, A., D'Angelo, E., and Casellato, C. (2017). "Spiking cerebellar model with damaged cortical neural population reproduces human ataxic behaviors in perturbed upper limb reaching," in *Proceedings of the 2017 8th International IEEE/EMBS Conference on Neural Engineering (NER)*, (Shanghai), doi: 10.1109/NER.2017.8008396
- Gilmer, J. I., and Person, A. L. (2017). Morphological constraints on cerebellar granule cell combinatorial diversity. *J. Neurosci.* 37, 12153–12166. doi: 10.1523/JNEUROSCI.0588-17.2017
- Hahne, J., Helias, M., Kunkel, S., Igarashi, J., Bolten, M., Frommer, A., et al. (2015). A unified framework for spiking and gap-junction interactions in distributed neuronal network simulations. *Front. Neuroinform.* 9:22. doi: 10.3389/fninf.2015.00022
- Hansel, C., Linden, D. J., and D'Angelo, E. (2001). Beyond parallel fiber LTD: the diversity of synaptic and non-synaptic plasticity in the cerebellum. *Nat. Neurosci.* 4, 467–475. doi: 10.1038/87419
- He, Q., Duguid, I., Clark, B., Panzanelli, P., Patel, B., Thomas, P., et al. (2015). Interneuron- and GABAA receptor-specific inhibitory synaptic plasticity in cerebellar Purkinje cells. *Nat. Commun.* 6, 1–13. doi: 10.1038/ncomms8364
- Heck, D. H., De Zeeuw, C. I., Jaeger, D., Khodakhah, K., and Person, A. L. (2013). The neuronal code(s) of the cerebellum. *J. Neurosci.* 33, 17603–17609. doi: 10.1523/JNEUROSCI.2759-13.2013
- Heiney, S. A., Kim, J., Augustine, G. J., and Medina, J. F. (2014). Precise control of movement kinematics by optogenetic inhibition of Purkinje Cell activity. *J. Neurosci.* 34, 2321–2330. doi: 10.1523/JNEUROSCI.4547-13.2014
- Herzfeld, D. J., Kojima, Y., Soetedjo, R., and Shadmehr, R. (2015). Encoding of action by the Purkinje cells of the cerebellum. *Nature* 526, 439–442. doi: 10.1038/nature15693
- Hoebeek, F. E., Witter, L., Ruigrok, T. J. H., and De Zeeuw, C. I. (2010). Differential olivo-cerebellar cortical control of rebound activity in the cerebellar nuclei. *Proc. Natl. Acad. Sci. U.S.A.* 107, 8410–8415. doi: 10.1073/PNAS.0907118107
- Ito, M. (1984). The modifiable neuronal network of the cerebellum. *Jpn. J. Physiol.* 34, 781–792. doi: 10.2170/jjphysiol.34.781
- Ivry, R. (2000). Exploring the role of the cerebellum in sensory anticipation and timing: commentary on Tesche and Karhu. *Hum. Brain Mapp.* 9, 115–118. doi: 10.1002/(sici)1097-0193(200003)9:3<115::aid-hbm1>3.0.co;2-5
- Jirenhed, D.-A., Bengtsson, F., and Hessel, G. (2007). Acquisition, extinction, and reacquisition of a cerebellar cortical memory trace. *J. Neurosci.* 27, 2493–2502. doi: 10.1523/JNEUROSCI.4202-06.2007
- Jirenhed, D.-A., Bengtsson, F., and Jörntell, H. (2013). Parallel fiber and climbing fiber responses in rat cerebellar cortical neurons in vivo. *Front. Syst. Neurosci.* 7:16. doi: 10.3389/fnsys.2013.00016
- Jörntell, H., Bengtsson, F., Schonewille, M., and De Zeeuw, C. I. (2010). Cerebellar molecular layer interneurons - computational properties and roles in learning. *Trends Neurosci.* 33, 524–535. doi: 10.1016/j.tins.2010.08.004
- Kalmbach, B. E., Ohyama, T., and Mauk, M. D. (2010). Temporal patterns of inputs to cerebellum necessary and sufficient for trace eyelid conditioning. *J. Neurophysiol.* 104, 627–640. doi: 10.1152/JN.00169.2010
- Kanichay, R. T., and Silver, R. A. (2008). Synaptic and cellular properties of the feedforward inhibitory circuit within the input layer of the cerebellar cortex. *J. Neurosci.* 28, 8955–8967. doi: 10.1523/JNEUROSCI.5469-07.2008
- Latorre, R., Aguirre, C., Rabinovich, M. I., and Varona, P. (2013). Transient dynamics and rhythm coordination of inferior olive spatio-temporal patterns. *Front. Neural Circuits* 7:138. doi: 10.3389/fncir.2013.00138
- Lein, E. S., Hawrylycz, M. J., Ao, N., Ayres, M., Bensinger, A., Bernard, A., et al. (2007). Genome-wide atlas of gene expression in the adult mouse brain. *Nature* 445, 168–176. doi: 10.1038/nature05453
- Leznik, E., and Llinás, R. (2005). Role of gap junctions in synchronized neuronal oscillations in the inferior olive. *J. Neurophysiol.* 94, 2447–2456. doi: 10.1152/jn.00353.2005
- Llinás, R., and Negrello, M. (2011). Cerebellum. *Scholarpedia* 10:4606.
- Llinás, R. R. (2014). The olivo-cerebellar system: a key to understanding the functional significance of intrinsic oscillatory brain properties. *Front. Neural Circuits* 7:96. doi: 10.3389/fncir.2013.00096
- Luque, N. R., Naveiros, F., Carrillo, R. R., Ros, E., and Arleo, A. (2019). Spike burst-pause dynamics of Purkinje cells regulate sensorimotor adaptation. *PLoS Comput. Biol.* 15:e1006298. doi: 10.1371/journal.pcbi.1006298

- Maex, R., and De Schutter, E. (2013). Synchronization of golgi and granule cell firing in a detailed network model of the cerebellar granule cell layer. *J. Neurophysiol.* 80, 2521–2537. doi: 10.1152/jn.1998.80.5.2521
- Mapelli, L., Rossi, P., Nieuws, T., and D'Angelo, E. (2009). Tonic activation of GABA B receptors reduces release probability at inhibitory connections in the cerebellar glomerulus. *J. Neurophysiol.* 101, 3089–3099. doi: 10.1152/jn.91190.2008
- Marr, D. (1969). A theory of cerebellar cortex. *J. Physiol.* 202, 437–470. doi: 10.1113/jphysiol.1969.sp008820
- Masoli, S., and D'Angelo, E. (2017). Synaptic activation of a detailed Purkinje cell model predicts voltage-dependent control of burst-pause responses in active dendrites. *Front. Cell. Neurosci.* 11:278. doi: 10.3389/fncel.2017.00278
- Masoli, S., Solinas, S., and Angelo, E. D. (2015). Action potential processing in a detailed Purkinje cell model reveals a critical role for axonal compartmentalization. *Front. Cell. Neurosci.* 9:1–22. doi: 10.3389/fncel.2015.00047
- Miyazaki, T., Hashimoto, K., Shin, H.-S., Kano, M., and Watanabe, M. (2004). P/Q-type Ca²⁺ channel $\alpha 1A$ regulates synaptic competition on developing cerebellar Purkinje cells. *J. Neurosci.* 24, 1734–1743. doi: 10.1523/JNEUROSCI.4208-03.2004
- Morissette, J., and Bower, J. (1996). Contribution of somatosensory cortex to responses in the rat cerebellar granule cell layer following peripheral tactile stimulation. *Exp. Brain Res.* 109, 240–250. doi: 10.1007/BF00231784
- Moscato, L., Montagna, I., De Propriis, L., Tritto, S., Mapelli, L., and D'Angelo, E. (2019). Long-lasting response changes in deep cerebellar nuclei in vivo correlate with low-frequency oscillations. *Front. Cell. Neurosci.* 13:84. doi: 10.3389/fncel.2019.00084
- Peter, S., ten Brinke, M. M., Stedehouder, J., Reinelt, C. M., Wu, B., Zhou, H., et al. (2016). Dysfunctional cerebellar Purkinje cells contribute to autism-like behaviour in Shank2- deficient mice. *Nat. Commun.* 7:12627. doi: 10.1038/ncomms12627
- Powell, K., Mathy, A., Duguid, I., and Häusser, M. (2015). Synaptic representation of locomotion in single cerebellar granule cells. *eLife* 4:e07290. doi: 10.7554/eLife.07290
- Prestori, F., Rossi, P., Bearzatto, B., Lainé, J., Necchi, D., Diwakar, S., et al. (2008). Altered neuron excitability and synaptic plasticity in the cerebellar granular layer of juvenile prion protein knock-out mice with impaired motor control. *Neurobiol. Dis.* 28, 7091–7103. doi: 10.1523/JNEUROSCI.0409-08.2008
- Pugh, J. R., and Raman, I. M. (2006). Potentiation of mossy fiber EPSCs in the cerebellar nuclei by NMDA receptor activation followed by postinhibitory rebound current. *Neuron* 51, 113–123. doi: 10.1016/j.neuron.2006.05.021
- Rancz, E. A., Ishikawa, T., Duguid, I., Chadderton, P., Mahon, S., and Häusser, M. (2007). High-fidelity transmission of sensory information by single cerebellar mossy fibre boutons. *Nature* 450, 1245–1248. doi: 10.1038/nature05995
- Rasmussen, A., Jirenhed, D.-A., and Hesslow, G. (2008). Simple and complex spike firing patterns in Purkinje cells during classical conditioning. *Cerebellum* 7, 563–566. doi: 10.1007/s12311-008-0068-2
- Romano, V., De Propriis, L., Bosman, L. W., Warnaar, P., ten Brinke, M. M., Lindeman, S., et al. (2018). Potentiation of cerebellar Purkinje cells facilitates whisker reflex adaptation through increased simple spike activity. *eLife* 7:e38852. doi: 10.7554/eLife.38852
- Ruigrok, T. J. H. (2011). Ins and outs of cerebellar modules. *Cerebellum* 10, 464–474. doi: 10.1007/s12311-010-0164-y
- Schmahmann, J. D. (2019). The cerebellum and cognition. *Neurosci. Lett.* 688, 62–75. doi: 10.1016/J.NEULET.2018.07.005
- Schonewille, M., Belmeguenai, A., Koekkoek, S. K., Houtman, S. H., Boele, H. J., van Beugen, B. J., et al. (2010). Purkinje cell-specific knockout of the protein phosphatase PP2B impairs potentiation and cerebellar motor learning. *Neuron* 67, 618–628. doi: 10.1016/j.neuron.2010.07.009
- Shadmehr, R. (2017). Learning to predict and control the physics of our movements. *J. Neurosci.* 37, 1663–1671. doi: 10.1523/JNEUROSCI.1675-16.2016
- Si Feng, S., Lin, R., Gauck, V., and Jaeger, D. (2013). Gain control of synaptic response function in cerebellar nuclear neurons by a calcium activated potassium conductance. *Cerebellum* 12, 692–706. doi: 10.1007/s12311-013-0476-9
- Solinas, S., Forti, L., Cesana, E., Mapelli, J., De Schutter, E., and D'Angelo, E. (2007). Fast-reset of pacemaking and theta-frequency resonance patterns in cerebellar golgi cells: simulations of their impact in vivo. *Front. Cell. Neurosci.* 1:4. doi: 10.3389/neuro.03.004.2007
- Szapiro, G., and Barbour, B. (2007). Multiple climbing fibers signal to molecular layer interneurons exclusively via glutamate spillover. *Nat. Neurosci.* 10, 735–742. doi: 10.1038/nn1907
- Ten Brinke, M. M., Boele, H. J., Spanke, J. K., Potters, J. W., Kornysheva, K., Wulff, P., et al. (2015). Evolving models of pavlovian conditioning: cerebellar cortical dynamics in awake behaving mice. *Cell Rep.* 13, 1977–1988. doi: 10.1016/j.celrep.2015.10.057
- Ten Brinke, M. M., Heiney, S. A., Wang, X., Proietti-Onori, M., Boele, H. J., Bakermans, J., et al. (2017). Dynamic modulation of activity in cerebellar nuclei neurons during pavlovian eyeblink conditioning in mice. *eLife* 6:e28132. doi: 10.7554/eLife.28132
- Uusisaari, M., and de Schutter, E. (2011). The mysterious microcircuitry of the cerebellar nuclei. *J. Physiol.* 589, 3441–3457. doi: 10.1113/jphysiol.2010.201582
- Uusisaari, M., and Knöpfel, T. (2008). GABAergic synaptic communication in the GABAergic and non-GABAergic cells in the deep cerebellar nuclei. *Neuroscience* 156, 537–549. doi: 10.1016/J.NEUROSCIENCE.2008.07.060
- Voogd, J., and Glickstein, M. (1998). The anatomy of the cerebellum. *Trends Cogn. Sci.* 2, 307–313. doi: 10.1016/0166-2236(95)93903-B
- Wu, Y., and Raman, I. M. (2017). Facilitation of mossy fibre-driven spiking in the cerebellar nuclei by the synchrony of inhibition. *J. Physiol.* 595, 5245–5264. doi: 10.1113/JP274321
- Yamazaki, T., and Igarashi, J. (2013). Realtime cerebellum: a large-scale spiking network model of the cerebellum that runs in realtime using a graphics processing unit. *Neural Netw.* 47, 103–111. doi: 10.1016/J.NEUNET.2013.01.019
- Zanjani, H., Herrup, K., and Mariani, J. (2004). Cell number in the inferior olive of nervous and leaner mutant mice. *J. Neurogenet.* 18, 327–339. doi: 10.1080/01677060390449482
- Zheng, N., and Raman, I. M. (2010). Synaptic inhibition, excitation, and plasticity in neurons of the cerebellar nuclei. *Cerebellum* 9, 56–66. doi: 10.1007/s12311-009-0140-6
- Zhou, H., Lin, Z., Voges, K., Ju, C., Gao, Z., Bosman, L. W., et al. (2014). Cerebellar modules operate at different frequencies. *eLife* 3:e02536. doi: 10.7554/elif.02536

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Geminiani, Pedrocchi, D'Angelo and Casellato. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Embodied Synaptic Plasticity With Online Reinforcement Learning

Jacques Kaiser^{1*†}, Michael Hoff^{1,2†}, Andreas Konle¹, J. Camilo Vasquez Tieck¹, David Kappel^{2,3,4}, Daniel Reichard¹, Anand Subramoney², Robert Legenstein², Arne Roennau¹, Wolfgang Maass² and Rüdiger Dillmann¹

¹ FZI Research Center for Information Technology, Karlsruhe, Germany, ² Institute for Theoretical Computer Science, Graz University of Technology, Graz, Austria, ³ Bernstein Center for Computational Neuroscience, III Physikalisches Institut-Biophysik, Georg-August Universität, Göttingen, Germany, ⁴ Technische Universität Dresden, Chair of Highly Parallel VLSI Systems and Neuromorphic Circuits, Dresden, Germany

OPEN ACCESS

Edited by:

Judith Peters,
Maastricht University, Netherlands

Reviewed by:

Eiji Uchibe,
Advanced Telecommunications
Research Institute International
(ATR), Japan
Emmanuel Dauce,
Centrale Marseille, France

*Correspondence:

Jacques Kaiser
jkaiser@fzi.de

[†]These authors have contributed
equally to this work

Received: 01 February 2019

Accepted: 13 September 2019

Published: 03 October 2019

Citation:

Kaiser J, Hoff M, Konle A, Vasquez Tieck JC, Kappel D, Reichard D, Subramoney A, Legenstein R, Roennau A, Maass W and Dillmann R (2019) Embodied Synaptic Plasticity With Online Reinforcement Learning. *Front. Neurobot.* 13:81. doi: 10.3389/fnbot.2019.00081

The endeavor to understand the brain involves multiple collaborating research fields. Classically, synaptic plasticity rules derived by theoretical neuroscientists are evaluated in isolation on pattern classification tasks. This contrasts with the biological brain which purpose is to control a body in closed-loop. This paper contributes to bringing the fields of computational neuroscience and robotics closer together by integrating open-source software components from these two fields. The resulting framework allows to evaluate the validity of biologically-plausible plasticity models in closed-loop robotics environments. We demonstrate this framework to evaluate Synaptic Plasticity with Online REinforcement learning (SPORE), a reward-learning rule based on synaptic sampling, on two visuomotor tasks: reaching and lane following. We show that SPORE is capable of learning to perform policies within the course of simulated hours for both tasks. Provisional parameter explorations indicate that the learning rate and the temperature driving the stochastic processes that govern synaptic learning dynamics need to be regulated for performance improvements to be retained. We conclude by discussing the recent deep reinforcement learning techniques which would be beneficial to increase the functionality of SPORE on visuomotor tasks.

Keywords: neurobotics, synaptic plasticity, spiking neural networks, neuromorphic vision, reinforcement learning

1. INTRODUCTION

The brain evolved over millions of years for the sole purpose of controlling the body in a goal-directed fashion. Computations are performed relying on neural dynamics and asynchronous communication. Spiking neural network models base their computations on these computational principles. Biologically plausible synaptic plasticity rules for functional learning in spiking neural networks are regularly proposed (Pfister et al., 2006; Urbanczik and Senn, 2014; Neftci, 2017; Kaiser et al., 2018; Zenke and Ganguli, 2018). In general, these rules are derived to minimize a distance (referred to as error) between the output of the network and a target. Therefore, the evaluation of these rules is usually carried out on open-loop pattern classification tasks. By neglecting the embodiment, this type of evaluation disregards the closed-loop dynamics the brain has to handle with the environment. Indeed, the decisions taken by the brain have an impact on the environment, and this change is sensed back by the brain. To get a deeper understanding of the plausibility

of these rules, an embodied evaluation is necessary. This evaluation is technically complicated since spiking neurons are dynamical systems that must be synchronized with the environment. Additionally, as in biological bodies, sensory information, and motor commands need to be encoded and decoded respectively.

In this paper, we bring the fields of computational neuroscience and robotics closer together by integrating open-source software components from these two fields. The resulting framework is capable of learning online the control of simulated and real robots with a spiking network in a modular fashion. This framework is demonstrated in the evaluation of the promising neural reward-learning rule SPORE (Kappel et al., 2014, 2015, 2018; Yu et al., 2016) on two closed-loop robotic tasks. SPORE is an instantiation of the synaptic sampling scheme introduced in Kappel et al. (2018, 2015). It incorporates a policy sampling method which models the growth of dendritic spines with respect to dopamine influx. Unlike current state-of-the-art reinforcement learning methods implemented with conventional neural networks (Lillicrap et al., 2015; Mnih et al., 2015, 2016), SPORE learns online from precise spike-time and is entirely implemented with spiking neurons. We evaluate this learning rule in a closed-loop reaching and a lane following (Kaiser et al., 2016; Bing et al., 2018a) setup. In both tasks, an end-to-end visuomotor policy is learned, mapping visual input to motor commands. In the last years, important progress have been made on learning control from visual input with deep learning. However, deep learning approaches are computationally expensive and rely on biologically implausible mechanisms such as dense synchronous communication and batch learning. For networks of spiking neurons learning visuomotor tasks online with synaptic plasticity rules remains challenging. In this paper, visual input is encoded in Address Event Representation with a Dynamic Vision Sensor (DVS) simulation (Lichtsteiner et al., 2008; Kaiser et al., 2016). This representation drastically reduces the redundancy of the visual input as only motion is sensed, allowing more efficient learning. It agrees with the two pathways hypothesis which states that motion is processed separately than color and shape in the visual cortex (Kruger et al., 2013).

The main contribution of this paper is the embodiment of SPORE and its evaluation on two neurorobotic tasks using a combination of open-source software components. This embodiment allowed us to identify crucial techniques to regulate SPORE learning dynamics, not discussed in previous works where this learning rule was only evaluated on simple proof-of-concept learning problems (Kappel et al., 2014, 2015, 2018; Yu et al., 2016). Our results suggest that an external mechanism such as learning rate annealing is beneficial to retain a performing policy on advanced lane following task.

This paper is structured as follows. We provide a review of the related work in section 2. In section 3, we give a brief overview of SPORE and discuss the contributed techniques required for its embodiment. The implementation and evaluation on the two chosen neurorobotic tasks is carried out in section 4. Finally, we discuss in section 5 how the method could be improved.

2. RELATED WORK

The year 2015 marked a significant breakthrough in deep reinforcement learning. Artificial neural networks of analog neurons are now capable of solving a variety of tasks ranging from playing video games (Mnih et al., 2015), to controlling multi-joints robots (Lillicrap et al., 2015; Schulman et al., 2017), and lane following (Wolf et al., 2017). Most recent methods (Lillicrap et al., 2015; Schulman et al., 2015, 2017; Mnih et al., 2016) are based on policy-gradients. Specifically, policy parameters are updated by performing ascending gradient steps with backpropagation to maximize the probability of taking rewarding actions. While functional, these methods are not based on biologically plausible processes. First, a large part of neural dynamics are ignored. Importantly, unlike SPORE, these methods do not learn online—weight updates are performed with respect to entire trajectories stored in rollout memory. Second, learning is based on backpropagation which is not biologically plausible learning mechanism, as stated in Bengio et al. (2015).

Spiking network models inspired by deep reinforcement learning techniques were introduced in Bellec et al. (2018) and Tieck et al. (2018). In both papers, the spiking networks are implemented with deep learning frameworks (PyTorch and TensorFlow, respectively) and rely on automatic differentiation. Their policy-gradient approach is based on (PPO; Schulman et al., 2017). As the learning mechanism consists of backpropagating the Proximal Policy Optimization (PPO) loss (through-time in the case of Bellec et al., 2018), most biological constraints stated in Bengio et al. (2015) are still violated. Indeed, the computations are based on spikes (4), but the backpropagation is purely linear (1), the feedback paths require precise knowledge of the derivatives (2) and weights (3) of the corresponding feedforward paths, and the feedforward and feedback phases alternate synchronously (5) (the enumeration refers to Bengio et al., 2015).

Only a small body of work focused on reinforcement learning with spiking neural networks, while addressing the previous points. Groundwork of reinforcement learning with spiking networks was presented in Florian (2007), Izhikevich (2007), and Legenstein et al. (2008). In these works, a mathematical formalization is introduced characterizing how dopamine modulated spike-timing-dependent plasticity (DA-STDP) solves the distal reward problem with eligibility traces. Specifically, since the reward is received only after a rewarding action is performed, the brain needs a form of memory to reinforce previously chosen actions. This problem is solved with the introduction eligibility traces, which assign credit to recently active synapses. This concept has been observed in the brain (Frey and Morris, 1997; Pan et al., 2005), and SPORE also relies on eligibility traces. Fewer works evaluated DA-STDP in an embodiment for reward maximization—a recent survey encompassing this topic is available in Bing et al. (2018b).

The closest previous work related to this paper are Daucé (2009), Kaiser et al. (2016), and Bing et al. (2018a). In Kaiser et al. (2016), a neurorobotic lane following task is presented, where a simulated vehicle is controlled end-to-end from event-based

vision to motor command. The task is solved with an hard-coded spiking network of 16 neurons implementing a simple Braitenberg vehicle. The performance is evaluated with respect to distance and orientation differences to the middle of the lane. In this paper, these performance metrics are combined into a reward signal which the spiking network maximizes with the SPORE learning rule.

In Bing et al. (2018a), the authors evaluate DA-STDP (referred to as R-STDP for reward-modulated STDP) in a similar lane following environment. Their approach outperforms the hard-coded Braitenberg vehicle presented in Kaiser et al. (2016). The two motor neurons controlling the steering receive different (mirrored) reward signals whether the vehicle is on the left or on the right of the lane. This way, the reward provides the information of what motor command should be taken, similar to a supervised learning setup. Conversely, the approach presented in this paper is more generic since a global reward is distributed to all synapses and does not indicate which action the agent should take.

A similar plasticity rule implementing a policy-gradient approach is derived in Daucé (2009). Also relying on eligibility traces, this reward-learning rule uses a “slow” noise term to drive the exploration. This rule is demonstrated on a target reaching task comparable to the one discussed in section 4.1.1 and achieves impressive learning times (in the order of 100s) with proper tuning of the noise term.

In Nakano et al. (2015), a spiking version of the free-energy-based reinforcement learning framework proposed in Otsuka et al. (2010) is introduced. In this framework, a spiking Restricted Boltzmann Machine (RBM) is trained with a reward-modulated plasticity rule which decreases the free-energy of rewarding state-action pairs. The approach is evaluated on discrete-actions tasks where the observations consist of MNIST digits processed by a pre-trained feature extractor. However, some characteristics of RBM are biologically implausible and make their implementation cumbersome: symmetric synapses and clocked network activity. With our approach, network activity does not have to be manually synchronized into observation and action phases of arbitrary duration for learning to take place.

In Gilra and Gerstner (2017), a supervised synaptic learning rule named Feedback-based Online Local Learning Of Weights (FOLLOW) is introduced. This rule is used to learn the inverse dynamics of a two-link arm—the model predicts control commands (torques) for a given arm trajectory. The loop is closed in Gilra and Gerstner (2018) by feeding the predicted torques as control commands. In contrast, SPORE learns from a reward signal and can solve a variety of tasks.

3. METHODS

In this section, we give a brief overview of the reward-based learning rule SPORE. We then discuss how SPORE was embodied in closed-loop, along with our modifications to increase the robustness of the learned policy.

3.1. Synaptic Plasticity With Online Reinforcement Learning (SPORE)

Throughout our experiments we use an implementation of the reward-based online learning rule for spiking neural networks, named *synaptic sampling*, that was introduced in Kappel et al. (2018). The learning rule employs synaptic updates that are modulated by a global reward signal to maximize the expected reward. More precisely, the learning rule does not converge to a local maximum θ^* of the synaptic parameter vector θ , but it continuously samples different solutions $\theta \sim p^*(\theta)$ from a target distribution that peaks at parameter vectors that likely yield high reward. A temperature parameter T allows to make the distribution $p^*(\theta)$ flatter (high exploration) or more peaked (high exploitation).

SPORE (Kappel et al., 2017) is an implementation of the reward-based synaptic sampling rule (Kappel et al., 2018), that uses the NEST neural simulator (Gewaltig and Diesmann, 2007). SPORE is optimized for closed-loop applications to form an online policy-gradient approach. We briefly review here the main features of the synaptic sampling algorithm.

We consider the goal of reinforcement learning to maximize the expected future discounted reward $\mathcal{V}(\theta)$ given by

$$\mathcal{V}(\theta) = \left\langle \int_0^\infty e^{-\frac{\tau}{\tau_e}} r(\tau) d\tau \right\rangle_{p(r|\theta)}, \quad (1)$$

where $r(\tau)$ denotes the reward at time τ and τ_e is a time constant that discounts remote rewards. We consider non-negative reward $r(\tau) \geq 0$ at any time such that $\mathcal{V}(\theta) \geq 0$ for all θ . The distribution $p(r|\theta)$ denotes the probability of observing the sequence of reward r under a given parameter vector θ . Note that computing this expectation involves averaging over a number of experimental trials and network responses.

As proposed in Kappel et al. (2018) we replace the standard goal of reinforcement learning to maximize the objective function in Equation (1) by a probabilistic framework that generates samples from the parameter vector θ according to some target distribution $\theta \sim p^*(\theta)$. We will focus on sampling from the target distribution $p^*(\theta)$ of the form

$$p^*(\theta) \propto p(\theta) \times \mathcal{V}(\theta), \quad (2)$$

where $p(\theta)$ is a prior distribution over the network parameters that allows us, for example, to introduce constraints on the sparsity of the network parameters. It has been shown in Kappel et al. (2018) that the learning goal in is achieved, if all synaptic parameters θ_i obey the stochastic differential equation

$$d\theta_i = \beta \left(\frac{\partial}{\partial \theta_i} \log p(\theta) + \frac{\partial}{\partial \theta_i} \log \mathcal{V}(\theta) \right) dt + \sqrt{2\beta T} dW_i, \quad (3)$$

where β is a scaling parameter that functions as a learning rate, dW_i are the stochastic increments and decrements of a Wiener process, and T is the temperature parameter. $\frac{\partial}{\partial \theta_i}$ denotes the partial derivative with respect to the synaptic parameter θ_i . The stochastic process in generates samples of θ that are with high probability close to the local optima of the target distribution $p^*(\theta)$.

It has been further shown in Kappel et al. (2018) that can be implemented using a synapse model with local update rules. The state of each synapse i consists of the dynamic variables $y_i(t)$, $e_i(t)$, $g_i(t)$, $\theta_i(t)$, and $w_i(t)$. The variable $y_i(t)$ is the pre-synaptic spike train filtered with a post-synaptic-potential kernel. $e_i(t)$ is the eligibility trace that maintains a brief history of pre-/post neural activity. $g_i(t)$ is a variable to estimate the reward gradient, i.e., the gradient of the objective function in Equation (1) with respect to the synaptic parameter $\theta_i(t)$. $w_i(t)$ denotes the weight of synapse i at time t . In addition each synapse has access to the global reward signal $r(t)$. The variables $e_i(t)$, $g_i(t)$, and $\theta_i(t)$ are updated by solving the differential equations:

$$\frac{de_i(t)}{dt} = -\frac{1}{\tau_e} e_i(t) + w_i(t) y_i(t) (z_{post_i}(t) - \rho_{post_i}(t)) \quad (4)$$

$$\frac{dg_i(t)}{dt} = -\frac{1}{\tau_g} g_i(t) + r(t) e_i(t) \quad (5)$$

$$d\theta_i(t) = \beta \left(c_p (\mu - \theta_i(t)) + c_g g_i(t) \right) dt + \sqrt{2T_\theta \beta} \mathcal{W}_i, \quad (6)$$

where $z_{post_i}(t)$ is a sum of Dirac delta pulses placed at the firing times of the post-synaptic neuron, μ is the prior mean of synaptic parameters [$p(\theta)$ in Equation 2] and $\rho_{post_i}(t)$ is the instantaneous firing rate of the post-synaptic neuron at time t . The constants c_p and c_g are tuning parameters of the algorithm that scale the influence of the prior distribution $p(\theta)$ against the influence of the reward-modulated term. Setting $c_p = 0$ corresponds to a non-informative (flat) prior. In general, the prior distribution is modeled as a Gaussian centered around μ : $p(\theta) = \mathcal{N}(\mu, \frac{1}{c_p})$. We used $\mu = 0$ in our simulations. The variance of the reward gradient estimation (Equation 5) could be reduced by subtracting a baseline to the reward as introduced in Williams (1992), although this was not investigated in this paper.

Finally the synaptic weights are given by the projection

$$w_i(t) = \begin{cases} w_0 \exp(\theta_i(t) - \theta_0) & \text{if } \theta_i(t) > 0 \\ 0 & \text{otherwise} \end{cases}, \quad (7)$$

which scaling and offset parameters w_0 and θ_0 , respectively.

In SPORE the differential equations Equations (4) to (6) are solved using the Euler method with a time step of 1 ms. The dynamics of the post-synaptic term $y_i(t)$, the eligibility trace $e_i(t)$, and the reward gradient $g_i(t)$ are updated at each time step. The dynamics of $\theta_i(t)$ and $w_i(t)$ are updated on a coarser time grid with step width 100 ms for the sake of simulation speed. The synaptic weights remain constant between two updates. Synaptic parameters are clipped at θ_{min} and θ_{max} . Parameter gradients $g_i(t)$ are clipped at $\pm \Delta \theta_{max}$. The parameters used in our evaluation are stated in Tables 1–3.

3.2. Closed-Loop Embodiment Implementation

Usually, synaptic learning rules are solely evaluated on open-loop pattern classification tasks (Pfister et al., 2006; Urbanczik and Senn, 2014; Neftci, 2017; Zenke and Ganguli, 2018). An embodied evaluation is technically more involved and requires a

TABLE 1 | NEST parameters.

Time-step/resolution	1 ms
Synapse update interval	100 ms
(reaching) exploration noise	35 Hz
(reaching) noise to exploration exc.	750.0
(reaching) visual to exploration inh.	$\mathcal{N}(-500, 50)$
(reaching) exploration to motor exc.	10.0

TABLE 2 | SPORE parameters.

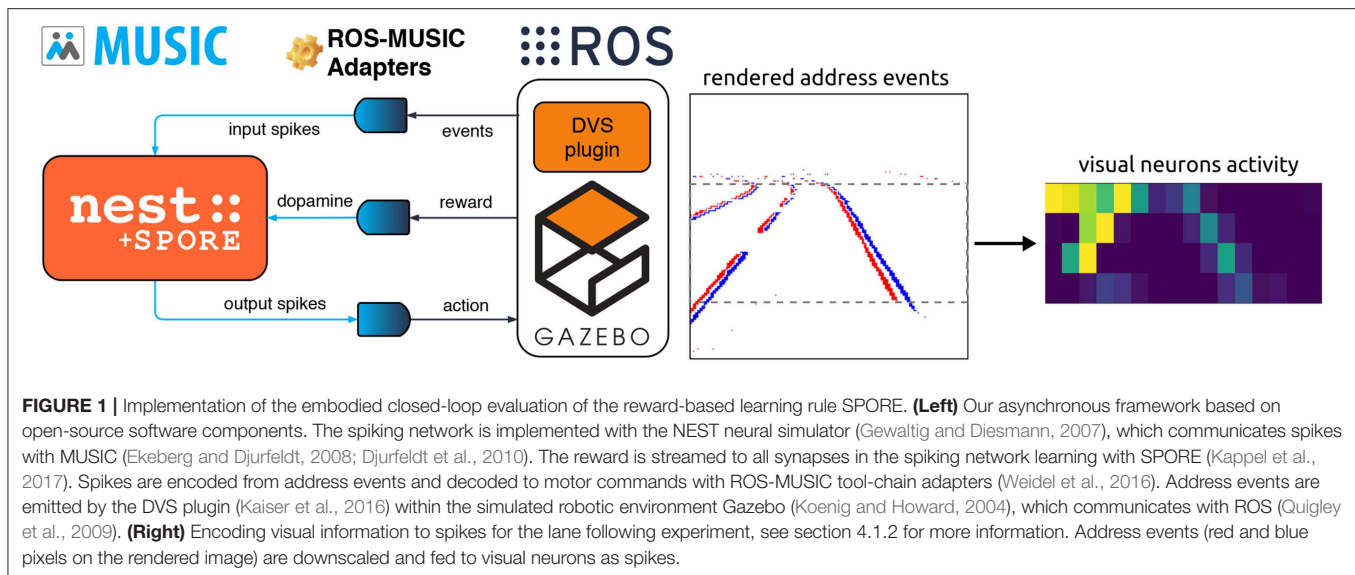
Visual to motor exc.	$\mathcal{N}(0.8, 0.6)$ (clipped at 0)
Visual to motor mul.	10
Temperature (T)	0.1
Initial learning rate (β)	1×10^{-7}
Learning rate decay (λ)	8.5×10^{-5}
Integration time	50 s
Max synaptic parameter (θ_{max})	5.0
Min synaptic parameter (θ_{min})	-2.0
(reaching) episode length	1 s
(lane following) episode length	2 s

TABLE 3 | ROS-MUSIC parameters.

MUSIC time-step	1 ms ... 3 ms
DVS adapter time-step	1 ms
Decoder time constant	100 ms

closed-loop environment simulation. A core contribution of this paper is the implementation of a framework allowing to evaluate the validity of bio-plausible plasticity models in closed-loop robotics environments. We rely on this framework to evaluate the synaptic sampling rule SPORE (Kappel et al., 2017), as depicted in Figure 1. This framework is tailored for evaluating spiking network learning rules in an embodiment. Visual sensory input is sensed, encoded as spikes, processed by the network, and output spikes are converted to motor commands. The motor commands are executed by the agent, which modifies the environment. This modification of the environment is sensed by the agent. Additionally, a continuous reward signal is emitted from the environment. SPORE tries to maximize this reward signal online by steering the ongoing synaptic plasticity processes of the network toward configurations which are expected to yield more overall reward. Unlike classical reinforcement learning setup, the spiking network is treated as a dynamical system continuously receiving input and outputting motor commands. This allows us to report learning progress with respect to (biological) simulated time, unlike classical reinforcement learning which reports learning progress in number of iterations. Similarly, we reset the agent only when the task is completed (in the reaching task) or when the agent goes off-track (in the lane following task). We do not enforce finite-time episodes and neither the agent nor SPORE are notified of the reset.

This framework relies on many open-source software components: As neural simulator we use NEST (Gewaltig



and Diesmann, 2007) combined with the open-source implementation of SPORE (Kappel et al., 2018)¹. The robotic simulation is managed by Gazebo (Koenig and Howard, 2004) and ROS (Quigley et al., 2009) and visual perception is realized using the open-source DVS plugin for Gazebo (Kaiser et al., 2016)². This plugin emits polarized address events when variations in pixel intensity cross a threshold. The robotic simulator and the neural network run in different processes. We rely on MUSIC (Ekeberg and Djurfeldt, 2008; Djurfeldt et al., 2010) to communicate and transform the spikes and we employ the ROS-MUSIC tool-chain by Weidel et al. (2016) to bridge between the two communication frameworks. The latter also synchronizes ROS time with spiking network time. Most of these components are also integrated in the Neurorobotics Platform (NRP) Falotico et al. (2017), except for MUSIC and the ROS-MUSIC tool-chain. Therefore, the NRP does not support streaming a reward signal to all synapses, required in our experiments.

As part of this work, we contributed to the Gazebo DVS plugin by integrating it to ROS-MUSIC, and to the SPORE module by integrating it with MUSIC. These contributions enable researchers to design new ROS-MUSIC experiments using event-based vision to evaluate SPORE or their own biologically-plausible learning rules. A clear advantage of this framework is that the robotic simulation can be substituted for a real robot seamlessly. However, the necessary human supervision in real robotics coupled with the many hours needed by SPORE to learn a performing policy is currently prohibitive. The simulation of the whole framework was conducted on a Quad core Intel Core i7-4790K with 16GB RAM in real-time.

¹<https://github.com/IGITUGraz/spore-nest-module>

²https://github.com/HBPNeurorobotics/gazebo_dvs_plugin

3.3. Learning Rate Annealing

In the original work presenting SPORE (Kappel et al., 2014, 2015, 2018; Yu et al., 2016), the learning rate β and the temperature T were kept constant throughout the learning process. Note that in deep learning, learning rates are often regulated by the optimization processes (Kingma and Ba, 2014). We found that the learning rate β of SPORE plays an important role in learning and benefit from an annealing mechanism. This regulation allows the synaptic weights to converge to a stable configuration and prevents the network to forget previous policy improvements. For the lane following experiment presented in this paper, the learning rate β is decreased over time, which also reduces the temperature (random exploration), see Equation (3). Specifically, we decay the learning rate β exponentially with respect to time:

$$\frac{d\beta(t)}{dt} = -\lambda\beta(t). \quad (8)$$

The learning rate is updated following this equation every 10 min. Independently decaying the temperature term T was not investigated, however we expect a minor impact on the performance because of the high variance of the reward gradient estimation, intrinsically leading the agent to explore.

4. EVALUATION

We evaluate our approach on two neurorobotic tasks: a reaching task and the lane following task presented in Kaiser et al. (2016) and Bing et al. (2018a). In the following sections, we describe these tasks and the ability of SPORE to solve them. Additionally, we analyze the performance and stability of the learned policies with respect to the prior distribution $p(\theta)$ and learning rate β (see Equation 3).

4.1. Experimental Setup

The tasks used for our evaluation are depicted in Figure 2. In both tasks, a feed-forward all-to-all two-layers network of

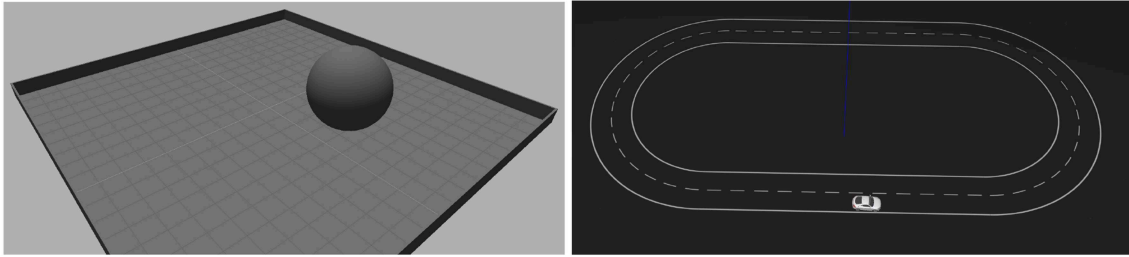


FIGURE 2 | Visualization of the setup for the two experiments. **(Left)** Reaching experiment. The goal of the task is to control the ball to the center of the plane. Visual input is provided by a DVS simulation above the plane looking downward. The ball is controlled with Cartesian velocity vectors. **(Right)** Lane following experiment. The goal of the task is to keep the vehicle on the right lane of the road. Visual input is provided by a DVS simulation attached to the vehicle looking forward to the road. The vehicle is controlled with steering angles.

spiking neurons is trained with SPORE to maximize a task-specific reward. Previous work has shown that this architecture was sufficient for the task complexity considered (Daucé, 2009; Kaiser et al., 2016; Bing et al., 2018a). The network is end-to-end and maps the address events of a simulated DVS to motor commands. The parameters used for the evaluation are presented in **Tables 1–3**. In the next paragraphs, we describe the tasks together with their decoding schemes and reward functions.

4.1.1. Reaching Task

The reaching task is a natural extension of the open-loop blind reaching task on which SPORE was evaluated in Yu et al. (2016). A similar visual tracking task was presented in Daucé (2009), with a different visual input encoding. In our setup, the agent controls a ball of 2 m radius which has to move toward the 2 m radius center of a 20×20 m plane enclosed with walls. Sensory input is provided by a simulated DVS with a resolution of 16×16 pixels located above the center which perceives the ball and the entire plane. There is one visual neuron corresponding to each DVS pixel—we make no distinctions between ON and OFF events. We additionally enhance the input space with an axis feature neuron for each row and each column. These neurons fire for each spikes in the respective row or column of neurons they cover. Both 16×16 visual neurons and 2×16 axis feature neurons are connected to all 8 motor neurons with 10 plastic SPORE synapses, resulting in 23,040 learnable parameters. The network controls the ball with instantaneous velocity vectors through the Gazebo Planar Move Plugin. Velocity vectors are decoded from output spikes with the linear decoder:

$$v = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos(\beta_1) & \cos(\beta_2) & \dots & \cos(\beta_N) \\ \sin(\beta_1) & \sin(\beta_2) & \dots & \sin(\beta_N) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} \quad (9)$$

$$\beta_k = \frac{2k\pi}{N},$$

with a_k the activity of motor neuron k obtained by applying a low-pass filter on the spikes with time constant τ . This decoding scheme consists of equally distributing N motor neurons on a circle representing their contribution to the displacement vector.

For our experiment, we set $N = 8$ motor neurons. We add an additional exploration neuron to the network which excites the motor neurons and is inhibited by the visual neurons. This neuron prevents long periods of immobility. Indeed, when the agent decides to stay motionless, it does not receive any sensory input as the DVS simulation only senses change. Since the network is feedforward, the absence of sensory input causes the neural activity to drop, leading to more immobility.

The ball is reset to a random position on the plane if it has reached the center. This reset is not signaled to the network—aside from the abrupt change in visual input—and does not mark the end of an episode. Let β_{err} denote the absolute value of the angle between the straight line to the goal and the direction taken by the ball. The agent is rewarded if the ball moves in the direction toward the goal $\beta_{err} < \beta_{lim}$ at a sufficient velocity $v > v_{lim}$. Specifically, the reward $r(t)$ is computed as:

$$r(t) = 35\sqrt{r_v}(r_\beta + 1)^5$$

$$r_\beta = \begin{cases} 1 - \frac{\beta_{err}}{\beta_{lim}}, & \text{if } \beta_{err} < \beta_{lim} \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

$$r_v = \begin{cases} |v|, & \text{if } |v| > v_{lim} \\ 0, & \text{otherwise} \end{cases}.$$

This signal is smoothed with an exponential filter before being streamed to the agent. This formulation provides a continuous feedback to the agent, unlike delivering a discrete terminal reward upon reaching the goal state. In our experiments, discrete terminal rewards did not suffice for the agent to learn performing policies in a reasonable amount of time. On the other hand, distal rewards are supported by SPORE through eligibility traces, as was demonstrated in Yu et al. (2016) and Kappel et al. (2018), for open-loop tasks with clearly delimited episodes. This suggests that additional mechanisms or hyperparameter tuning would be required for SPORE to learn from distal rewards online.

4.1.2. Lane Following Task

The lane following task was already used to demonstrate spiking neural controllers in Kaiser et al. (2016) and Bing et al. (2018a). The goal of the task is to steer a vehicle to stay on the right lane of a track. Sensory input is provided by a simulated DVS

with a resolution of 128x32 pixels mounted on top of the vehicle showing the track in front. There are 16x4 visual neurons covering the pixels, each neuron responsible for a 8x8 pixel window. Each visual neuron spikes at a rate correlated to the amount of events in its window (see **Figure 1**). The vehicle starts driving on a fixed starting point with a constant velocity on the right lane of the track. As soon as the vehicle leaves the track, it is reset to the starting point. As in the reaching task, this reset is not explicitly signaled to the network and does not mark the end of a learning episode.

The network controls the angle of the vehicle by steering it, while its linear velocity is constant. The output layer is separated into two neural populations. The steering commands sent to the agent consist of the difference of activity between these two populations. Specifically, steering commands are decoded from output spikes as a ratio between the following linear decoders:

$$\begin{aligned} a_L &= \sum_{i=1}^{N/2} a_i, \\ a_R &= \sum_{i=N/2+1}^N a_i, \\ r &= \frac{a_L - a_R}{a_L + a_R}. \end{aligned} \quad (11)$$

The first $N/2$ neurons pull the steering on one side, while the remaining $N/2$ neurons pull steering to the other side. We set $N = 8$ so that there are 4 left motor neurons and 4 right motor neurons. The steering command is obtained by discretizing the ratio r into five possible commands: hard left (-30°), left (-15°), straight (0°), right (15°), and hard right (30°). The decision boundaries between these steering angles

are $r = \{-10, -2.5, 2.5, 10\}$, respectively. This discretization is similar than the one used in Wolf et al. (2017). It yielded better performance than directly using r (multiplied with a scaling constant k) as a continuous-space steering command as in Kaiser et al. (2016).

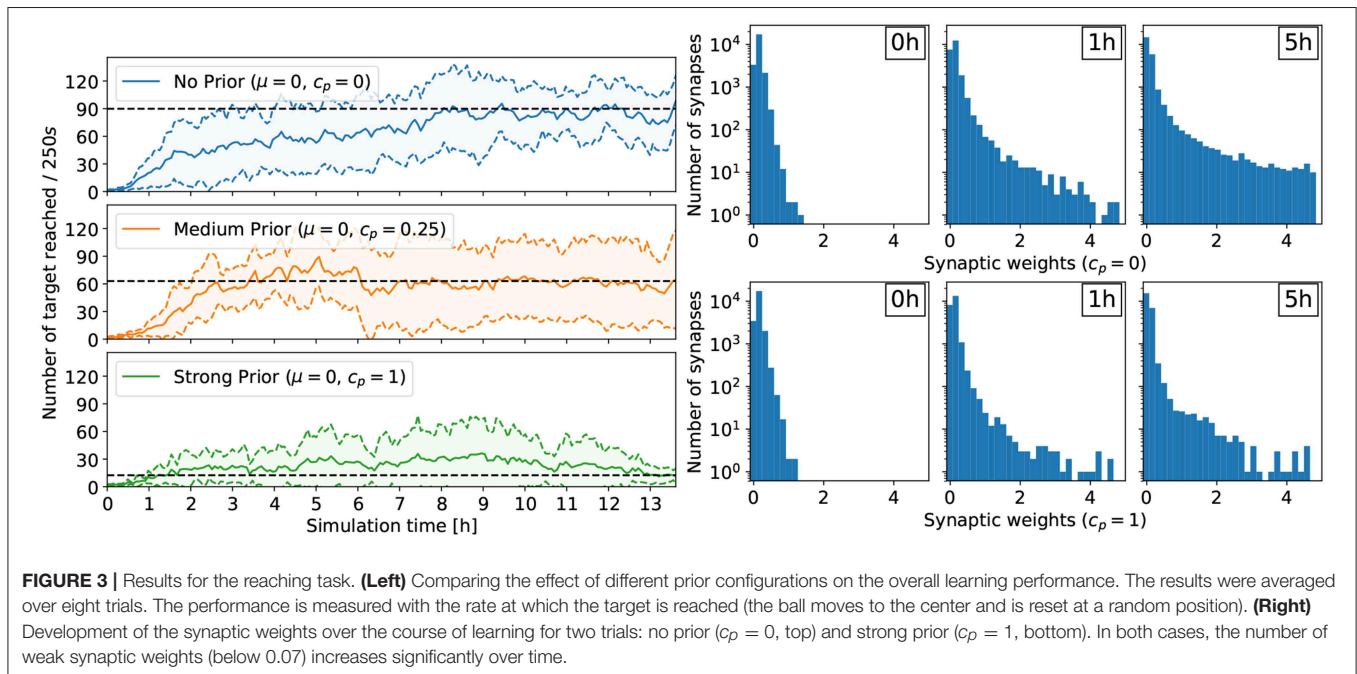
The reward signal delivered to the vehicle is equivalent to the performance metrics used in Kaiser et al. (2016) to evaluate the policy. As in the reaching task, the reward depends on two terms—the angular error β_{err} and the distance error d_{err} . The angular error β_{err} is the absolute value of the angle between the right lane and the vehicle. The distance error d_{err} is the distance between the vehicle and the center of the right lane. The reward $r(t)$ is computed as:

$$r(t) = e^{-0.03 \beta_{\text{err}}^2} \times e^{-70 d_{\text{err}}^2}. \quad (12)$$

The constants are chosen so that the score is halved every 0.1m distance error or 5° angular error. Note that this reward function is comprised between $[0, 1]$ and is less informative than the error used in Bing et al. (2018a). In our case, the same reward is delivered to all synapses, and a particular reward value does not indicate whether the vehicle is on the left or on the right of the lane. The decay of the learning rate is $\lambda = 8.5 \times 10^{-5}$ (see **Table 2**).

4.2. Results

Our results show that SPORE is capable of learning policies online for moderately difficult embodied tasks within some simulated hours (see **Supplementary Video**). We first discuss the results on the reaching task, where we evaluated the impact of the prior distribution. We then present the results on the lane following task, where the impact of the learning rate was evaluated.



4.2.1. Impact of Prior Distribution

For the reaching task, a flat prior $c_p = 0$ yielded the policy with highest performance (see Figure 3). In this case, the performance improves rapidly within a few hours of simulated time, and the ball reaches the center about 90 times every 250 s. Conversely, a strong prior ($c_p = 1$) forcing the synaptic weights close to 0 prevented performing policies to emerge. In this case, after 13 h of learning, the ball reaches the center only about 10 times on average every 250 s, a performance comparable to the random policy. Less constraining priors also affected the performance of the learned policies compared to the unconstrained case, but allowed learning to happen. With $c_p = 0.25$, the ball reaches the center about 60 times on average every 250 s. Additionally,

the number of retracting synapses increases over time—even in the flat prior case—reducing the computational overhead, important for a neuromorphic hardware implementation (Bellec et al., 2017). Indeed, for $c_p = 0$, the number of weak synaptic weights (below 0.07) increased from 3,329 to 7,557 after 1 h of learning to 14,753 after 5 h of learning (out of 23,040 synapses in total). In other words, only 36% of all synapses are active. The weight distribution for $c_p = 0.25$ is similar to the no-prior case $c_p = 0$. The strong prior $c_p = 1$ prevented strong weights to form, trading-off performance. The same trend is observed for the lane following task, where only 33% of all synapses are active after 4 h of learning (see Figure 5).

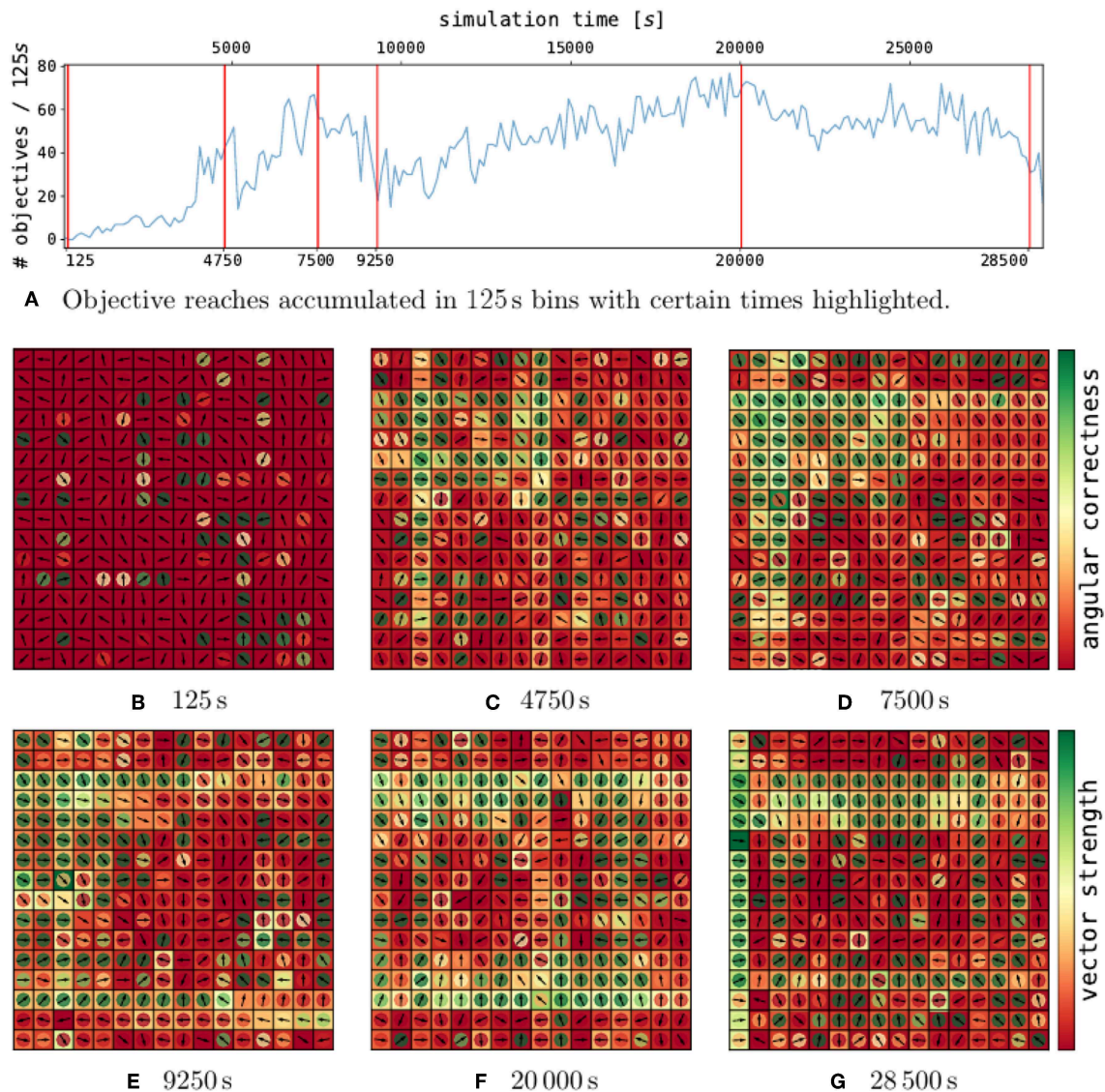


FIGURE 4 | Policy development for selected points in time in a single trial. On the top (A), the performance over time for a single, well-performing trial is depicted. The red lines indicate certain points in time, for which the policies are shown in (B–G). Each policy plot consists of a 2d-grid representing the DVS pixels. Hereby, every pixel contains a vector, which indicates the motion corresponding to the contribution of an event emitted by this pixel. The magnitude of the contribution (vector strength) is indicated by the outer pixel area. The inner circle color represents the assessment of the vector direction (angular correctness).

The analysis of a single trial with $c_p = 0.25$ is depicted in **Figure 4**. The performance does not converge and rather rise and drop while the network is sampling configurations. On initialization (**Figure 4B**), the policy employs weak actions with random directions.

After over 4.750 s of learning (**Figure 4C**), the first local maximum is reached. Vector directions have largely turned toward the grid center (see inner pixel colors). Additionally, the overall magnitude of the weights has largely increased, as could be expected from the weight histogram in **Figure 3**. In particular, patterns of single rows and columns emerge, due to the 2×16 axis feature neurons described in section 4.1.1. One drawback of the axis feature neurons can be seen in the center column of pixel. The axis feature neuron responsible for this column learned to push the ball down, since the ball mostly visited the upper part of the grid. However, at the center, the correct direction to push the ball toward the center is flipped.

At 7.500 s (**Figure 4D**), the performance has further increased. The policy, as shown in the second peak has grown even stronger for many pixels which also point in the right direction. The pixels pointing in the wrong direction mostly have a low vector strength.

After 9.250 s (**Figure 4E**), the performance drops to half its previous performance. As we can see from the policy, the weights grew even stronger. Some strong pixels vectors pointing toward each other have emerged, which can lead to the ball constantly moving up and down, without receiving any reward.

After this valley, the performance rises slowly again and at 20 000 s of simulation time (**Figure 4F**) the policy has reached the maximum performance of this trial. Around the whole grid, strong motion vectors push the ball toward the center, and the ball reaches the center around 140 times every 250 s.

Just before the end of the trial, the performance drops again (**Figure 4G**). Most vectors still point toward the right direction, however, the overall strength has largely decreased.

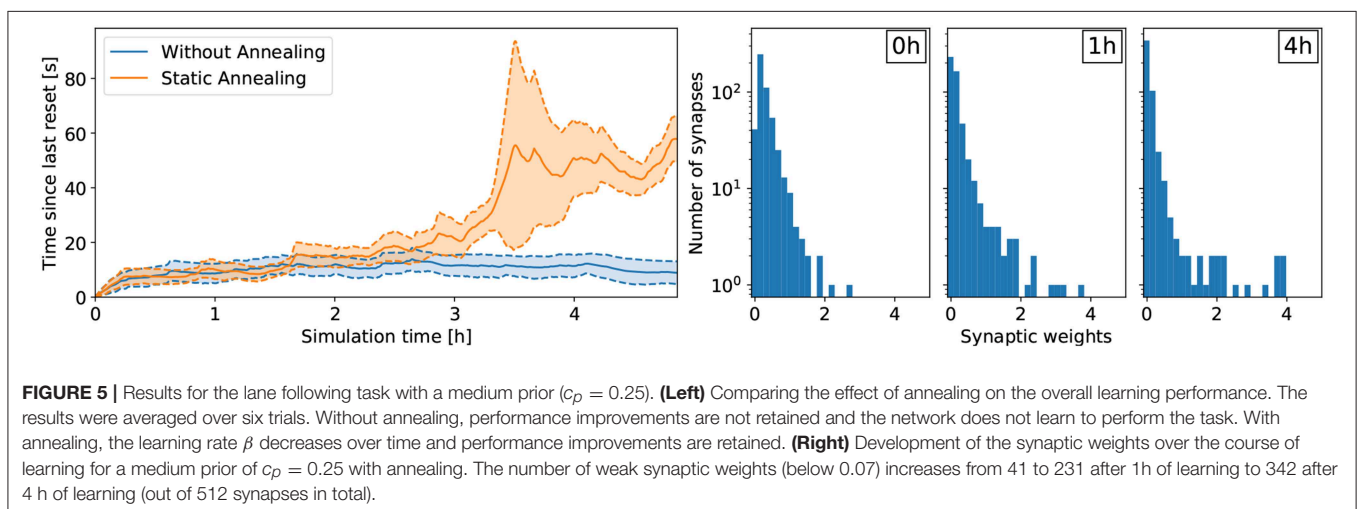
4.2.2. Impact of Learning Rate

For the lane following experiment, we show that the learning rate β plays an important role for retaining policy improvements.

Specifically, when the learning rate β remains constant over the course of learning, the policy does not improve compared to random (see **Figure 5**). In the random case, the vehicle remains about 10 s on the right lane until triggering a reset. After about 3 h of learning, the learning rate β decreased to 40% of its initial value and the policy starts to improve. After 5 h of learning, the learning rate β approaches 20% of its initial value and the performance improvements are retained. Indeed, while the weights are not frozen, the amplitude of subsequent synaptic updates are drastically reduced. In this case, the policy is significantly better than random and the vehicle remains on the right lane about 60 s on average.

5. CONCLUSION

The endeavor to understand the brain spans over multiple research fields. Collaborations allowing synaptic learning rules derived by theoretical neuroscientists to be evaluated in closed-loop embodiment are an important milestone of this endeavor. In this paper, we successfully implemented a framework allowing this evaluation by relying on open-source software components for spiking network simulation (Gewaltig and Diesmann, 2007; Kappel et al., 2017), synchronization and communication (Ekeberg and Djurfeldt, 2008; Quigley et al., 2009; Djurfeldt et al., 2010; Weidel et al., 2016), and robotic simulation (Koenig and Howard, 2004; Kaiser et al., 2016). The resulting framework is capable of learning online the control of simulated and real robots with a spiking network in a modular fashion. This framework is used to evaluate the reward-learning rule SPORE (Kappel et al., 2014, 2015, 2018; Yu et al., 2016) on two closed-loop visuomotor tasks. Overall, we have shown that SPORE was capable of learning shallow feedforward policies online for moderately difficult embodied tasks within some simulated hours. This evaluation allowed us to characterize the influence of the prior distribution on the learned policy. Specifically, constraining priors deteriorate the performance of the learned policy but prevent strong synaptic weights to emerge (see **Figure 3**). Additionally, for the lane following experiment, we have shown how learning rate regulation enabled policy



improvements to be retained. Inspired by simulated annealing, we presented a simple method decreasing the learning rate over time. This method does not model a particular biological mechanism, but seems to work better in practice. On the other hand, novelty is known to modulate plasticity through a number of mechanisms (Hamid et al., 2016; Rangel-Gomez and Meeter, 2016). Therefore, a decrease in learning rate after familiarization with the task is reasonable.

On a functional scale, deep learning methods still outperform biologically plausible learning rules such as SPORE. For future work, the performance gap between SPORE and deep learning methods should be tackled by taking inspiration from deep learning methods. Specifically, the online learning method inherent to SPORE is impacted by the high variance of the policy evaluation. This problem was alleviated in policy-gradient methods by introducing a critic trained to estimate the expected return of a given state. This expected return is used as a baseline which reduces the variance of the policy evaluation. Decreasing the variance could also be achieved by considering an action-space noise as in Daucé (2009) instead of a parameter-space noise implemented by the Wiener process in . Lastly, an automatic mechanism to regulate the learning rate β is beneficial for more complex task. Such a mechanism could be inspired by trust-region methods (Schulman et al., 2015), which constrains weight updates to alter the policy little by little. These improvements should increase SPORE performance so that more complex tasks such as multi-joint effector control and discrete terminal rewards—supported by design by the proposed framework—could be considered.

REFERENCES

- Bellec, G., Kappel, D., Maass, W., and Legenstein, R. (2017). Deep rewiring: training very sparse deep networks. *arXiv preprint arXiv:1711.05136*.
- Bellec, G., Salaj, D., Subramoney, A., Legenstein, R., and Maass, W. (2018). “Long short-term memory and learning-to-learn in networks of spiking neurons,” in *Conference on Neural Information Processing Systems (NIPS)* (Montreal, QC).
- Bengio, Y., Lee, D.-H., Bornschein, J., Mesnard, T., and Lin, Z. (2015). Towards biologically plausible deep learning. *arXiv preprint arXiv:1502.04156*.
- Bing, Z., Meschede, C., Huang, K., Chen, G., Rohrbein, F., Akl, M., et al. (2018a). “End to end learning of spiking neural network based on r-stdp for a lane keeping vehicle,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (Brisbane, QLD: IEEE), 1–8.
- Bing, Z., Meschede, C., Röhrbein, F., Huang, K., and Knoll, A. C. (2018b). A survey of robotics control based on learning-inspired spiking neural networks. *Front. Neurobot.* 12:35. doi: 10.3389/fnbot.2018.00035
- Daucé, E. (2009). “A model of neuronal specialization using hebbian policy-gradient with “slow” noise,” in *International Conference on Artificial Neural Networks* (Limassol: Springer), 218–228.
- Djurfeldt, M., Hjorth, J., Eppler, J. M., Dudani, N., Helias, M., Potjans, T. C., et al. (2010). Run-time interoperability between neuronal network simulators based on the MUSIC framework. *Neuroinformatics* 8, 43–60. doi: 10.1007/s12021-010-9064-z
- Ekeberg, O., and Djurfeldt, M. (2008). MUSIC – multisimulation coordinator: request for comments. *Nature Precedings*. doi: 10.1038/npre.2008.1830.1
- Falotico, E., Vannucci, L., Ambrosano, A., Albanese, U., Ulbrich, S., Vasquez Tieck, J. C., et al. (2017). Connecting artificial brains to robots in a comprehensive

AUTHOR CONTRIBUTIONS

All the authors participated in writing the paper. JK, MH, AK, JV, and DK conceived the experiments and analyzed the data.

FUNDING

This research has received funding from the European Union’s Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreement No. 720270 (Human Brain Project SGA1) and No. 785907 (Human Brain Project SGA2), as well as a fellowship within the FITweltweit programme of the German Academic Exchange Service (DAAD) (MH). In addition, this work was supported by the H2020-FETPROACT project Plan4Act (#732266) (DK).

ACKNOWLEDGMENTS

The collaboration between the different institutes that led to the results reported in the present paper was carried out under CoDesign Project 5 (CDP5—Biological Deep Learning) of the Human Brain Project.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fnbot.2019.00081/full#supplementary-material>

- simulation framework: the neurobotics platform. *Front. Neurobot.* 11:2. doi: 10.3389/fnbot.2017.00002
- Florian, R. V. (2007). Reinforcement learning through modulation of spike-timing-dependent synaptic plasticity. *Neural Comput.* 19, 1468–1502. doi: 10.1162/neco.2007.19.6.1468
- Frey, U., and Morris, R. G. (1997). Synaptic tagging and long-term potentiation. *Nature* 385, 533–536. doi: 10.1038/385533a0
- Gewaltig, M.-O., and Diesmann, M. (2007). Nest (neural simulation tool). *Scholarpedia* 2:1430. doi: 10.4249/scholarpedia.1430
- Gilra, A., and Gerstner, W. (2017). Predicting non-linear dynamics by stable local learning in a recurrent spiking neural network. *Elife* 6:e28295. doi: 10.7554/eLife.28295
- Gilra, A., and Gerstner, W. (2018). “Non-linear motor control by local learning in spiking neural networks,” in *Proceedings of the 35th International Conference on Machine Learning* Vol. 80 of *Proceedings of Machine Learning Research*, eds J. Dy and A. Krause (Stockholm: Stockholmssmässan; PMLR), 1773–1782.
- Hamid, A. A., Pettibone, J. R., Mabrouk, O. S., Hetrick, V. L., Schmidt, R., Vander Weele, C. M., et al. (2016). Mesolimbic dopamine signals the value of work. *Nat. Neurosci.* 19:117. doi: 10.1038/nn.4173
- Izhikevich, E. M. (2007). Solving the distal reward problem through linkage of stdp and dopamine signaling. *Cereb. Cortex* 17, 2443–2452. doi: 10.1093/cercor/bhl152
- Kaiser, J., Mostafa, H., and Neftci, E. (2018). Synaptic plasticity dynamics for deep continuous local learning. *arXiv preprint arXiv:1811.10766*.
- Kaiser, J., Tieck, J. C. V., Hubschneider, C., Wolf, P., Weber, M., Hoff, M., et al. (2016). “Towards a framework for end-to-end control of a simulated vehicle with spiking neural networks,” in *2016 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAP)* (San Francisco, CA: IEEE), 127–134.

- Kappel, D., Habenschuss, S., Legenstein, R., and Maass, W. (2015). Network plasticity as Bayesian inference. *PLOS Comput. Biol.* 11:e1004485. doi: 10.1371/journal.pcbi.1004485
- Kappel, D., Hoff, M., and Subramoney, A. (2017). *IGITUGraz/Spore-Nest-Module: SPORE version 2.14.0*. doi: 10.5281/zenodo.1043486
- Kappel, D., Legenstein, R., Habenschuss, S., Hsieh, M., and Maass, W. (2018). A dynamic connectome supports the emergence of stable computational function of neural circuits through reward-based learning. *Eneuro*. doi: 10.1523/ENEURO.0301-17.2018. [Epub ahead of print].
- Kappel, D., Nessler, B., and Maass, W. (2014). STDP installs in winner-take-all circuits an online approximation to hidden Markov model learning. *PLoS Comput. Biol.* 10:e1003511. doi: 10.1371/journal.pcbi.1003511
- Kingma, D. P., and Ba, J. (2014). Adam: a method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Koenig, N., and Howard, A. (2004). "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No. 04CH37566)*, Vol. 3 (IEEE), 2149–2154.
- Kruger, N., Janssen, P., Kalkan, S., Lappe, M., Leonardis, A., Piater, J., et al. (2013). Deep hierarchies in the primate visual cortex: what can we learn for computer vision? *IEEE Trans. Softw. Eng.* 35, 1847–1871. doi: 10.1109/TPAMI.2012.272
- Legenstein, R., Pecevski, D., and Maass, W. (2008). A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback. *PLOS Comput. Biol.* 4:e1000180. doi: 10.1371/journal.pcbi.1000180
- Lichtsteiner, P., Posch, C., and Delbruck, T. (2008). A 128-128 120 dB 15-s latency asynchronous temporal contrast vision sensor. *IEEE J. Solid State Circuits* 43,566–576. doi: 10.1109/JSSC.2007.914337
- Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., et al. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.
- Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., et al. (2016). "Asynchronous methods for deep reinforcement learning," in *International Conference on Machine Learning* (New York, NY), 1928–1937.
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., et al. (2015). Human-level control through deep reinforcement learning. *Nature* 518:529. doi: 10.1038/nature14236
- Nakano, T., Otsuka, M., Yoshimoto, J., and Doya, K. (2015). A spiking neural network model of model-free reinforcement learning with high-dimensional sensory input and perceptual ambiguity. *PLoS ONE* 10:e0115620. doi: 10.1371/journal.pone.0115620
- Neftci, E. (2017). "Stochastic synapses as resource for efficient deep learning machines," in *Electron Devices Meeting (IEDM), 2017 IEEE International* (San Francisco, CA: IEEE).
- Otsuka, M., Yoshimoto, J., and Doya, K. (2010). "Free-energy-based reinforcement learning in a partially observable environment," in *ESANN* (Bruges).
- Pan, W.-X., Schmidt, R., Wickens, J. R., and Hyland, B. I. (2005). Dopamine cells respond to predicted events during classical conditioning: evidence for eligibility traces in the reward-learning network. *J. Neurosci.* 25, 6235–6242. doi: 10.1523/JNEUROSCI.1478-05.2005
- Pfister, J.-P., Toyozumi, T., Barber, D., and Gerstner, W. (2006). Optimal spike-timing-dependent plasticity for precise action potential firing in supervised learning. *Neural Comput.* 18, 1318–1348. doi: 10.1162/neco.2006.18.6.1318
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., et al. (2009). "Ros: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, Vol. 3 (Kobe), 5.
- Rangel-Gomez, M., and Meeter, M. (2016). Neurotransmitters and novelty: a systematic review. *J. Psychopharmacol.* 30, 3–12. doi: 10.1177/0269881115612238
- Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. (2015). "Trust region policy optimization," in *International Conference on Machine Learning* (Lille), 1889–1897.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- Tieck, J. C. V., Pogančić, M. V., Kaiser, J., Roennau, A., Gewaltig, M.-O., and Dillmann, R. (2018). "Learning continuous muscle control for a multi-joint arm by extending proximal policy optimization with a liquid state machine," in *International Conference on Artificial Neural Networks* (Rhodes: Springer), 211–221.
- Urbanczik, R., and Senn, W. (2014). Learning by the dendritic prediction of somatic spiking. *Neuron* 81, 521–528. doi: 10.1016/j.neuron.2013.11.030
- Weidel, P., Djurfeldt, M., Duarte, R. C., and Morrison, A. (2016). Closed loop interactions between spiking neural network and robotic simulators based on MUSIC and ROS. *Front. Neuroinform.* 10:31. doi: 10.3389/fninf.2016.00031
- Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.* 8, 229–256. doi: 10.1007/BF00992696
- Wolf, P., Hubschneider, C., Weber, M., Bauer, A., Härtl, J., Dürr, F., et al. (2017). "Learning how to drive in a real world simulation with deep q-networks," in *2017 IEEE Intelligent Vehicles Symposium (IV)* (Redondo Beach, CA), 244–250.
- Yu, Z., Kappel, D., Legenstein, R., Song, S., Chen, F., and Maass, W. (2016). Camkii activation supports reward-based neural network optimization through hamiltonian sampling. *arXiv preprint arXiv:1606.00157*.
- Zenke, F., and Ganguli, S. (2018). Superspike: supervised learning in multilayer spiking neural networks. *Neural Comput.* 30, 1514–1541. doi: 10.1162/neco_a_01086

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Kaiser, Hoff, Konle, Vasquez Tieck, Kappel, Reichard, Subramoney, Legenstein, Roennau, Maass and Dillmann. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.



Autonomous Sequence Generation for a Neural Dynamic Robot: Scene Perception, Serial Order, and Object-Oriented Movement

Jan Tekülve^{1*}, Adrien Fois², Yulia Sandamirskaya³ and Gregor Schöner¹

¹ Institute for Neural Computation, Ruhr-University Bochum, Bochum, Germany, ² Lorraine Research Laboratory in Computer Science and its Applications, Vandœuvre-lès-Nancy, France, ³ Institute for Neuroinformatics, University of Zürich and ETZ Zürich, Zurich, Switzerland

OPEN ACCESS

Edited by:

Florian Röhrbein,
Technical University of Munich,
Germany

Reviewed by:

Nicolas Cuperlier,
Université de Cergy-Pontoise, France
Zhenshan Bing,
Technical University of Munich,
Germany
Dietmar Heinke,
University of Birmingham,
United Kingdom

*Correspondence:

Jan Tekülve
jan.tekuelve@ini.rub.de

Received: 16 April 2019

Accepted: 28 October 2019

Published: 15 November 2019

Citation:

Tekülve J, Fois A, Sandamirskaya Y and Schöner G (2019) Autonomous Sequence Generation for a Neural Dynamic Robot: Scene Perception, Serial Order, and Object-Oriented Movement. *Front. Neurobot.* 13:95. doi: 10.3389/fnbot.2019.00095

Neurally inspired robotics already has a long history that includes reactive systems emulating reflexes, neural oscillators to generate movement patterns, and neural networks as trainable filters for high-dimensional sensory information. Neural inspiration has been less successful at the level of cognition. Decision-making, planning, building and using memories, for instance, are more often addressed in terms of computational algorithms than through neural process models. To move neural process models beyond reactive behavior toward cognition, the capacity to autonomously generate sequences of processing steps is critical. We review a potential solution to this problem that is based on strongly recurrent neural networks described as neural dynamic systems. Their stable states perform elementary motor or cognitive functions while coupled to sensory inputs. The state of the neural dynamics transitions to a new motor or cognitive function when a previously stable neural state becomes unstable. Only when a neural robotic system is capable of acting autonomously does it become useful to a human user. We demonstrate how a neural dynamic architecture that supports autonomous sequence generation can engage in such interaction. A human user presents colored objects to the robot in a particular order, thus defining a serial order of color concepts. The user then exposes the system to a visual scene that contains the colored objects in a new spatial arrangement. The robot autonomously builds a scene representation by sequentially bringing objects into the attentional foreground. Scene memory updates if the scene changes. The robot performs visual search and then reaches for the objects in the instructed serial order. In doing so, the robot generalizes across time and space, is capable of waiting when an element is missing, and updates its action plans online when the scene changes. The entire flow of behavior emerges from a time-continuous neural dynamics without any controlling or supervisory algorithm.

Keywords: neural dynamic modeling, autonomous robot, sequence generation, scene perception, reaching movement

1. INTRODUCTION

Neurally inspired robotics already has a long history. To position our work in this history and review our conceptual commitments, we discuss three strands of neurally inspired robotics.

1.1. Reactive Behaviors

One strand goes back to Grey's electronic turtle (Grey, 1950) and Braitenberg's thought experiments on vehicles (Braitenberg, 1984). This line of work reached maturity in behavior-based robotics (Brooks, 1991; Mataric, 1998) in which flexibility emerges from the coordination of elementary behaviors, each establishing a direct link from sensory inputs to actuators, in the manner of reflex loops. This is particularly suited to conceptual "vehicles," robotic systems in which the sensors are mounted on the moving actuator. This enables closed loop situations that greatly reduce the demands on representation and abstraction. For instance, a visual sensor mounted in a robot hand makes it possible to achieve reaching by visual servoing without an explicit representation of objects in the world (Ruf and Horaud, 1999).

By organizing closed action-perception loops in architectures, most famously the subsumption architecture (Brooks, 1986), this form of reactive robotics may generate behaviors of a certain complexity (Proetzsch et al., 2010). The behavior is generated *autonomously* in the sense that sensory information from a structured environment may trigger the activation of elementary behaviors, which may lead to chains of activation and deactivation events through the architecture, inducing sequences of behavioral decisions, without the need for an explicit internal plan, schedule, or program. The organization of such behaviors is implicitly encoded in the architecture itself.

Avoiding representation and abstraction is a feature of the approach (Brooks, 1990), but also points to a limitation of this line of neurally inspired robotics: Behavior-based robots are not very good at cognition. Minimally, cognition is engaged when the link between sensing and acting becomes less direct. Building and exploiting memory is an example (Engels and Schöner, 1995). So when an action is based on sensory information that is no longer directly available on the sensory surface at the time the action unfolds, relevant information must be represented in memory. Memories are useful only if they are represented in a form in which they remain invariant under changes the system experiences between the acquisition of the memory and its use. For instance, the memory representation of a movement target for a vehicle needs to be invariant under rotation of the vehicle (Bicho et al., 2000). A more demanding form of cognition is the capacity to perceive sequences of events and store them in a memory for serial order so that a sequence with a matching serial order can then be acted out (such as hearing a phone number and then dialing it). Again, the information needs to abstract from the sensor data to be useful for the required actions.

Our approach is historically based on behavior-based thinking, which we extended by adding neural memory representations and neural mechanism of decision making (Schöner et al., 1995). Here we will study how memory

for serial order can be built and used to act sequentially in new environments.

1.2. Neuronal Oscillators and Pattern Generators

A second strand of neurally inspired robotics is based on the idea that neural oscillators may generate rhythmic movement patterns. That idea has been used to generate legged locomotion in biologically inspired robots (Holmes et al., 2006; Ijspeert, 2008). Such neural oscillator ideas can be integrated with the dynamics of limbs and muscles and their interaction with the ground, enabling stable locomotion patterns (Full and Koditschek, 1999; Ghigliazza et al., 2003). Neural oscillators are one important class of neural networks in which recurrent connections are strong enough to induce endogenous patterns of neural activation that are not mere transformations of input. That class can be extended to neural timers that generate complex temporal patterns that may be the basis for certain motor skills (Buonomano and Laje, 2010). Coupling neural oscillators provides an account for coordination (Schöner and Kelso, 1988) and adaptation enables the modulation of rhythmic movement patterns (Aoi et al., 2017).

Typically, however, these kinds of models do not address how movement may be directed at targets in the world, such as when reaching for an object or intercepting a ball. A related class of neural models going back, perhaps, to Bullock and Grossberg (1988), generates time courses by integrating neural activity toward an end-point that may ultimately be determined by perceptual processes. This is the basis of the notion of dynamic movement primitives (Schaal et al., 2003), which is still broadly neurally inspired although it is typically implemented in a mathematical form that does not explicitly reference neural processing principles (see Ijspeert et al., 2013 for an excellent review). The dynamical systems framework for reaching toward objects can address how such movement is directed at objects in the world (Hersch and Billard, 2008). Typically, however, the representation of the object's pose and kinematic state remains clearly outside the neural metaphor (while achieving superhuman performance in skills such as catching, Kim et al., 2014).

Our approach builds on this tradition of using neural oscillators for timing. We generate individual goal-directed reaches from an active transient solution of a recurrent neural dynamics. We extend this tradition by providing a neural dynamic architecture that obtains from the visual array a neural representation of the targets of a reaching movement. This requires that an object's visual coordinates are transformed into coordinates anchored in the initial position of the hand (Schöner et al., 2019). We show how such a neural representation of movement targets may be linked to the visual array, enabling online updating of movement generation when the scene changes (see Knips et al., 2017 for an earlier version of such online updating).

1.3. Neural Networks for Perception

A third strand of neural inspiration for embodied cognitive systems is, of course, the use of neural networks to extract

relevant information about the environment from sensory (e.g., image, sound) data (Kriegeskorte, 2015). This strand is currently undergoing explosive growth as the scaling of deep neural networks in size and learning examples enables superhuman performance in certain classification and detection tasks (Lecun et al., 2015; Schmidhuber, 2015). These neural networks essentially serve as intelligent filters of sensory information, a critical function when robot cognition is to be linked to the world.

While these networks by themselves do not perform cognitive functions, they may provide outputs that enable cognition. For instance, networks may deliver labels for a relational description of a visual scene (e.g., Kelleher and Dobnik, 2017). In most cases, the actual reasoning about spatial or other relations is, however, performed outside a neural processes model, based on algorithms and probabilistic inference. First steps are being made, however, toward such models generating the sequential attentional selection on which human visual cognition is centrally based (Ba et al., 2015).

Our approach is based on the classical notion of feature extraction along the visual pathway, the simplest step in these kinds of systems (e.g., Serre et al., 2007). As we do not address object recognition, we limit ourselves to very simple features here (see Lomp et al., 2017 for how the approach may link to object recognition). Instead, we demonstrate how a neural dynamic system may autonomously generate the sequence of attentional selections to build a visual scene memory that is intermittently coupled to the visual array, and thus is sensitive to change and capable of updating in response to such change.

1.4. Goals

In this paper, we integrate these three strands of neurally inspired robotics which requires us to extend each of them. Our emphasis is on how the integrated system—essentially a network of neural dynamic populations—is continuously or intermittently coupled to sensory information, while at the same time being capable of autonomously generating sequences of decisions, actions, and events. Neural activation is thus generated endogenously in this system, while retaining the coupling to the sensory surfaces.

The system addressed four key elements of grounded cognition: (1) It autonomously builds scene memory, a neural map of locations and feature values bound to those locations. Different objects are sequentially brought into the attentional foreground, in each case creating an entry into scene memory, which can be updated if change is detected. (2) The system learns the serial order of events that occur in its visual array. Each time an attended object changes, the system registers the transition and learns the new feature value as associated with its serial position. This provides a possible interface through which a human user can interact with the system. (3) The system generates a sequence of actions oriented at objects in the world in the learned serial order. At any point in the sequence, this entails finding an object in the visual surrounding that matches the feature values currently sought, generating the action, and then transitioning to the next sub-task within the learned sequence. This exemplifies the capacity of the system to autonomously generate organized behavior that is not merely reactive but reflective of a learned plan. (4) Each action consists of a pointing

gesture oriented at an attended object. The action is initiated once the object has been brought into the attentional foreground, but may be updated any time if object shifts to a new location. This is a minimal instantiation of object-oriented action that any form of cognitive robotics must be capable of.

Key to this demonstration is the notion of neural dynamics, in which strongly recurrent neural networks, approximated as spatially and temporally continuous neural fields, evolve primarily under the influence of their internal interaction that sets up attractor states. Inputs induce instabilities that bring about switches of neural states from which sequences of cognitive or motor states emerge. Such neural dynamics are capable of making decisions, building working memories, and organizing sequential transitions (Schöner et al., 2016). Because their neural states are stable, neural fields retain their functional properties when they are coupled to other fields. Fields may thus serve as building blocks of networks of fields, which could be thought of as neural dynamic architectures. These networks may be coupled to sensory inputs, while evolving under their own, endogenous dynamics, resolving the tension between reactive and cognitive systems.

To make the ideas accessible, we restrict the demonstration to a very simple scenario. A robot observes a table top on which a human user places and removes colored objects in a particular serial order. The user then builds a new visual scene, that includes the objects with colors contained in the taught series. The robot points at these objects in the order defined by the human teacher. When an object of the next required color is not available, the system waits until such a color is presented. When the visual array changes, the robot updates its reaching plans. This may happen online if the change occurs while the robot is already attempting to point at the object. All action and observation run autonomously in neural dynamics. There is no control algorithm outside the neural dynamics. See **Supplementary Videos 1 and 2** for exemplary demonstrations of teaching and executing the series.

2. DYNAMIC FIELD THEORY

We use Dynamic Field Theory (DFT) (Schöner et al., 2016) as a conceptual framework. DFT provides neural process accounts for elementary cognitive functions such as decision making, memory creation, or the generation of sequences. The core elements of DFT are neural populations which may generate activation patterns that are not primarily dictated by input. This is based on structured and strong recurrent connectivity within the population. Excitatory recurrent connectivity enables detection decisions in which neural activation is induced by input, but then stabilized against decay even as input may weaken again. The initial detection occurs through an instability, in which the resting state becomes unstable. The detection is reversed when the activated state becomes unstable, typically at a lower level of input than needed for initial detection. If the excitatory recurrency is sufficiently strong, the reverse detection instability does not happen, leading to activation that is sustained even when the inducing input is removed entirely. This is the basis for working memory.

Inhibitory recurrent connectivity enables selection decisions, in which one sub-population becomes activated even if multiple sub-populations receive supra-threshold input. Such selection decisions are also stabilized so that the selection of a sub-population may persist even as inputs to other sub-populations become stronger (up to a limit, when the selection instability is encountered). So even though neural populations may be driven by input, they may realize non-unique mappings from input to activation states based on their activation history.

When different populations are coupled, they may induce these kinds of instabilities in each other. This is the basis for generating sequences of neural activation states. When the coupling occurs between excitatory and inhibitory sub-populations, the instabilities may trigger active transients, well-defined time courses of neural activation from temporally unstructured input. Neural oscillations are another possible dynamic regime.

Through their connectivity to sensory or motor surfaces, neural populations may effectively represent continuous feature dimensions, \mathbf{x} . This leads to the notion of neural dynamic fields, $u(\mathbf{x})$. We employ a particular mathematical formalization of the dynamics of such neural populations that goes back to Amari (1977),

$$\tau \dot{u}(\mathbf{x}) = -u(\mathbf{x}) + h + s + \int \sigma(u(\mathbf{x}')) \omega(\mathbf{x} - \mathbf{x}') d\mathbf{x}', \quad (1)$$

where τ describes the field's relaxation time, $h < 0$ the field's resting level, s the sum of input stimuli, and ω the field's interaction kernel that defines the pattern of recurrent connectivity within the field. Only sufficiently activated field locations contribute to interaction or project onto other fields, as formalized by the sigmoidal non-linearity, $\sigma(u)$. Thus, one may think of the activation variable, u , as something like a population-level membrane potential that reflects how close neurons in the population are to the firing threshold [other formalizations use the firing rate as a population variable, see Wilson and Cowan (1973)]. In the meantime, there is a large literature on the mathematics of such fields (Coombes et al., 2014).

The kernel, ω , combines short-range excitatory coupling with long-range inhibitory coupling. This leads to localized peaks of

activation as the activation states that emerge from the instability of the resting state when localized input reaches a threshold (Figure 1). These peaks are the units of representation in DFT that specify through their locations particular values along the represented dimension.

Fields may represent low-dimensional metric spaces. When their dimensionality grows, the binding problem arises and can be solved, see Chapter 5 of Schöner et al. (2016). A limit case are zero-dimensional fields which can be thought of as populations of neurons that represent categorical states. These may arise from larger populations through inhomogeneities in the input or output connectivity. We sometimes call such zero-dimensional fields neural dynamic nodes and model them by single activation variables, $u(t)$, subject to a neural dynamics analogous to Equation (1).

When fields of different dimensionality are coupled, new functions emerge (Zibner et al., 2011, see also Chapter 9 of Schöner et al., 2016). In projecting from a higher to a lower dimensional field, certain dimensions may be marginalized, which effectively probes for the existence of a peak anywhere along the marginalized dimensions. In projecting from a lower to a higher dimensional field, a boost may be given to a subspace, enabling locations within the subspace to reach the detection instability. This is the basis for visual search. The control of peak formation in a field through homogeneous boosting of its activation level is a mechanism of control that may effectively gate particular projections by enabling or disabling peak formation. This mechanism is also central to sequence generation through the condition of satisfaction (CoS) (Sandamirskaya and Schöner, 2010) that will play a central role in our sequence representation model. The neural representation of the CoS is a neural field or a neural node that is pre-activated by the currently active behavior. That behavior predicts the sensory or internal state that will indicate its successful completion. When a signal matching that prediction is received from sensory inputs or from other neural processes, the CoS goes through a detection instability. It then inhibits the current behavior in a reverse detection instability and enables the activation of a new behavior (Sandamirskaya and Schöner, 2010).

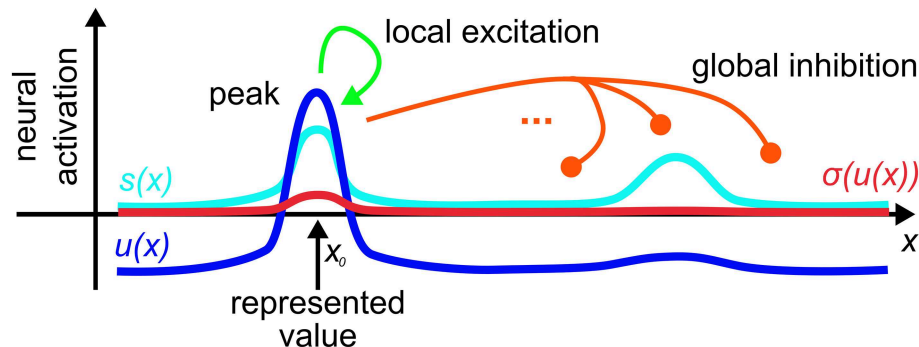


FIGURE 1 | A dynamic neural field spanning a metric dimension, x , represents a specific value, x_0 , along that dimension through a supra-threshold activation peak that is stabilized by local excitatory and global inhibitory interactions.

3. MODEL

The neural dynamic architecture described here is a network of neural fields that are coupled to a camera and a robotic arm. These links enable online connection to a changing visual scene and online control of the arm. Three sub-networks (**Figure 2**) autonomously organize sequences of activation states to build visual representations, learn or perform serially ordered sequences, and generate object-oriented movements.

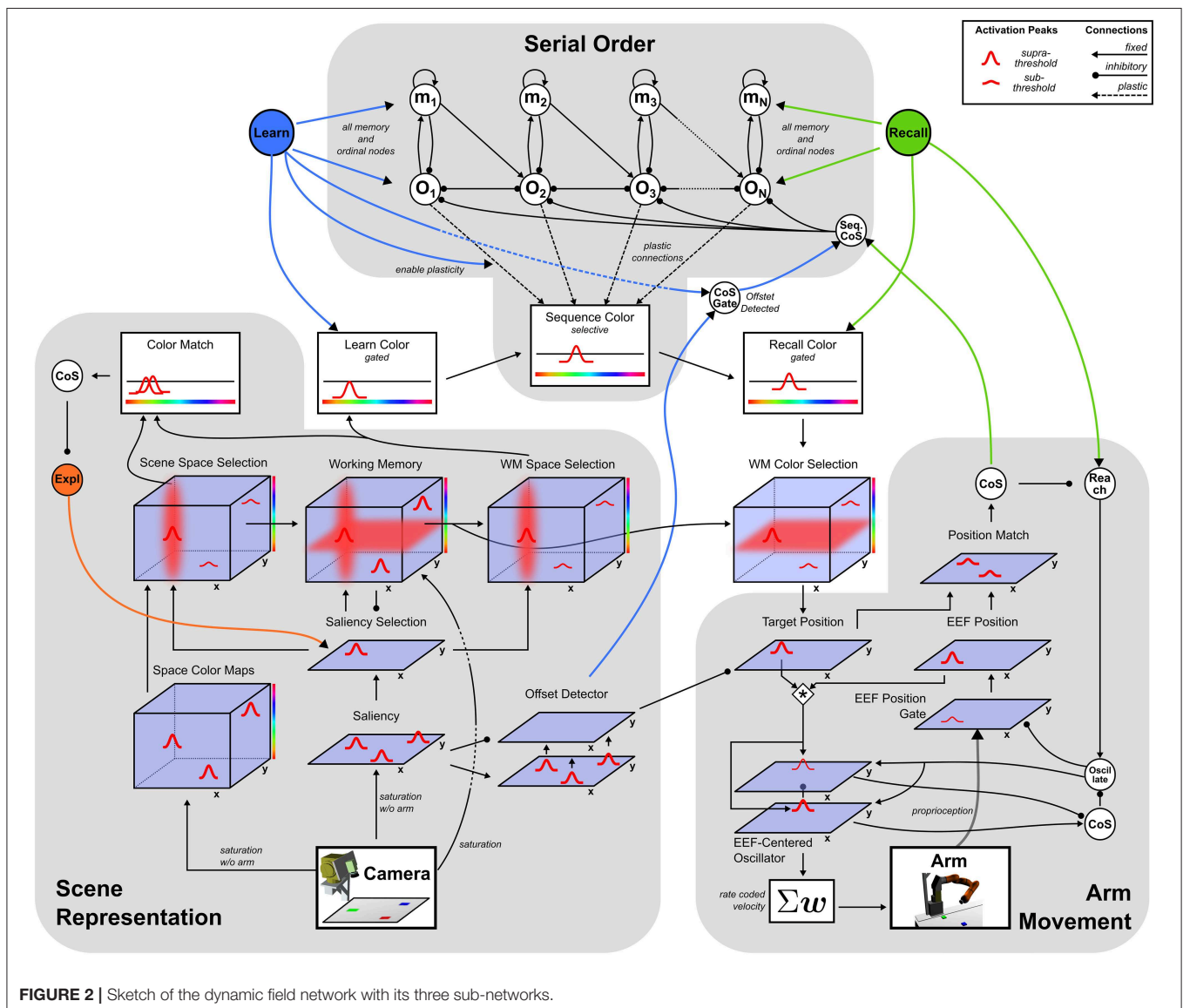
The perceptual sub-network, connected to the camera, creates a working memory representation of the visual scene through autonomous shifts of attention. A motor sub-network drives an oscillator generating velocity commands for the robotic arm. The cognitive sub-network represents ordinal positions in a sequence and may autonomously shift from one ordinal position to the next. The ordinal system may be used in two different manners, sequence learning and sequence recall,

controlled by the activation of one of two different task nodes. These task nodes activate behaviors by boosting fields' resting levels and enabling fields to generate task relevant attractor states.

The following sections describe for each sub-network the states that drive behavior and the mechanism for how the system switches between those states. The last section addresses the integration of all three sub-networks for the two tasks *Learn* and *Recall*.

3.1. Perception: Scene Representation

The scene representation sub-network is based on Grieben et al. (2018) and creates three-dimensional (2D space and 1D color) working memory representations of objects in the visual scene captured by the camera. Each entry into the representation is created sequentially as the sub-network autonomously shifts attention across different objects in the scene.



The network's attention is modeled through peaks of activation in the two-dimensional *Saliency Selection* field that arise at salient locations in the scene. These locations are represented in the *Saliency* field which receives input directly from the camera. Based on their distinctive colors, the table and the robot's own arm are subtracted from the image in a pre-processing step. The saturation channel of the resulting HSV-image serves as input amplitude at each location.

Combined with a homogeneous boost of its resting level from the *Exploration* intention node, activation in the *Saliency* field is sufficient to create a single peak in the *Saliency Selection* field. Attentional shifts occur whenever the *Exploration* node deactivates and subsequently reactivates, causing a destabilization of the present peak in the *Selection* field followed by the emergence of a new peak at a new location. Previously unattended locations are more likely to be selected because inhibitory influence from the working memory gives them a competitive advantage.

The activation of the *Saliency Selection* field, u_{sel} , is governed by the following neural dynamics:

$$\begin{aligned} \tau \dot{u}_{\text{sel}}(x, y) = & -u_{\text{sel}}(x, y) + h_{\text{sel}} + w_{\text{exp}}\sigma(u_{\text{exp}}) \\ & + w_{\text{sal}}\sigma(u_{\text{sal}}(x, y)) - w_{\text{mem}} \int \sigma(u_{\text{mem}}(x, y, c))dc \\ & + \int \sigma(u_{\text{sel}}(x', y'))\omega_{\text{sel}}(x - x', y - y')dx'dy', \end{aligned} \quad (2)$$

where h_{sel} describes the field's resting level, u_{exp} the homogeneous boost activation from the *Explore* intention node, $u_{\text{sal}}(x, y)$ the activation of the *Saliency* field, $\int \sigma(u_{\text{mem}}(x, y, c))dc$ the activation of the *Working Memory* projected onto the two spatial dimensions, x and y , and ω_{sel} the field's selective lateral interaction kernel. Each input, σu_{in} , to the field is weighted by a specific weight, w_{in} . The same notation is used in all following equations and the concrete parameter values can be found in the **Appendix**.

The currently attended location achieves spatial feature binding: It is forwarded to the three 3D space-color fields, *Scene Space Selection*, *Working Memory*, and *WM Space Selection*, ensuring that the color features represented in those fields originate from the same location. The *Scene Space Selection* field combines sub-threshold activation from the *Space Color Maps* field, that represents color information in the scene, with spatial sub-threshold activation from the *Saliency Selection* field. Together, these inputs induce a single peak in three dimensions that represents the attended spatial location and the color perceived at that location.

That color is extracted and combined with spatial information from the *Saliency Selection* field to add a peak in the 3D-*Working Memory* field.¹ The *Working Memory* field receives additional input directly from the camera image that includes the robot's arm. That input is proportional to the saturation channel

from which the table color saturation has been subtracted. This mask, seen, for instance, in the *Working Memory* row of **Figure 4**, makes it possible to sustain peaks of activation anywhere where the camera picks up visual structure. Peaks representing objects can thus remain stable in working memory when they become occluded by the robot's arm, but are removed from working memory when they disappear from the scene at any other location.

The activation, $u_{\text{mem}}(x, y, c)$, of the *Working Memory* field is governed by the following dynamics:

$$\begin{aligned} \tau \dot{u}_{\text{mem}}(x, y, c) = & -u_{\text{mem}}(x, y, c) + h_{\text{mem}} + w_{\text{sel}}(x, y, c)\sigma(u_{\text{sel}}(x, y)) \\ & + w_{\text{nbc}}(x, y, c)\sigma(i_{\text{nbc}}(x, y)) + w_{\text{ssl}} \int \sigma(u_{\text{ssl}}(x, y, c))dxdy \\ & + \int \sigma(u_{\text{mem}}(x', y', c'))\omega_{\text{mem}}(x - x', y - y', c - c')dx'dy'dc', \end{aligned} \quad (3)$$

where h_{mem} describes the field's resting level, $u_{\text{sel}}(x, y)$ the activation from the *Saliency Selection* field, $i_{\text{nbc}}(x, y)$ the saturation channel from the camera image, $\int \sigma(u_{\text{ssl}}(x, y, c))dxdy$ the activation of the *Scene Space Selection* field projected onto the color dimension, c , and ω_{mem} the field's lateral interaction kernel.

Supra-threshold activation of the *Working Memory* field is forwarded to the *Memory Space Selection* field, which works analogously to the *Scene Space Selection* field and thus forms a single 3D activation peak, representing color and spatial location of the attended location in working memory.

Color information represented in the *Scene* and *Memory Space Selection* fields is forwarded to the *Color Match* field. That field forms a peak only when the input from the scene overlaps in location and color with one of the peaks in the memory field. A peak in the match field thus signals successful entry of an item into the *Working Memory* at the currently attended location. Supra-threshold activation in the match field projects onto the *CoS Explore* node, which in turn inhibits the *Explore* intention node. Deactivation of the *Explore* intention node removes the resting level boost from the *Saliency Selection* field inducing a reverse detection instability that propagates to the *Scene* and *Memory Space Selection* fields, the *Color Match* field and ultimately to the *CoS Explore* node. The newly created peak in the *Working Memory* field is sustained and the *Explore* intention node is released from inhibition enabling attentional selection of a new location.

3.1.1. Offset Detector

The scene representation sub-network is capable of detecting sudden object movement with the help of a two-layer offset detector connected to the *Saliency* field. Both layers, u_{dfa} and u_{dsl} , are two-dimensional fields over image space that are governed by the following dynamics with timescales, $\tau_{\text{dfa}} < \tau_{\text{dsl}}$:

$$\begin{aligned} \tau_{\text{dfa}} \dot{u}_{\text{dfa}}(x, y) = & -u_{\text{dfa}}(x, y) + h_{\text{det}} - w_{\text{sin}}\sigma(u_{\text{sal}}(x, y)) \\ & + w_{\text{dsl}}\sigma(u_{\text{dsl}}(x, y)), \\ \tau_{\text{dsl}} \dot{u}_{\text{dsl}}(x, y) = & -u_{\text{dsl}}(x, y) + h_{\text{det}} + w_{\text{sex}}\sigma(u_{\text{sal}}(x, y)), \end{aligned} \quad (4)$$

¹Spatial information is taken from the *Saliency Selection* field rather than directly from the *Scene Space Selection* field to allow for possible coordinate transforms between image and memory space. In the present scenario, the camera is not moved so that there is no need for a coordinate transform.

where h_{det} describes the common resting level, and $\sigma(u_{\text{sal}}(x, y))$ the thresholded activation of the *Saliency* field which excites the slower layer, u_{dsl} , and inhibits the faster layer, u_{dfa} . Because inhibitory input is stronger than excitatory input ($w_{\text{sin}} > w_{\text{dsl}}$), static visual structure induces supra-threshold activation in the slow layer, u_{dsl} , not in the fast layer, u_{dsl} .

Once an object is removed from the scene, the inhibitory influence, w_{sin} , vanishes faster than the excitatory influence from the slow layer, u_{dsl} , leading to the formation of a peak in u_{dfa} that represents the detection of an object that moves away from the location of the peak.

3.2. Motor: Arm Movement

The sub-network responsible for reaching movements, based on Schöner et al. (2019), autonomously drives an oscillator that creates velocity commands which move a robotic arm to a given target in two-dimensional space. A hierarchy of intention and CoS nodes governs the behavior: The *Reach* intention node activates the *Oscillate* intention node, which initiates an active transient (see **Figure 3**). The *Cos Oscillate* node is activated once the transient reaches a new steady state, while the *CoS Reach* is activated when the representations of target and end-effector position match. Thus, multiple active transients (oscillations) are generated until the arm reaches the represented target.

Target and end-effector (EEF) are both represented as peaks of activation in two-dimensional fields defined over image space, the *Target Position* and the *EEF Position* DNFs, respectively. Activation originating from working memory causes the creation of a peak in the *Target Position* field. Proprioceptive information from the current arm configuration is mapped through a forward kinematics into end-effector space and then transformed from rate to space code inducing a peak in a two-dimensional *EEF Position* field. Target and end-effector representations are cross-correlated with each other to create an end-effector centered representation of the target position. This representation is input into a two-layer field of neural oscillators, u_{exc} and u_{inh} . The faster excitatory layer, u_{exc} , generates an active transient illustrated in **Figure 3**: Its input first drives up excitation, which is then suppressed by inhibition from the slower inhibitory layer, u_{inh} :

$$\begin{aligned}\tau_{\text{exc}} \dot{u}_{\text{exc}}(x, y) &= -u_{\text{exc}}(x, y) + h_{\text{osc}} + w_{\text{cct}}\sigma(u_{\text{cct}}(x, y)) \\ &\quad + w_{\text{osc}}\sigma(u_{\text{osc}}) - w_{\text{inh}}\theta(u_{\text{inh}}(x, y)) \\ \tau_{\text{inh}} \dot{u}_{\text{inh}}(x, y) &= -u_{\text{inh}}(x, y) + h_{\text{osc}} + w_{\text{cct}}\sigma(u_{\text{cct}}(x, y)) \\ &\quad + w_{\text{osc}}\sigma(u_{\text{osc}}),\end{aligned}\quad (5)$$

where $\tau_{\text{exc}} < \tau_{\text{inh}}$ are the different relaxation times, h_{osc} the resting level, $\sigma(u_{\text{cct}}(x, y))$ the end-effector-centered target representation, $\sigma(u_{\text{osc}})$ the homogeneous resting level boost from the *Oscillate* intention node, and θ – a semi-linear threshold function.

The thresholded activation, $\theta(u_{\text{exc}}(x, y))$, is transformed into a rate coded Cartesian velocity vector, \mathbf{v} , using a set of feed-forward weights, $\mathbf{w}_{\text{vel}}(x, y)$:

$$\mathbf{v}(t) = \int \int \mathbf{w}_{\text{vel}}(x, y) \theta(u_{\text{exc}}(x, y, t)) dx dy \quad (6)$$

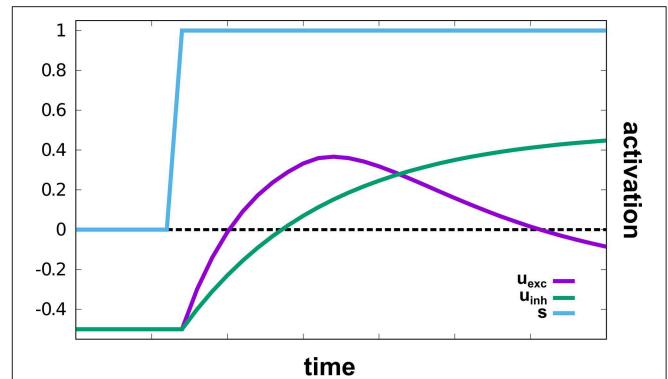


FIGURE 3 | An active transient in u_{exc} generated by a two-layer oscillator in response to an input stimulus, s . The supra-threshold activation level leads to a bell-shaped velocity profile on read-out. This illustration is zero-dimensional while in the model, a two-dimensional field of identical oscillators is used.

The weights, $\mathbf{w}_{\text{vel}}(x, y)$, describe a linear distance function in the end-effector centered representation of the target position. For different movement distances, (x, y) , these weights are tuned such that the arm reaches the target position within a fixed movement time. The velocity vector, \mathbf{v} , is transformed into a joint velocity vector, $\dot{\lambda}$, using the pseudo-inverse of the arm's Jacobian, \mathbf{J}^+ , which depends on the current joint configuration $\lambda(t)$:

$$\dot{\lambda} = \mathbf{J}^+(\lambda(t))\mathbf{v}(t) \quad (7)$$

For more details on the generated velocity profile see Schöner et al. (2019).

While the oscillator is going, its input is not updated, because the connection from proprioception to the *EEF-Position* field is gated by the *Oscillate* intention node. The *EEF-Position* field thus effectively represents the initial position of the hand. Termination of the transient is detected by the *CoS Oscillate* node, which receives excitatory activation from u_{inh} and inhibitory activation from u_{exc} . Activation of *CoS Oscillate* inhibits the *Oscillate* intention node, which resets the oscillator, and releases the *EEF Position Gate* from inhibition so that the end-effector position is updated. When the target representation overlaps sufficiently with the updated EEF Position, a peak forms in the *Position Match* field and activates the *CoS Reach*, which terminates the reach.

3.3. Cognition: Serial Order

The serial order sub-network, based on Sandamirskaya and Schöner (2010), allows for the autonomous storage and recall of a sequence of activation patterns. Each activation pattern is represented through learned inhomogeneous connections between an *ordinal* node and a feature field, here the one-dimensional *Sequence Color* field. Supra-threshold activation in a particular ordinal node thus induces a peak in the *Sequence Color* field that represents the color associated with that particular stage in the sequence.

The sub-network consisting of ordinal nodes, memory nodes and a single CoS node enforces the sequential activation of

ordinal nodes in a fixed order:

$$\begin{aligned}\tau \dot{o}_i &= -o_i + h + w_{o_i, o_i} \sigma(o_i) - w_{o_i, o_j} \sum_{j \neq i} \sigma(o_j) + w_{m_{i-1}, o_i} \sigma(m_{i-1}) \\ &\quad - w_{m_i, o_i} \sigma(m_i) - w_{CoS} \sigma(u_{CoS}) + w_{oh+} \sigma(u_{lrm}) \\ &\quad + w_{oh+} \sigma(u_{rcl}) \\ \tau \dot{m}_i &= -m_i + h + w_{m_i, m_i} \sigma(m_i) + w_{o_i, m_i} \sigma(o_i) + w_{mh+} \sigma(u_{lrm}) \\ &\quad + w_{mh+} \sigma(u_{rcl}).\end{aligned}\quad (8)$$

An active ordinal node, o_i , representing the i th position in the sequence, inhibits all other ordinal nodes, o_j , and activates its own self-sustained memory node, m_i . The memory node pre-activates the next ordinal node, o_{i+1} , through an excitatory connection and inhibits its own ordinal node, o_i , to prevent it from becoming reactivated after completion of the stage. While activated, an ordinal node's self excitation, w_{o_i, o_i} , is sufficient to overcome inhibition from its memory node, w_{m_i, o_i} . An ordinal node remains active until the CoS node, u_{CoS} , is activated and destabilizes all ordinal nodes, which, in turn, removes input from the CoS node that deactivates. The self-sustained memory nodes are unaffected, so that upon release from inhibition by the CoS, the pre-activated ordinal node of the next element in the sequence is activated. Recurring activation and deactivation of the CoS node thus creates a sequence of autonomous transitions between sequence elements in the order of ascending i . Ordinal and memory nodes can become activated only in the presence of an excitatory boost, w_{h+} , from one of the task nodes, *Learn* (u_{lrm}) or, *Recall* (u_{rcl}). Deactivation of an active task node leads to deactivation of all memory and ordinal nodes, effectively resetting the entire system.

Connection weights, $w_{o_i, u_{col}}$, between the active ordinal node, o_i , and the active region in the *Sequence Color* field, u_{col} , are strengthened according to a dynamic version of the Hebbian learning rule:

$$\tau \dot{w}_{o_i, u_{col}}(c) = \eta \sigma(u_{lrm}) \sigma(o_i) (\sigma(u_{col}(c)) - w_{o_i, u_{col}}(c)), \quad (9)$$

where η describes the learning rate and u_{lrm} the activation of the *Learn* task node that gates the learning process.

Before learning, peaks in the *Sequence Color* field arise when a color attended in the Working Memory Selection field is input through the gate field, *Learn color*, u_{lcol} :

$$\begin{aligned}\tau \dot{u}_{col}(c) &= -u_{col}(c) + h + \int \sigma(u_{col}(c')) \omega_{col}(c - c') dc' \\ &\quad + w_{lcol} \sigma(u_{lcol}(c)) + \sum_i w_{o_i, u_{col}}(c) \sigma(o_i),\end{aligned}\quad (10)$$

After learning, peaks in the *Sequence color* field may arise from previously learned connections, $w_{o_i, u_{col}}(c)$, of an ordinal node, o_i . The selective kernel, ω_{col} , ensures that only a single color is represented at all times.

3.4. Task Integration: Learn and Recall

The full network may operate in two different regimes: In the learning regime, a sequence of colors is presented to the system

and learned. In the recall regime, a learned sequence of colors is reproduced by pointing at colored objects in a specific order. Each regime is evoked by the activation of its corresponding task node, *Learn* and *Recall*, which alter the resting level of certain sub-sets of fields.

Both task nodes boost the resting level of all ordinal and memory nodes to allow supra-threshold activation. When task nodes are deactivated, the removal of the corresponding boost causes activation of all self-sustained nodes to decay, effectively resetting the system. This happens, for instance, at the end of the sequence due to activation of the sequence's condition of satisfaction.

The *Learn* node acts as a gate between the Scene Representation and the Serial Order sub-networks. By boosting the *Learn Color* field, the *Learn* node enables that field to form supra-threshold peaks. At which color such a peak is erected is controlled by input from the *Memory Space Selection* field that represents the color at the currently attended location. That color is then imprinted in the connections to the currently active ordinal node through the learning dynamics (Equation 9). The *Learn* node pre-activates the *Offset Detected* node, which connects to the *Sequence CoS*. Thus, whenever a single object is presented in the learning regime, its color is associated with the currently active ordinal node and its removal from the scene causes a transition in which the active ordinal node is replaced by the next ordinal node.

The *Recall* node is a gate between the sequence generation and the arm movement sub-networks. It boosts the *Recall Color* gating field so that the color represented in the *Sequence Color* field is passed on to the three-dimensional *Memory Color Selection field*. If an object in working memory overlaps with that color, a peak forms in the *Memory Color Selection* field. The peak's spatial position is forwarded to the *Target Position* field of the Arm Movement sub-network, which initiates a reaching movement. Once a reach has been successfully performed, the *Reach CoS* is activated, which triggers the *Sequence CoS*, causing the transition to the next ordinal node. In the recall regime, the arm will thus move autonomously to colored objects in the learned order, as long as appropriately colored objects are visible in the scene.

4. RESULTS

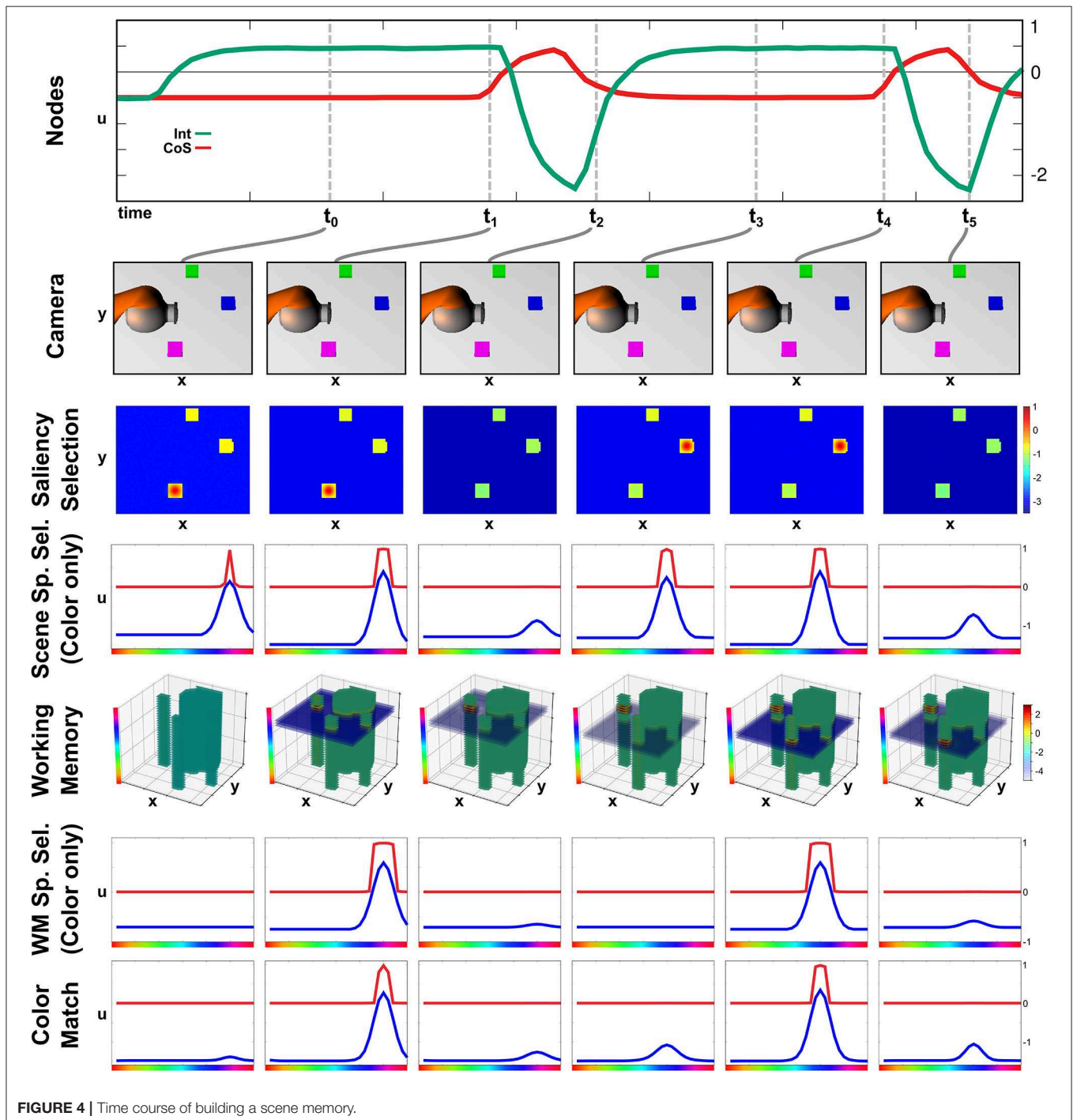
In this section we show how activation within the network unfolds in time during the learn and recall tasks. We visualize relevant activation fields to illustrate how the network's autonomy enables it to cope with variable timing during learning and with changes of the scene during recall.

The network is effectively a large dynamical system. We solved it numerically on digital computers, and that numerical solution was the only form in which algorithms intervened in the system. The numerical implementation of the model made use of CEDAR (Lomp et al., 2016), an open source framework in which DFT models can be graphically assembled and interactively tuned. Cedar can be used to simulate robotic behavior, which was done for the results illustrated in this paper. The visual scene, camera,

and robot arm were simulated using WEBOTS (Michel, 2004) that can be coupled into *Cedar*. The same *Cedar* code can also link to real sensors and robots. We did this, driving the model from a real camera and manipulating the visual scene by placing colored objects on a white table top. We also controlled a lightweight KUKA arm from the same *Cedar* code to verify its capacity to act out the planned movements. These informal robotic experiments are not further documented in this paper.

4.1. Scene Representation: Autonomous Build-up of Visual Working Memory

The build-up of the scene working memory is an ongoing process that provides visual information to the network irrespective of the currently active task node. In **Figure 4** we show activation snapshots of different points in time during working memory build-up in an exemplary scene containing three objects and the arm's end-effector.



At point t_0 , the *Exploration* intention node provides a homogeneous boost to the *Saliency Selection* field leading to an activation peak at the location of the purple object. This causes the emergence of a three-dimensional peak in the *Scene Selection* field, of which the color dimension is shown in the third row. The *Working Memory* field contains no supra-threshold activation yet but, at the locations of the non-background objects, the resting level is increased across the whole color dimension.

Once the peak in the *Scene Selection* field has fully emerged at t_1 , its color component is forwarded as a slice toward the *Working Memory*, where it overlaps with the tube originating from the *Saliency Selection* field and forms a three-dimensional peak. Subsequently a peak also forms in the *Memory Spatial Selection* field, which shares the same color as the peak in the *Scene Space Selection* causing an overlap in the *Color Match* field.

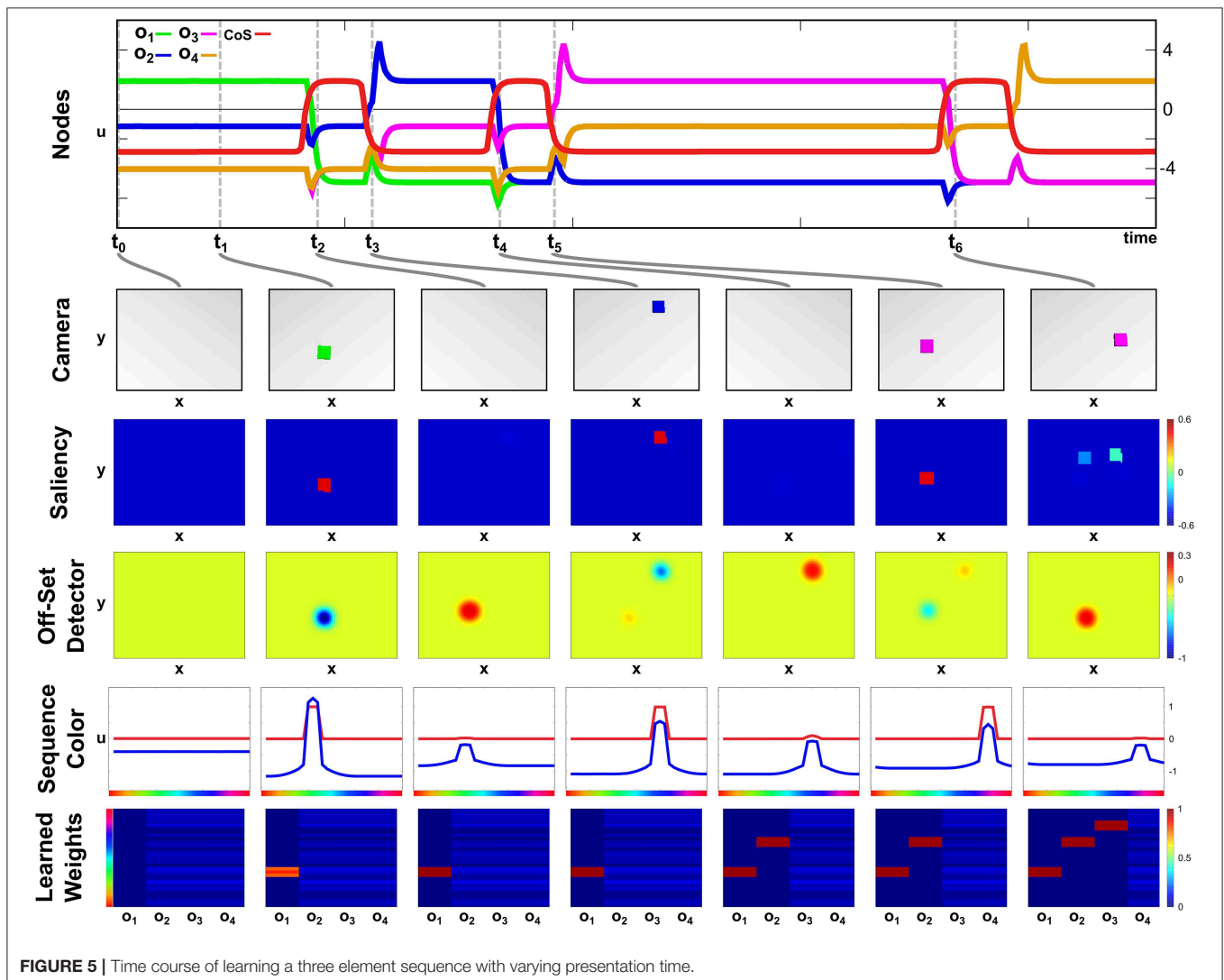
The peak forming in the *Color Match* field activates the *CoS Explore* node, which inhibits the *Explore* intention node. Thus the resting level boost is removed from the *Saliency Selection* field, which subsequently falls down to sub-threshold activation

at point t_2 . Only the self-sustained peak in the *Working Memory* field remains.

The absence of a peak in the *Color Match* field causes the *CoS* node to fall below threshold again, bringing the sub-network to its initial state. The following activation of the *Explore* intention node, depicted from t_3 until t_5 , follows the same temporal activation pattern as the previous one with different feature values for spatial location and color. The spatial location in the *Saliency Selection* field differs due to the inhibitory influence from the *Working Memory* field. See **Supplementary Video 3** for a different example of autonomous build-up of visual working memory in continuous time.

4.2. Learning Demonstration

A particular color sequence is taught to the network in its learning regime by presenting objects of a certain color one after another. In **Figure 5** activation snapshots of some points in time during an exemplary learning episode are shown. The top row depicts the temporal evolution of activation of the ordinal nodes



and the *Sequence CoS* node, while each snapshot column shows the camera image, the activation of the *Saliency* field, activation of the fast layer of the *Offset Detector*, activation of the *Sequence Color* field, and the weight values, $w_{o_i, u_{col}}$, for each ordinal node at one particular point in time.

In the initial phase of the learning at point t_0 no objects are in the scene, but the *Learning* task node has been activated leading to supra-threshold activation in the first ordinal node. All other ordinal nodes are below threshold activity with a slight advantage for o_2 , which already receives an excitatory bias through the active memory node, m_1 .

At t_1 , a green object is inserted into the scene, which forms a peak in the *Saliency* field leading to a localized inhibition in the fast *Offset Detector* field. It is also committed to working memory and leads to the emergence of a peak in the *Sequence Color* field encoding the green color. Due to present supra-threshold activation in the *Sequence Color* field and the ordinal node o_1 , the Hebbian learning rule strengthens weights between the ordinal node and the green color feature values.

The object is removed from the scene at t_2 , which destabilizes the peak in the *Saliency* field removing the inhibition from the fast layer of the *Offset Detector*. The slow layer (not depicted) still carries supra-threshold activation, exciting the fast layer leading to the formation of a peak, which will subsequently activate the *Sequence CoS* node inhibiting all ordinal nodes. This deactivates o_1 and causes the color peak in the *Sequence Color* field to vanish as it is no longer supported by either learned connections nor color input from the scene. The missing input in the scene will also ultimately lead to a decay of activation in the slow *Offset Detector* layer and subsequently cause a reverse-detection instability in the fast layer and the *Sequence CoS* node.

The deactivation of the *Sequence CoS* node is followed by an activation of the next ordinal node o_2 at t_3 . Between t_2 and t_3 a blue object has been added to the scene, whose color is then connected to the freshly activated ordinal node via the Hebbian learning rule. Removal of the object at t_4 triggers the *Offset Detector* and the *CoS* node enabling the activation of the next ordinal node o_3 at t_5 . The presented purple object is kept in the scene for a longer time span than the green or blue one, which does not influence the learning as the transition to the next sequence element at t_6 is based on the removal event rather than timing.

4.3. Recall Demonstration

We demonstrate successful sequence recall through a pointing task, where the network moves the arm to an object in the scene matching the color of the current sequence element. Only a successful reach toward that object allows a progress to the next sequence element. An exemplary recall of three sequence elements is depicted in **Figure 6**, which demonstrates the temporal evolution of the activation of ordinal nodes as well as the field activity of the *Sequence Color*, the *Target Position*, and the *Position Match* field at discrete points during the sequence recall.

At point t_0 , the *Recall* task node has been activated, which lead to the activation of the first ordinal node and the emergence of a peak in the *Sequence Color* field at the green location

due to the learned connections, $w_{o_1, u_{col}}$. The color information converges with the content of the *Working Memory* field in the *Memory Color Selection* field to form a three dimensional peak specifying position and color. Positional information is projected to the *Target Position* field of the Arm Movement sub-network, where it is forwarded to the movement generating oscillator and the *Position Match* field, which compares the current end effector position (center/left) with the current target position (bottom/right).

Due to a successful arm movement both positions match at point t_1 , which is represented through a peak in the *Position Match* field that activates the *Sequence CoS* deactivating the current ordinal node. The *CoS* node itself falls below threshold activity as soon as the peak in the *Position Match* field destabilizes through a missing target representation that vanished through insufficient color input from the *Sequence Color* field.

The missing inhibition from the *CoS* causes an activation of the next ordinal node o_2 , which is associated with blue color. At t_2 however the blue peak has emerged in the *Sequence Color* field, but the target position has not yet been extracted from working memory. The column of point t_3 depicts the end of the movement, where the overlap of end effector and target cause a peak that triggers the *Sequence CoS*. In this particular configuration the match representation is only possible due to the self-sustaining working memory representation that shields the blue object representation from the occlusion through the arm.

The movement toward the purple object depicted from t_4 until t_6 follows an analog activation pattern in which the ordinal node causes the formation of a purple peak in the *Sequence Color* field, which causes an extraction of the target position, leading to movement that terminates due to an represented match of positions. The movement times of all three movements are roughly the same despite their differences in distance, which results from the movement oscillator that enforces the same movement timing for all movements. See **Supplementary Video 4** for another sequence recall demonstration showing the activation development of selected fields in continuous time.

4.3.1. Recall With a Moving Object

The autonomy of all three parts of the field network makes the execution of the recall task robust against unforeseen changes in the scene. We demonstrate this in an exemplary recall episode, where one of the objects in the scene is moved while its color corresponds to the active sequence element. The episode is depicted in **Figure 7**, which shows activation snapshots analog to **Figure 6**. Additionally activation of the *Intention* and *CoS* node driving the two-layer oscillator are shown as well as snapshots of the *Memory Color Selection* field.

In this episode, build-up of the scene memory starts simultaneously with activation of the recall task, which causes a delay between the activation of the first ordinal node and the first movement as the green object, which is the first sequence element, is the second object committed to memory. This can be observed at t_0 in the *Memory Color Selection* field, where the green object forms a peak as it overlaps with the green color slice specified by the sequence color, while the purple object is present

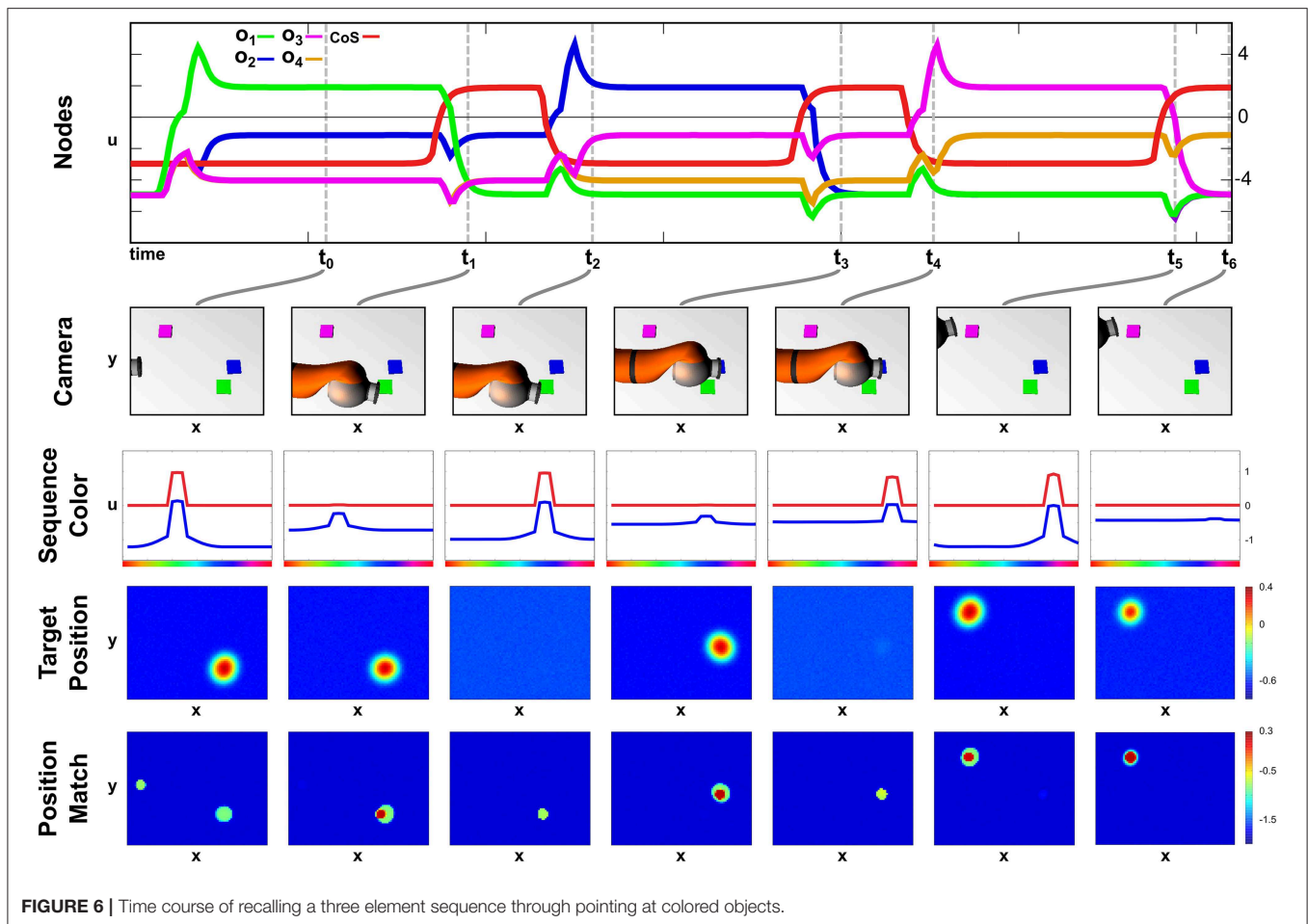


FIGURE 6 | Time course of recalling a three element sequence through pointing at colored objects.

as a sub-threshold activation blob, and the blue object is entirely absent. As the first movement is finished at t_1 all three objects are present in working memory as sub-threshold activation blobs.

Thus at t_2 , the second movement starts closely after the activation of the second ordinal node with the blue object as the target on the right side of the camera image. While the arm is moving the object is moved to the center/top position of the image, which results in a non-match between arm and target at the end of the movement, which can be seen at t_3 . Here working memory has updated the position of the blue object, which leads to an extraction of a different target position that does not match with the current position of the end effector. Only at t_4 after a second movement was generated, the blue object and the end effector match, which concludes the recall of the second element of the sequence.

The last movement toward the purple object is then conducted without any further perturbations and terminates after a single movement at t_6 .

4.3.2. Recall With a Missing Object

In this second recall episode demonstrating the robustness of the field network we start the recall in a scene that lacks the second object of the sequence. In **Figure 8**, activation snapshots of the

same sub-set of fields used in the previous perturbation episode are shown.

At points t_0 and t_1 , the network's activation develops analog to the previous two recall examples with a color slice used to extract the target position and the position match to determine the successful termination of the movement. However as the second ordinal node activates at t_2 no blue object is present in the scene, thus no sub-threshold activation blob overlaps with the blue color slice in the *Memory Color Selection* field and no peak forms.

At point t_3 , the blue object is added to the scene, which is committed to memory and afterwards extracted as a valid target position. The movement then concludes at t_4 with the arm occluding the purple object, which is kept in working memory due to the self-sustaining kernel. The working memory information is then used in t_5 , when the third ordinal node specifies purple as the next sequence color. Thus the sequence ends at t_6 with no further perturbations.

5. DISCUSSION

We have presented a network of dynamic neural fields that integrates the complete pathway from the sensor surface (vision)

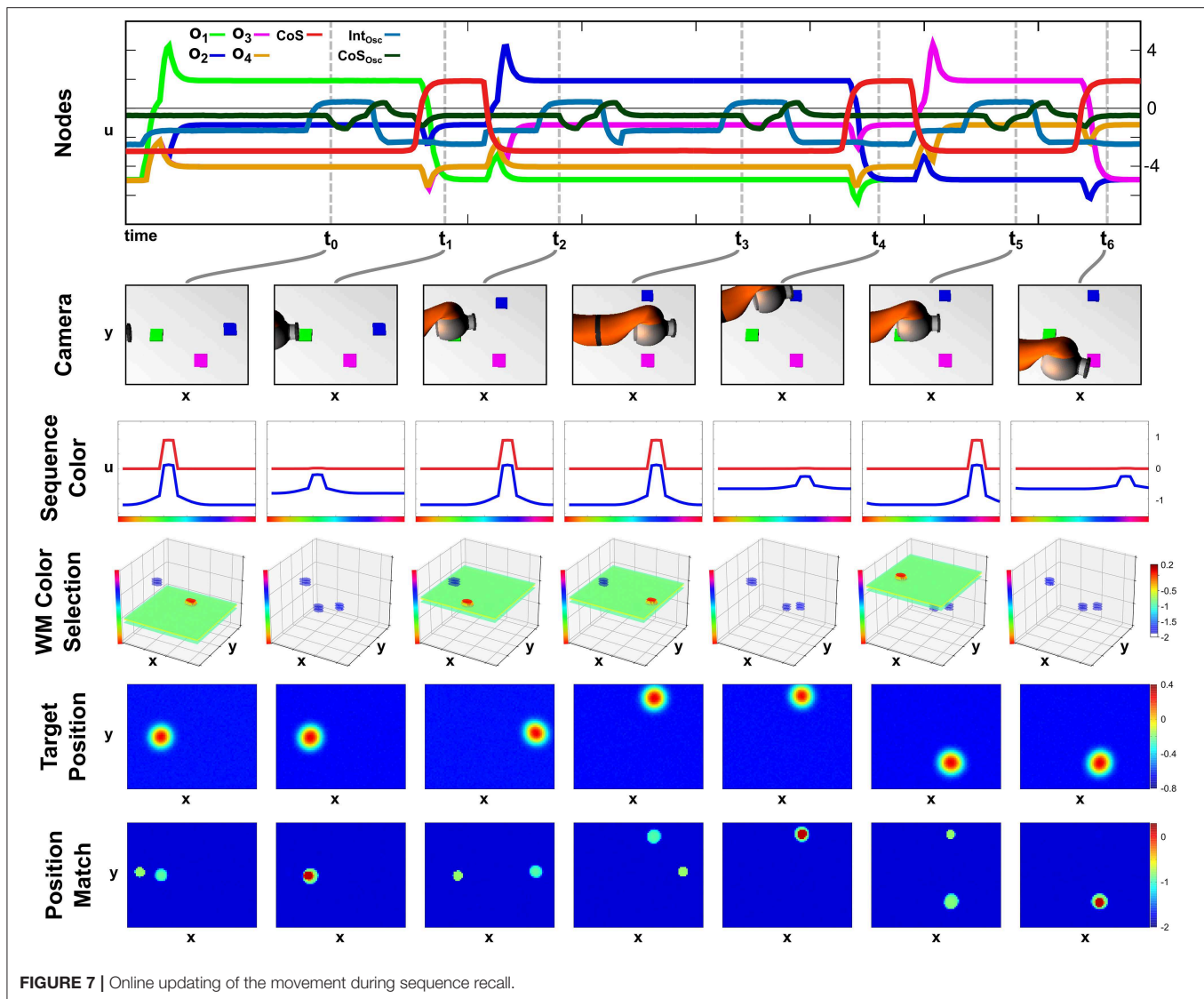


FIGURE 7 | Online updating of the movement during sequence recall.

to representations of higher cognition (serial order) and to the motor system (pointing). The network architecture enables a robotic agent to autonomously learn a sequence of colors from demonstration and then to act according to the defined serial order on a scene. Both during learning and while acting out the sequence, the transitions between elements of the sequence are detected without the need for an external control signal (The switch between learning and recall mode is not autonomous, however, reflecting a similar need for task instructions when a human operator performs such a task).

In each of the three sub-networks responsible for scene representation, the representation of serial order, and movement generation, sequential transitions between neural activation states are brought about through the mechanism of the condition of satisfaction. Thus, visual attention shifts only once a currently attended item has been committed to working memory. A transition to the next element in the serial order occurs only once the robot has successfully acted on the current element. And

an arm movement terminates only once the desired movement target has been reached. The mechanism of the condition of satisfaction thus reconciles the capacity to autonomously act according to learned or structurally determined plans with the capacity to be responsive to sensory or internal information about the achievement of goals.

5.1. What the Scenario Stands for

The scenario was simple, but meant to demonstrate the fundamental components of any neurally grounded autonomous robot.

- (1) A representation of the visual surround is the basis for any intelligent action directed at the world. It is also the basis for sharing an environment with a human user. We humans are particularly tuned to building scene representations which form the basis of much of our visual cognition (Henderson and Hollingworth, 1999). Scene representations need to

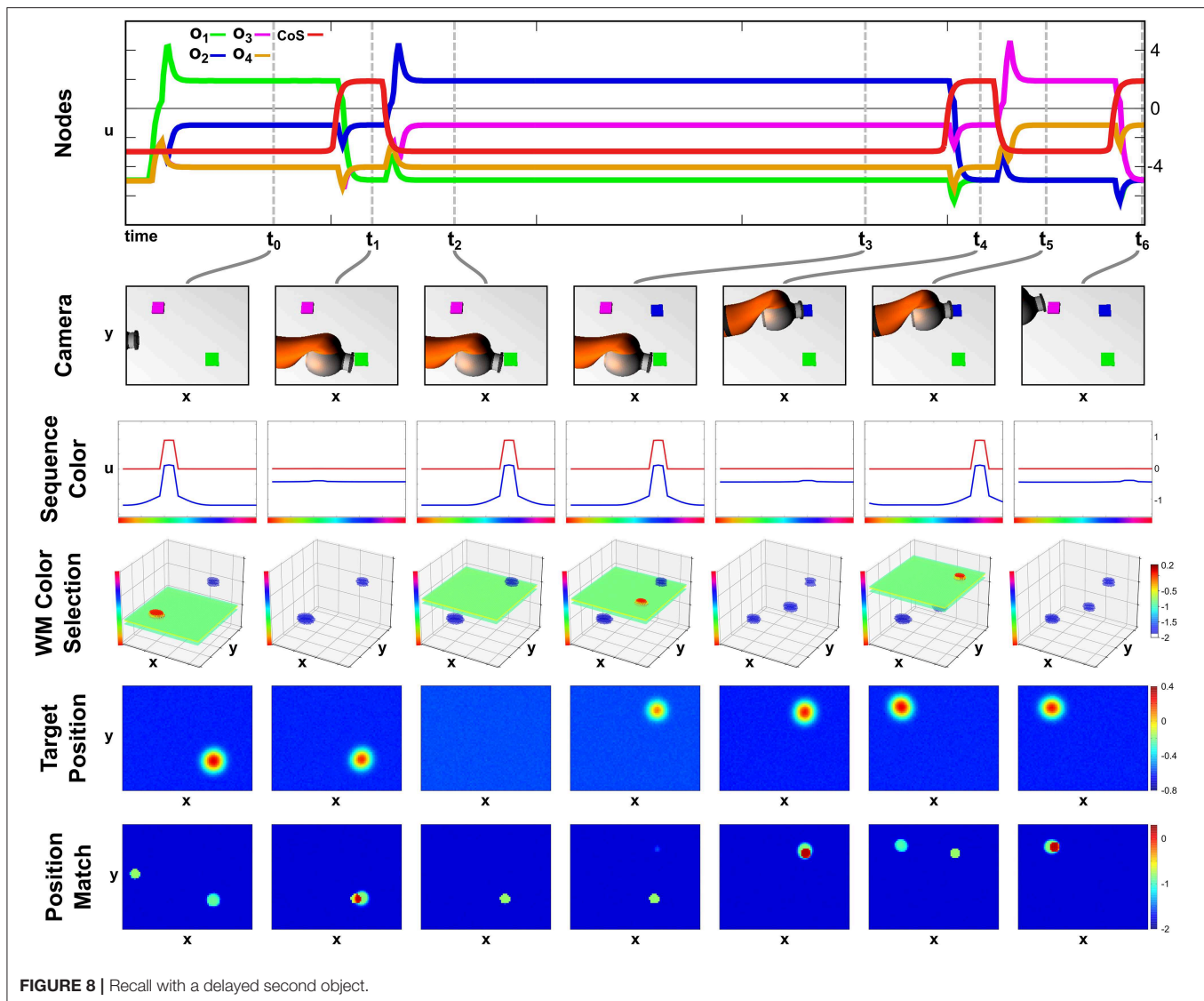


FIGURE 8 | Recall with a delayed second object.

include scene memory to deal with occlusions (e.g., by the agent's own body or body parts of a collaborating human user) and with a limited viewing range. Scene representations must also be open to updating, however, when the scene changes over time. Attentional selection is the key process that provides an interface between the scene and any action plan. So, while we stripped the system down to the bare essentials, the core processes of scene representation were covered.

- (2) Directing action to objects in the world requires transforming attentionally selected scene information into a coordinate frame anchored in the initial position of the actuator. In that representation, motor plans can be framed as movement parameters (Erlhagen and Schöner, 2002) that characterize the movement as a whole. Movements must be initiated and terminated, and time courses of motor commands must be generated that take the effector to the target. In dynamic environments, such as when a human user interferes with objects, the movement

parameters must be open to online updating. If movements still fail to reach the target, correction movements must be generated. Even in our extremely limited implementation, these core processes of movement generation were covered. Control issues, which are not trivial in human movement but are well-understood in robotics, were neglected.

- (3) The cognition of goal-directed action was simplified to serial order. Serial order is a cognitive construct in that it abstracts from the contents (what is serially ordered) and from time (when is each item addressed). Based on these abstractions, a broad set of actions can be conceived of as serially ordered processing steps. For instance, assembling a piece of IKEA furniture could be described this way. Unlike many classical, disembodied cognitive tasks, real action sequences require the capacity to deal with variable and perhaps unpredictable amounts of time needed to achieve each processing step. Learning the—a priori—arbitrary contents of a serially ordered sequence makes this scenario quite powerful. It goes beyond, for instance, a

mere capacity to imitate or emulate behavior, which would lead to the reproduction of the same movements or effects without generalization to new conditions. It also goes beyond the generation of sequences of behaviors that would be triggered by environmental conditions according to a fixed organizational scheme encoded in a behavior-based robotic architecture (Elements of fixed sequencing are contained in the present system such as when attentional selection always precedes pointing).

5.2. Scaling Beyond the Simplified Scenario

- (1) The scene representation system was built on a single, trivial color feature. We have previously explored and demonstrated elsewhere how a neural dynamic system of the same kind can deal with multiple feature dimensions (e.g., Chapter 8 of Schöner et al., 2016 or Grieben et al., 2018). This entails a feature binding problem that can be solved through a shared spatial dimension across multiple low-dimensional feature/space fields. Binding occurs by attending selectively to a spatial location and transmitting feature information separately in the different feature/space fields. In this account, such binding through space is ultimately the reason why objects need to be attended sequentially. Active gaze shifts would be another extension, the basis for which has been outline in previous work (e.g., Wilimzig et al., 2006). A further generalization would be to extend the scene representation into a semantic map in which objects are also classified and their class labels are stored. There is, of course, a plethora of neurally based feedforward classification system (reviewed in the section 1), and their outputs could be treated as such labels. Such modules may need improvement in terms of object segmentation and pose estimation (as discussed in Lomp et al., 2017), which clearly requires attentional processes. Visual search for such labels is a challenge that may need some research attention.
- (2) The motor domain is, of course, a much richer domain than we were able to demonstrate. There are myriad problems to be solved such as dealing with many degrees of freedom, dealing with obstacles, dealing with compliant actuators, controlling impedance, grasp planning and control, manipulating objects dynamically, and many more. For many of these, technical solutions are available or are being actively researched. Neurally grounded process accounts have not been developed as strongly as one would hope, however. The fundamental difference between human motor control and the control of current robot arms limits the extent to which approaches inspired by modern control theory and the theory of optimal control carry over to neurally inspired robotics (but see Driess et al., 2018).
- (3) We have looked at the learning and recall of a single sequence. To learn multiple sequences, additional processing substrate must be introduced that represents activation of such learned sequences as well as selection of a neural population when a new sequence is added to the sequence memory. In principle, an approach inspired by Adaptive Resonance Theory (Carpenter and Grossberg, 2017) may achieve that. We have outlined such an approach

in related work on contingency learning (Tekülve and Schöner, 2019), but important questions remain open such as how to align sequences of different lengths. The position encoding of serial order in the ordinal nodes makes it possible, however, to represent sequences that entail the same elements in different serial positions (Sandamirskaya and Schöner, 2010).

The sliver of cognition we have captured may be part of communication, showing each other what to do. If perception was better (e.g., recognizing events and perceiving relationships between actuators and objects), and if action was richer (e.g., the ability to use tools and manipulate objects), then the modeled interface would already make the robot quite useful. It would enable a robot to learn the solution of problems from a human user, as long as the perception system extracts the conceptual structure of the demonstrated action. A big extension would be the capacity of the system to solve problems by itself, devising the sequences of actions required to achieve a goal. This would require neural processes in new domains such as exploration, outcome representations, perhaps value systems. There is a growing literature on such models (Mnih et al., 2015), but their import for robotic learning is an open research problem.

5.3. Related Work

A number of groups have addressed object-directed action and the requisite perception in a similar neural-dynamic framework (Fard et al., 2015; Strauss et al., 2015; Tan et al., 2016). Serial order and the specific neural mechanism for sequencing neural activation patterns were not yet part of these efforts, which otherwise overlap with ours. A number of neural dynamic models of serial order or sequencing have been proposed (e.g., Deco and Rolls, 2005), but not been brought into robotic problems. One reason may be the lack of a control structure comparable to our condition of satisfaction, so that the sequences unfold in neural dynamics at a given rhythm that is not synchronized with perceptual events. Such systems would not remain tied to the actual performance of a sequence in the world.

Related attempts to model in neural terms the entire chain from perception to action have been made for robotic vehicles. For instance, Alexander and Sporns (2002) enabled a vehicle to learn from reward a task directed at objects that a robot vehicle was able to pick up. Gurney et al. (2004) realized a neurally inspired system the organized the organism (This paper is useful also for its careful discussion of different levels of descriptions for neurally inspired approaches to robotics). Both systems are conceptually in the fold of behavior-based robotics, in that the sequences of actions emerge from a neural architecture, modulated by adaptation. To our knowledge, systems of that kind have not yet been shown to be able to form serial order memories and acquire scene representations.

A different style of neural robotic model for cognition is SPAUN (Eliasmith et al., 2012). This is an approach based on the Neural Engineering Framework (Eliasmith, 2005), which is able to implement any neural dynamic model in a spiking neural network. Thus, models based on DFT may, in principle, be implemented within this framework. On the other hand,

SPAUN has also been turned to approaches to cognition that may not be compatible with the principles of DFT, in particular, the Vector Symbolic Architecture (VSA) framework that goes back to Smolensky, Kanerva, Plate, and Gayler (see Levy and Gayler, 2008 for review). In VSA, concepts are mapped onto high-dimensional vectors, that enable processing these concepts in the manner of symbol manipulation. If this approach is entirely free of non-neural algorithmic steps is not clear to us.

6. CONCLUSION

We have shown, in a minimal scenario, how sequences of attentional shifts, of movements, and of serially ordered actions can be autonomously generated in a neural dynamic framework that is free of any non-neural algorithmic control. The continuous or intermittent coupling to sensory and motor systems is made possible by creating neural attractor states. Inducing instabilities in a controlled manner enables the system to make sequential transitions between such states. As a result, the neural dynamic robot demonstrates a minimal form of cognition, learning and acting out serially ordered actions. Much work remains to be done to scale such systems to the real world.

DATA AVAILABILITY STATEMENT

All relevant data to reproduce the presented dynamic field network with our open-source software cedar (cedar.ini.rub.de)

REFERENCES

- Alexander, W. H., and Sporns, O. (2002). An embodied model of learning, plasticity, and reward. *Adapt. Behav.* 10:143. doi: 10.1177/1059712302010003001
- Amari, S.-I. (1977). Dynamics of pattern formation in lateral-inhibition type neural fields. *Biol. Cybernet.* 27, 77–87. doi: 10.1007/BF00337259
- Aoi, S., Manoonpong, P., Ambe, Y., Matsuno, F., and Wörgötter, F. (2017). Adaptive control strategies for interlimb coordination in legged robots: a review. *Front. Neurobot.* 11:39. doi: 10.3389/fnbot.2017.00039
- Ba, J. L., Mnih, V., and Kavukcuoglu (2015). “Multiple object recognition with visual attention,” in *IRLC* (San Diego, CA), 1–10.
- Bicho, E., Mallet, P., and Schöner, G. (2000). Target representation on an autonomous vehicle with low-level sensors. *Int. J. Robot. Res.* 19, 424–447. doi: 10.1177/02783640022066950
- Braitenberg, V. (1984). *Vehicles. Experiments in Synthetic Psychology*. Cambridge, MA: MIT Press.
- Brooks, R. A. (1986). A robust layered control system for a mobile robot. *IEEE J. Robot. Automat.* 2, 12–23. doi: 10.1109/JRA.1986.1087032
- Brooks, R. A. (1990). Elephants don't play chess. *Robot. Auton. Syst.* 6, 3–15. doi: 10.1016/S0921-8890(05)80025-9
- Brooks, R. A. (1991). New approaches to robotics. *Science* 253, 1227–1232. doi: 10.1126/science.253.5025.1227
- Bullock, D., and Grossberg, S. (1988). Neural dynamics of planned arm movements: emergent invariants and speed-accuracy properties during trajectory formation. *Psychol. Rev.* 95, 49–90. doi: 10.1037/0033-295X.95.1.49
- Buonomano, D. V., and Laje, R. (2010). Population clocks: motor timing with neural dynamics. *Trends Cogn. Sci.* 14, 520–527. doi: 10.1016/j.tics.2010.09.002
- Carpenter, G. A., and Grossberg, S. (2017). “Adaptive resonance theory,” in *Encyclopedia of Machine Learning and Data Mining*, eds C. Sammut and G. I. Webb (Boston, MA: Springer).
- Coombes, S., beim Graben, P., Potthast, R., and Wright, J. (eds.). (2014). *Neural Fields: Theory and Applications*. New York, NY: Springer Verlag.
- Deco, G., and Rolls, E. T. (2005). Sequential memory: a putative neural and synaptic dynamical mechanism. *J. Cogn. Neurosci.* 17, 294–307. doi: 10.1162/0899729053124875
- Driess, D., Zimmermann, H., Wolfen, S., Suissa, D., Haeufle, D., Hennes, D., et al. (2018). “Learning to control redundant musculoskeletal systems with neural networks and SQP: exploiting muscle properties,” in *ICRA* (Brisbane), 6461–6468.
- Eliasmith, C. (2005). A unified approach to building and controlling spiking attractor networks. *Neural Comput.* 17, 1276–1314. doi: 10.1162/0899766053630332
- Eliasmith, C., Stewart, T. C., Choo, X., Bekolay, T., DeWolf, T., Tang, C., et al. (2012). A large-scale model of the functioning brain. *Science* 338, 1202–1205. doi: 10.1126/science.1225266
- Engels, C., and Schöner, G. (1995). Dynamic fields endow behavior-based robots with representations. *Robot. Auton. Syst.* 14, 55–77. doi: 10.1016/0921-8890(94)00020-3
- Erlhagen, W., and Schöner, G. (2002). Dynamic field theory of movement preparation. *Psychol. Rev.* 109, 545–572. doi: 10.1037/0033-295X.109.3.545
- Fard, F. S., Hollensen, P., Heinke, D., and Trappenberg, T. P. (2015). Modeling human target reaching with an adaptive observer implemented with dynamic neural fields. *Neural Netw.* 72, 13–30. doi: 10.1016/j.neunet.2015.10.003
- Full, R. J., and Koditschek, D. E. (1999). Templates and anchors: neuromechanical hypotheses of legged locomotion on land. *J. Exp. Biol.* 2, 3–125.
- Ghigliazza, R. M., Altendorfer, R., Holmes, P., and Koditschek, D. (2003). A simply stabilized running model. *SIAM J. Appl. Dyn. Syst.* 2, 187–218. doi: 10.1137/S1111111102408311
- Grey, C. (1950). An imitation of life. *Sci. Am.* 182, 42–45. doi: 10.1038/scientificamerican0550-42

is contained within the manuscript. For technical questions regarding the software please contact the authors.

AUTHOR CONTRIBUTIONS

JT and AF performed the work, and contributed to the writing of the manuscript. YS and GS conceived of the project, provided research supervision, and contributed to the writing of the manuscript.

FUNDING

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - 402791869 (SCHO 336/12-1) within the SPP The Active Self (SPP 2134) and the SNSF grant PZ00P2_168183_1 Ambizione.

ACKNOWLEDGMENTS

We acknowledge support by the DFG Open Access Publication Funds of the Ruhr-Universität Bochum.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fnbot.2019.00095/full#supplementary-material>

- Grieben, R., Tekülve, J., Zibner, S. K. U., Schneegans, S., and Schöner, G. (2018). "Sequences of discrete attentional shifts emerge from a neural dynamic architecture for conjunctive visual search that operates in continuous time," in *CogSci 2018*, eds C. Kalish, M. Rau, J. Zhu, and T. Rogers, Madison, WI: Cognitive Science Society, 427–432.
- Gurney, K., Prescott, T. J., Wickens, J. R., and Redgrave, P. (2004). Computational models of the basal ganglia: from robots to membranes. *Trends Neurosci.* 27, 453–459. doi: 10.1016/j.tins.2004.06.003
- Henderson, J. M., and Hollingworth, A. (1999). High-level scene perception. *Annu. Rev. Psychol.* 50, 243–271. doi: 10.1146/annurev.psych.50.1.243
- Hersch, M., and Billard, A. G. (2008). Reaching with multi-referential dynamical systems. *Auton. Robots* 25, 71–83. doi: 10.1007/s10514-007-9070-7
- Holmes, P., Full, R. J., Koditschek, D., and Guckenheimer, J. (2006). The dynamics of legged locomotion: models, analyses, and challenges. *SIAM Rev.* 48, 207–304. doi: 10.1137/S0036144504445133
- Ijspeert, A. J. (2008). Central pattern generators for locomotion control in animals and robots: a review. *Neural Netw.* 21, 642–653. doi: 10.1016/j.neunet.2008.03.014
- Ijspeert, A. J., Nakanishi, J., Hoffmann, H., Pastor, P., and Schaal, S. (2013). Dynamical movement primitives : learning attractor models for motor behaviors. *Neural Comput.* 25, 328–373. doi: 10.1162/NECO_a_00393
- Kelleher J. D., and Dobnik S. (2017). "What is not where: the challenge of integrating spatial representations into deep learning architectures," in *CLASP Papers in Computational Linguistics Vol 1: Proceedings of the Conference on Logic and Machine Learning in Natural Language (LaML 2017)*, eds S. Dobnik and S. Lappin (Gothenburg), 41–52. Available online at: <http://hdl.handle.net/2077/54911>
- Kim, S., Shukla, A., and Billard, A. (2014). Catching objects in flight. *IEEE Trans. Robot.* 30, 1049–1065. doi: 10.1109/TRO.2014.2316022
- Knips, G., Zibner, S. K. U., Reimann, H., and Schöner, G. (2017). A neural dynamic architecture for reaching and grasping integrates perception and movement generation and enables on-line updating. *Front. Neurobot.* 11:9. doi: 10.3389/fnbot.2017.00009
- Kriegeskorte, N. (2015). Deep neural networks : a new framework for modeling biological vision and brain information processing. *Annu. Rev. Vis. Sci.* 1, 417–446. doi: 10.1146/annurev-vision-082114-035447
- Lecun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521, 436–444. doi: 10.1038/nature14539
- Levy, S. D., and Gayler, R. (2008). "Vector symbolic architectures: a new building material for artificial general intelligence," in *Proceedings of the 2008 Conference on Artificial General Intelligence 2008: Proceedings of the First AGI Conference* (Memphis, TN: IOS Press), 414–418.
- Lomp, O., Faubel, C., and Schöner, G. (2017). A neural-dynamic architecture for concurrent estimation of object pose and identity. *Front. Neurobot.* 11:23. doi: 10.3389/fnbot.2017.00023
- Lomp, O., Richter, M., Zibner, S. K., and Schöner, G. (2016). Developing dynamic field theory architectures for embodied cognitive systems with cedar. *Front. Neurobot.* 10:14. doi: 10.3389/fnbot.2016.00014
- Mataric, M. J. (1998). Behavior-based robotics as a tool for synthesis of artificial behavior and analysis of natural behavior. *Trends Cogn. Sci.* 2, 82–87. doi: 10.1016/S1364-6613(98)01141-3
- Michel, O. (2004). Cyberbotics Ltd. webots: professional mobile robot simulation. *Int. J. Adv. Robot. Syst.* 1:5. doi: 10.5772/5618
- Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., et al. (2015). Human-level control through deep reinforcement learning. *Nature* 518, 529–533. doi: 10.1038/nature14236
- Proetzs, M., Luksch, T., and Berns, K. (2010). Development of complex robotic systems using the behavior-based control architecture iB2C. *Robot. Auton. Syst.* 58, 46–67. doi: 10.1016/j.robot.2009.07.027
- Ruf, A., and Horaud, R. (1999). Visual servoing of robot manipulators part I: projective kinematics. *Int. J. Robot. Res.* 18, 1101–1118. doi: 10.1177/02783649922067744
- Sandamirskaya, Y., and Schöner, G. (2010). An embodied account of serial order: how instabilities drive sequence generation. *Neural Netw.* 23, 1164–1179. doi: 10.1016/j.neunet.2010.07.012
- Schaal, S., Ijspeert, A., and Billard, A. (2003). Computational approaches to motor learning by imitation. *Philos. Trans. R. Soc. Lond. B* 358, 537–547. doi: 10.1098/rstb.2002.1258
- Schmidhuber, J. (2015). Deep learning in neural networks : an overview. *Neural Netw.* 61, 85–117. doi: 10.1016/j.neunet.2014.09.003
- Schöner, G., Dose, M., and Engels, C. (1995). Dynamics of behavior: {t}heory and applications for autonomous robot architectures. *Robot. Auton. Syst.* 16, 213–245. doi: 10.1016/0921-8890(95)00049-6
- Schöner, G., and Kelso, J. A. S. (1988). Dynamic pattern generation in behavioral and neural systems. *Science* 239, 1513–1520. doi: 10.1126/science.3281253
- Schöner, G., Spencer, J. P., and DFT Research Group, T. (2016). *Dynamic Thinking: A Primer on Dynamic Field Theory*. Oxford University Press.
- Schöner, G., Tekülve, J., and Zibner, S. (2019). "Reaching for objects: a neural process account in a developmental perspective," in *Reach-to-Grasp Behavior*, eds D. Corbetta and M. Santello (New York, NY: Routledge).
- Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., and Poggio, T. (2007). Robust object recognition with cortex-like mechanisms. *IEEE Trans. Pattern Anal. Mach. Intell.* 29, 411–426. doi: 10.1109/TPAMI.2007.56
- Strauss, S., Woodgate, P. J., Sami, S. A., and Heinke, D. (2015). Choice reaching with a LEGO arm robot (CoRLEGO): the motor system guides visual attention to movement-relevant information. *Neural Netw.* 72, 3–12. doi: 10.1016/j.neunet.2015.10.005
- Tan, C. K., Plöger, P. G., and Trappenberg, T. P. (2016). "A neural field approach to obstacle avoidance," in *KI 2016: Advances in Artificial Intelligence; Lecture Notes in Computer Science*, Vol. 9904, eds G. Friedrich, M. Helmert, and F. Wotawa (Springer International Publishing), 69–87.
- Tekülve, J., and Schöner, G. (2019). "Autonomously learning beliefs is facilitated by a neural dynamic network driving an intentional agent," in *International Conference on Development and Learning and Epigenetic Robotics* (Oslo).
- Wilimzig, C., Schneider, S., and Schöner, G. (2006). The time course of saccadic decision making : dynamic field theory. *Neural Netw.* 19, 1059–1074. doi: 10.1016/j.neunet.2006.03.003
- Wilson, H. R., and Cowan, J. D. (1973). A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. *Kybernetik* 13, 55–80. doi: 10.1007/BF00288786
- Zibner, S. K. U., Faubel, C., Iossifidis, I., and Schöner, G. (2011). Dynamic neural fields as building blocks for a cortex-inspired architecture of robotic scene representation. *IEEE Trans. Auton. Mental Dev.* 3, 74–91. doi: 10.1109/TAMD.2011.2109714

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2019 Tekülve, Fois, Sandamirskaya and Schöner. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

APPENDIX

Network Parameters

The following tables list the field parameters of the different sub-networks. Lateral kernels are constructed according to the following equation: $\omega(a, \sigma, c_{inh}) = c_{inh} + \sum_i a_i \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp(\frac{-x_i^2}{2\sigma_i^2})$. The learning rate used in Equation (9) is $\eta = 0.05$.

TABLE A1 | Parameter values of the scene representation sub-network.

Name	Resting level	Lateral Kernel	Inputs
SCENE REPRESENTATION			
Saliency	$u_{sal} \quad h_{sal} = -0.5$	$\omega_{sal} : a_1 = 1, \sigma_1 = 3, c_{inh} = -0.0001$	$w_{img} = 1$
Saliency selection	$u_{sel} \quad h_{sel} = -2.5$	$\omega_{sel} : a_1 = 3, \sigma_1 = 2, c_{inh} = -0.08$	$w_{sal} = 2.2, w_{exp} = 2.0, w_{mem} = -0.1$
Space color maps	$u_{scm} \quad h_{scm} = -0.5$	$\omega_{scm} : c_{inh} = -0.0001$	$w_{img} = 1$
Scene space selection	$u_{ssl} \quad h_{ssl} = -1.5$	$\omega_{ssl} : c_{inh} = -0.0001$	$w_{scm} = 1, w_{sel} = 1$
Working memory	$u_{mem} \quad h_{mem} = -5$	$\omega_{mem} : a_1 = 5, \sigma_1 = 2, a_2 = -2.5, \sigma_2 = 6, c_{inh} = -0.0001$	$w_{sel} = 1, w_{nbk} = 3.5, w_{ssl} = 1$
WM space selection	$u_{wsl} \quad h_{wsl} = -1.5$	$\omega_{wsl} : c_{inh} = -0.0001$	$w_{mem} = 1, w_{sel} = 1$
Color match	$u_{cm} \quad h_{cm} = -1.5$	$\omega_{cm} : c_{inh} = -0.0001$	$w_{ssl} = 1, w_{wsl} = 1$
Explore int	$u_{exp} \quad h_{exp} = -0.5$	—	$w_{ecs} = -3$
Explore CoS	$u_{ecs} \quad h_{ecs} = -0.5$	—	$w_{cm} = 1$
Off-set detector fast	$u_{dfa} \quad h_{dfa} = -0.2$	—	$w_{sin} = -1, w_{dsl} = 0.5$
Off-set detector slow	$u_{dsl} \quad h_{dsl} = -0.2$	—	$w_{sex} = 1$

TABLE A2 | Parameter values of the arm movement sub-network.

Name	Resting level	Lateral Kernel	Inputs
ARM MOVEMENT			
Target position	$u_{tp} \quad h_{tp} = -0.5$	$\omega_{tp} : c_{inh} = -0.001$	$w_{wcl} = 1, w_{dfa} = -2$
EEF position	$u_{ep} \quad h_{ep} = -0.5$	$\omega_{ep} : a = 25, \sigma = 2, c_{inh} = -0.3$	$w_{epg} = 10$
EEF position gate	$u_{epg} \quad h_{epg} = -0.5$	$\omega_{epg} : c_{inh} = -0.001$	$w_{ppc} = 1, w_{oin} = -2$
Position match	$u_{pm} \quad h_{pm} = -1.5$	—	$w_{ep} = 1, w_{tp} = 1$
Oscillator fast	$u_{ofa} \quad h_{ofa} = -2.5$	$\omega_{ofa} : c_{inh} = -0.001$	$w_{oin} = 2, w_{efc} = 1, w_{osl} = -4$
Oscillator slow	$u_{osl} \quad h_{osl} = -2.5$	$\omega_{osl} : c_{inh} = -0.001$	$w_{oin} = 2, w_{efc} = 1$
Oscillate intention	$u_{oin} \quad h_{oin} = -0.5$	—	$w_{rea} = 1, w_{ocs} = -2$
Oscillate CoS	$u_{ocs} \quad h_{ocs} = -0.5$	—	$w_{ofa} = -1, w_{osl} = 1$
Reach intention	$u_{rea} \quad h_{rea} = -0.5$	—	$w_{rcl} = 1, w_{rcs} = -2$
Reach CoS	$u_{rcs} \quad h_{rcs} = -0.5$	—	$w_{pm} = 1$

TABLE A3 | Parameter values of the serial order sub-network.

Name	Resting level	Lateral Kernel	Inputs
SERIAL ORDER			
Ordinal node	$o_i \quad h_o = -5$	$w_{o_i, o_i} = 4.8$	$w_{o_i, o_i} = 2, w_{m_{i-1}, o_i} = 2.9, w_{m_i, o_i} = 3.8, w_{CoS} = 2, w_{oh+} = 3$
Memory node	$m_i \quad h_m = -5$	$w_{m_i, m_i} = 5$	$w_{o_i, m_i} = 2.6, w_{mh+} = 3$
Sequence CoS	$u_{CoS} \quad h_{CoS} = -0.5$	—	$w_{rcs} = 1, w_{dfa} = 1$
Sequence color	$u_{col} \quad h_{col} = -0.3$	$\omega_{col} : a = 1, \sigma = 3, c_{inh} = -0.3$	$w_{col} = 1, w_{o_i, u_{col}} \in [0, 1]$

TABLE A4 | Parameter values of task related fields of the dynamic field network.

Name	Resting level	Lateral Kernel	Inputs
LEARN AND RECALL			
Learn node	$u_{lrn} \quad h_{lrn} = -0.5$	—	$w_{manual} = 1$
Recall node	$u_{rcl} \quad h_{rcl} = -0.5$	—	$w_{manual} = 1$
Learn color	$u_{lcol} \quad h_{lcol} = -1.5$	—	$w_{lrn} = 1, w_{wsl} = 1$
Recall color	$u_{rcol} \quad h_{rcol} = -1.5$	—	$w_{col} = 1, w_{rcl} = 1$
WM color selection	$u_{wcl} \quad h_{wcl} = -1.5$	$\omega_{wcl} : c_{inh} = -0.0001$	$w_{rcol} = 1, w_{mem} = 1$



Experimental and Computational Study on Motor Control and Recovery After Stroke: Toward a Constructive Loop Between Experimental and Virtual Embodied Neuroscience

OPEN ACCESS

Edited by:

Judith Peters,
Maastricht University, Netherlands

Reviewed by:

Manish Sreenivasa,
University of Wollongong, Australia
Philipp Beckerle,
Technical University
Dortmund, Germany

*Correspondence:

Anna Letizia Allegra Mascaro
allegra@lens.unifi.it

† These authors have contributed
equally to this work

Received: 28 June 2019

Accepted: 08 May 2020

Published: 07 July 2020

Citation:

Allegra Mascaro AL, Falotico E, Petkoski S, Pasquini M, Vannucci L, Tort-Colet N, Conti E, Resta F, Spalletti C, Ramalingasetty ST, von Arnim A, Formento E, Angelidis E, Blixhavn CH, Leergaard TB, Caleo M, Destexhe A, Ijspeert A, Micera S, Laschi C, Jirsa V, Gewaltig M-O and Pavone FS (2020) Experimental and Computational Study on Motor Control and Recovery After Stroke: Toward a Constructive Loop Between Experimental and Virtual Embodied Neuroscience. *Front. Syst. Neurosci.* 14:31. doi: 10.3389/fnsys.2020.00031

Anna Letizia Allegra Mascaro^{1,2*†}, Egidio Falotico^{3†}, Spase Petkoski^{4†}, Maria Pasquini³, Lorenzo Vannucci³, Núria Tort-Colet⁵, Emilia Conti^{2,6}, Francesco Resta^{2,6}, Cristina Spalletti¹, Shravan Tata Ramalingasetty⁷, Axel von Arnim⁸, Emanuele Formento⁹, Emmanouil Angelidis^{8,10}, Camilla H. Blixhavn¹¹, Trygve B. Leergaard¹¹, Matteo Caleo^{1,12}, Alain Destexhe⁵, Auke Ijspeert⁷, Silvestro Micera^{3,9}, Cecilia Laschi³, Viktor Jirsa⁴, Marc-Oliver Gewaltig^{13†} and Francesco S. Pavone^{2,6†}

¹ Neuroscience Institute, National Research Council, Pisa, Italy, ² European Laboratory for Non-Linear Spectroscopy, Sesto Fiorentino, Italy, ³ Department of Excellence in Robotics & AI, The BioRobotics Institute, Scuola Superiore Sant'Anna, Pontedera, Italy, ⁴ Aix-Marseille Université, Inserm, INS UMR_S 1106, Marseille, France, ⁵ Paris-Saclay University, Institute of Neuroscience, CNRS, Gif-sur-Yvette, France, ⁶ Department of Physics and Astronomy, University of Florence, Florence, Italy, ⁷ Biorobotics Laboratory, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, ⁸ Fortiss GmbH, Munich, Germany, ⁹ Bertarelli Foundation Chair in Translational NeuroEngineering, Institute of Bioengineering, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, ¹⁰ Chair of Robotics, Artificial Intelligence and Embedded Systems, Department of Informatics, Technical University of Munich, Munich, Germany, ¹¹ Institute of Basic Medical Sciences, University of Oslo, Oslo, Norway, ¹² Department of Biomedical Sciences, University of Padua, Padua, Italy, ¹³ Blue Brain Project (BBP), École Polytechnique Fédérale de Lausanne (EPFL), Geneva, Switzerland

Being able to replicate real experiments with computational simulations is a unique opportunity to refine and validate models with experimental data and redesign the experiments based on simulations. However, since it is technically demanding to model all components of an experiment, traditional approaches to modeling reduce the experimental setups as much as possible. In this study, our goal is to replicate all the relevant features of an experiment on motor control and motor rehabilitation after stroke. To this aim, we propose an approach that allows continuous integration of new experimental data into a computational modeling framework. First, results show that we could reproduce experimental object displacement with high accuracy via the simulated embodiment in the virtual world by feeding a spinal cord model with experimental registration of the cortical activity. Second, by using computational models of multiple granularities, our preliminary results show the possibility of simulating several features of the brain after stroke, from the local alteration in neuronal activity to long-range

connectivity remodeling. Finally, strategies are proposed to merge the two pipelines. We further suggest that additional models could be integrated into the framework thanks to the versatility of the proposed approach, thus allowing many researchers to achieve continuously improved experimental design.

Keywords: motor control, stroke, rehabilitation, neural mass, spiking neuronal networks, brain network models, Kuramoto oscillators, closed-loop simulation

1. INTRODUCTION

In nature, the activity of the brain of an individual interacting with the environment is conditioned by the response of the environment itself, in that the output of the brain is relevant only if it has the ability to impact the future and hence the input the brain receives. This “closed-loop” can be simulated in a virtual world, where simulated experiments reproduce actions (output from the brain) that have consequences (future input to the brain) (Zrenner et al., 2016). To the aim of reproducing *in silico* the complexity of real experiments, different levels of modeling shall be integrated. However, since modeling all components of an experiment is very difficult, traditional approaches of computational neuroscience reduce the experimental setups as much as possible. An “Embodied brain” (or “task dynamics,” see Zrenner et al., 2016) approach could overcome these limits by associating the modeled brain activity with the generation of behavior within a virtual or real environment, i.e., an entailment between an output of the brain and a feedback signal into the brain (Reger et al., 2000; DeMarse et al., 2001; Tessadori et al., 2012). The experimenter can interfere with the flow of information between the neural system and environment on the one hand and the state and transition dynamics of the environment on the other. Closing the loop can be performed effectively by (i) validating the models on experimental data, and (ii) designing new experiments based on the hypotheses formulated by the simulations. On the example shown in **Figure 1**, data on brain activity (be it, for instance, from electrophysiological recordings or imaging) and on the environment (e.g., by means of kinematic or dynamic measures) from the real experiment are used to feed the models of the *in silico* representation of the experiment. From a comparison of the real and model-based data, the features that are most important to replicate the real experiment are identified, and thus novel insights are generated (**Figure 1**). To realize such a complex virtual system, many choices can be made, for instance on the brain model or spinal cord model that best represent the salient features of experimental measures to be replicated. The ideal framework shall comprise a library of tools to choose from, to reproduce a variety of experimental paradigms in the virtual environment. By briefly introducing the state of the art in brain and spinal cord modeling, we will discuss few classes of models to pick from an ideal library.

1.1. State of the Art

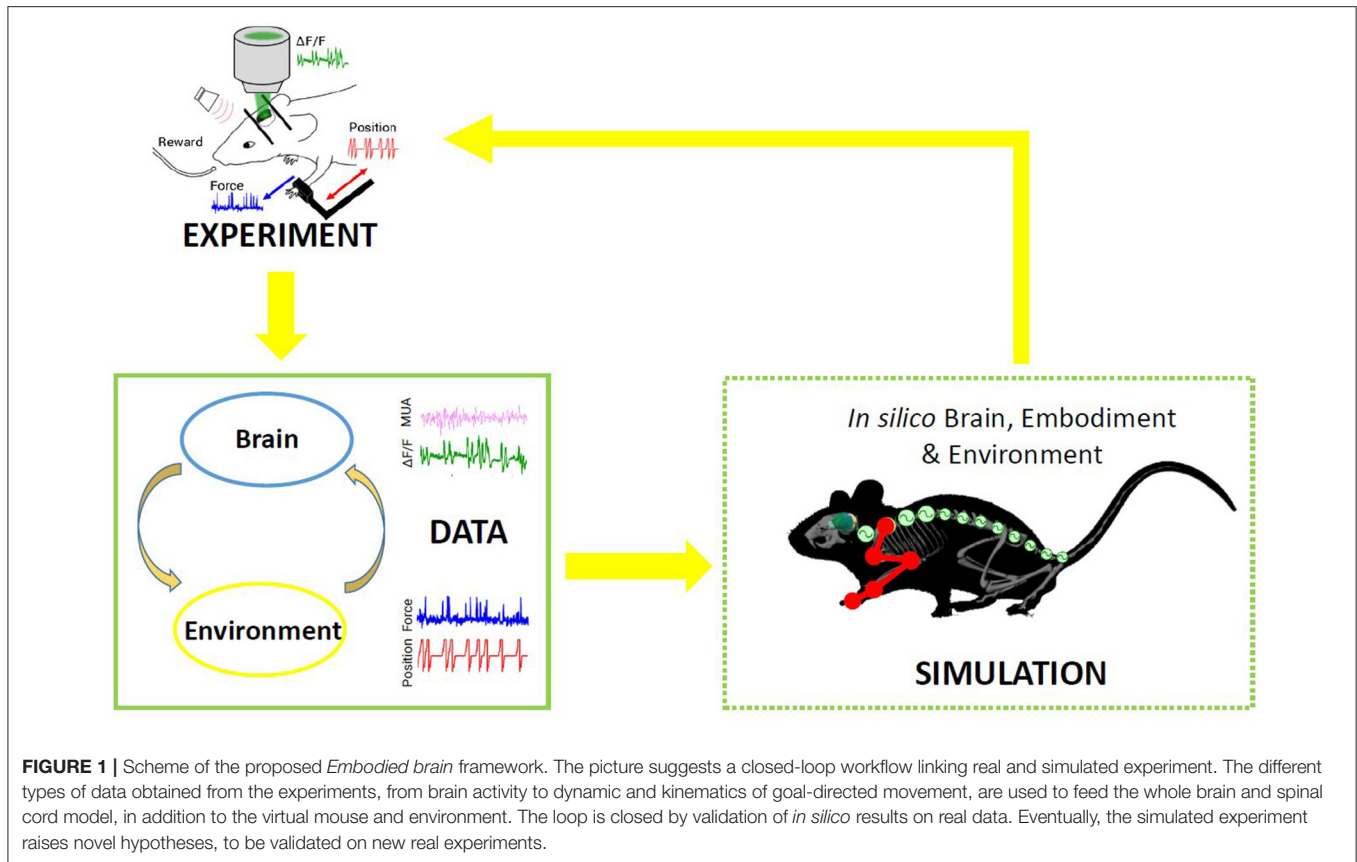
1.1.1. Local Cortical Network Modeling

Biologically detailed models of a single neuron such as the Hodgkin and Huxley model (Hodgkin and Huxley, 1952)

take into account the activity of the ion channels in the cell membrane that lead to changes in the membrane potential, eventually causing the neuron to spike. However, simpler but still biologically realistic models of the single cell are preferable when interested in modeling the dynamics of a larger number of cells. A good candidate is the adaptive exponential integrate and fire (adex) neuron model (Brette and Gerstner, 2005), which has been shown to reproduce the intrinsic neuronal properties of a number of cell types, including those with spike frequency adaptation (Destexhe, 2009). Interestingly, adex neurons network models with different adaptation levels can reproduce the dynamical properties of distinctive brain states such as wakefulness, sleep, or anesthesia (Zerlaut et al., 2017; Nghiem et al., 2020). This property makes adex neuron networks suitable to model the dynamics of the emerging activity in a local network of neurons after injury, given that, after a stroke, the dynamics of the local network switches to a slow oscillatory rhythm which resembles that of sleep or anesthesia (Butz et al., 2004). Moreover, alterations in low-frequency cortical activity in the peri-infarct cortex after stroke are known to correlate with motor recovery (e.g., Yilmaz et al., 2015; Ramanathan et al., 2018).

1.1.2. Brain Network Modeling

Efforts have been made to reconstruct single brain regions with as many details as possible (Markram et al., 2015), or to build detailed networks of multi-compartment oscillators (Izhikevich and Edelman, 2008). Contrary to the detailed models, top-down modeling seeks to elucidate whole-brain network mechanisms, which may underpin a variety of apparently diverse neurophysiological phenomena. Neural masses formalisms have been used over many years to develop macroscopic models that capture the collective dynamics of large neural assemblies (Deco et al., 2008; Sanz-Leon et al., 2015). In this case the activity of a macroscopic brain region is often directly derived from populations of spiking neurons as a mean-field using concepts from statistical physics (e.g., Wong and Wang, 2006; Stefanescu and Jirsa, 2008; Zerlaut et al., 2017). In other cases the statistics of the macroscopic brain activity is derived more phenomenologically while still conserving some basic physiological principles such as division on excitatory and inhibitory neurons, e.g., the seminal Wilson Cowan model (Wilson and Cowan, 1972). The third subclass of neural masses contains purely phenomenologically derived computational models that aim to reproduce certain dynamical properties of the macroscopic neuronal activity, such as e.g., seizure dynamics (Jirsa et al., 2014; Saggio et al., 2017), whilst different



realizations of damped or self-sustained oscillators are often used to model the coherent fluctuations of resting-state activity (Cabral et al., 2011; Deco et al., 2016). Depending on the working point of the system, the macroscopic dynamics can be described not only by physiologically derived mean-fields, but also by phenomenological models in their canonical form (Izhikevich, 1998; Deco et al., 2009, 2011). Hence, phase oscillators (Kuramoto, 1984) are often chosen to model and study the coactivation patterns in the brain, as some kind of a minimal model explanation (Batterman and Rice, 2014) for the synchronized behavior over a network (Pikovsky et al., 2001; Breakspear et al., 2010).

Connecting the neural masses in large-scale brain network models (BNM) became possible with the progress of non-invasive structural brain imaging (Johansen-Berg and Rushworth, 2009). This allowed extraction of biologically realistic brain connectivity, the so-called connectome, which shapes the local neuronal activity to the emergent network dynamics (Honey et al., 2007; Ghosh et al., 2008; Deco et al., 2009; Sanz-Leon et al., 2015; Petkoski et al., 2018; Petkoski and Jirsa, 2019).

The large-scale BNM have been used to interpret healthy (Cabral et al., 2011; Deco et al., 2016) or pathological (Nakagawa et al., 2013; Zimmermann et al., 2016; Saenger et al., 2018) brain activity. This is often reflected in the coherence between brain rhythms (Lachaux et al., 1999) that also describes the functional

connectivity (FC) of the brain as an important marker of its spatio-temporal organization (Ghosh et al., 2008; Deco et al., 2009, 2011; Deco and Jirsa, 2012; Petkoski et al., 2018).

The Virtual Brain (TVB) (Sanz Leon et al., 2013; Sanz-Leon et al., 2015) is a commonly used neuroinformatics platform for full brain simulations. It supports a systematic exploration of the underlying components of a large-scale BNM: the structural connectivity (SC) and the local dynamics that depend on the neurophysiological mechanisms or phenomena being studied. In this way, the BNM allows to describe structural changes (through connectivity variation including stroke, motor learning and recovery) and subsequent functional consequences accessible to modeling and empirical data collection on the meso, macro and behavioral level. The modeling with TVB thus represents a useful paradigm for multi-scale integration. TVB has been already utilized in modeling functional mechanism of recovery after stroke in humans (Falcon et al., 2015, 2016), identifying that the post-stroke brain favors excitation-over-inhibition and local-over-global dynamics. For studying the changes in synchronization, as we intend to do, TVB offers a range of oscillatory models for the neural activity. One of these is the Kuramoto model (KM), which captures the emergent behavior of a large class of oscillators that are near an Andronov-Hopf bifurcation (Kuramoto, 1984), including some population rate models (Ton et al., 2014). This makes the KM well-suited for assessing how the connectome governs the synchronization

between distant brain regions (Breakspear et al., 2010; Cabral et al., 2011, 2012; Ponce-Alvarez et al., 2015; Petkoski et al., 2018).

1.1.3. Spinal Cord Modeling

The brain controls its body through neural signals originating from the brain and processed by the spinal cord to control muscle activation in order to perform a large variety of behaviors. Several biologically realistic functional models of the spinal cord have been developed and tested in closed loop simulations with musculoskeletal embodiments. Stienen et al. (2007) developed a fairly complete model that includes Ia, Ib, and II sensory afferents, both monosynaptic and polysynaptic reflexes as well as Renshaw cells, improving a previous work by Bashor (1998). The model was tested with a musculoskeletal model consisting of a generic antagonistic couple of muscles, thus lacking a realistic validation scenario. Cisi and Kohn developed a web-based framework for the simulation of generic spinal cord circuits with associated muscles, that aims at replicating realistic experimental conditions (i.e., electrical stimulation) (Cisi and Kohn, 2008). Sreenivasa et al. (2016) developed a specific neuro-musculoskeletal system, upper limb with biceps and triceps, and validated it against human recordings. In Moraud et al. (2016), a simple spinal cord model of the rat, lacking any descending stimuli, was developed in order to study how such circuitry can correct the gait after a spinal cord injury and embedded in a closed loop simulation with biomechanical hindlimbs. All of the mentioned works were tested primarily for the generation of reflex motions, and not as intermediate levels of more complex controllers such as ones capable of generating voluntary movements.

1.2. Aim of the Work

We propose a framework (*“Embodied brain closed loop”*) endowed with a library of modeling tools that will eventually allow to realize entirely virtual experiments. We focused on an experiment on motor control and motor recovery after stroke described in Spalletti et al. (2017) and Allegra Mascaro et al. (2019), whose simulation requires two main tiles. The first is the realization of voluntary movements in a virtual milieu. This piece requires monitoring and modeling of many components of movement control, from brain activity to body kinematics and displacement of virtual objects. The second is the simulation of brain injury. This includes modeling of acute consequences but also of neuronal plasticity after brain damage, either spontaneous or supported by treatment. Both local and long-range modulation of neuronal activity should be accounted to simulate the brain after stroke, since local alteration of neuronal activity in the peri-infarct area is known to be associated to remodeling of long-range functional and structural connectivity (several comprehensive reviews have summarized this research, e.g., Carmichael et al., 2017). To build those tiles, we developed two pipelines that target, on one side, the physiological execution of movements and, on the other, pathological alterations and plasticity (**Figure 2**). The first (*“Movement-driven models”* pipeline) aims at reproducing in a virtual environment how a goal-directed movement is performed and represented in the healthy brain. Data recorded on healthy mice are used as an input to the spinal cord model,

attached to the muscles of the simulated embodiment (see **Figure 2**, red box). The goal of the second pipeline (*“Stroke models”*) is to reproduce both local and long-range consequences of stroke. We developed a spiking neurons model that could simulate the local brain dynamics, and in particular the abnormal oscillatory activity taking place in the peri-infarct cortex (see **Figure 2**, lower line in the green box). Also, we show how the simulation of brain activity by neural mass models allows replicating the evolution of functional connectivity in mouse brain after a stroke and under rehabilitation (see **Figure 2**, upper line in the green box).

2. METHODS

Cortical recordings and behavioral data from the experiments described in this section are used to build and validate the brain models and the output in the virtual environment.

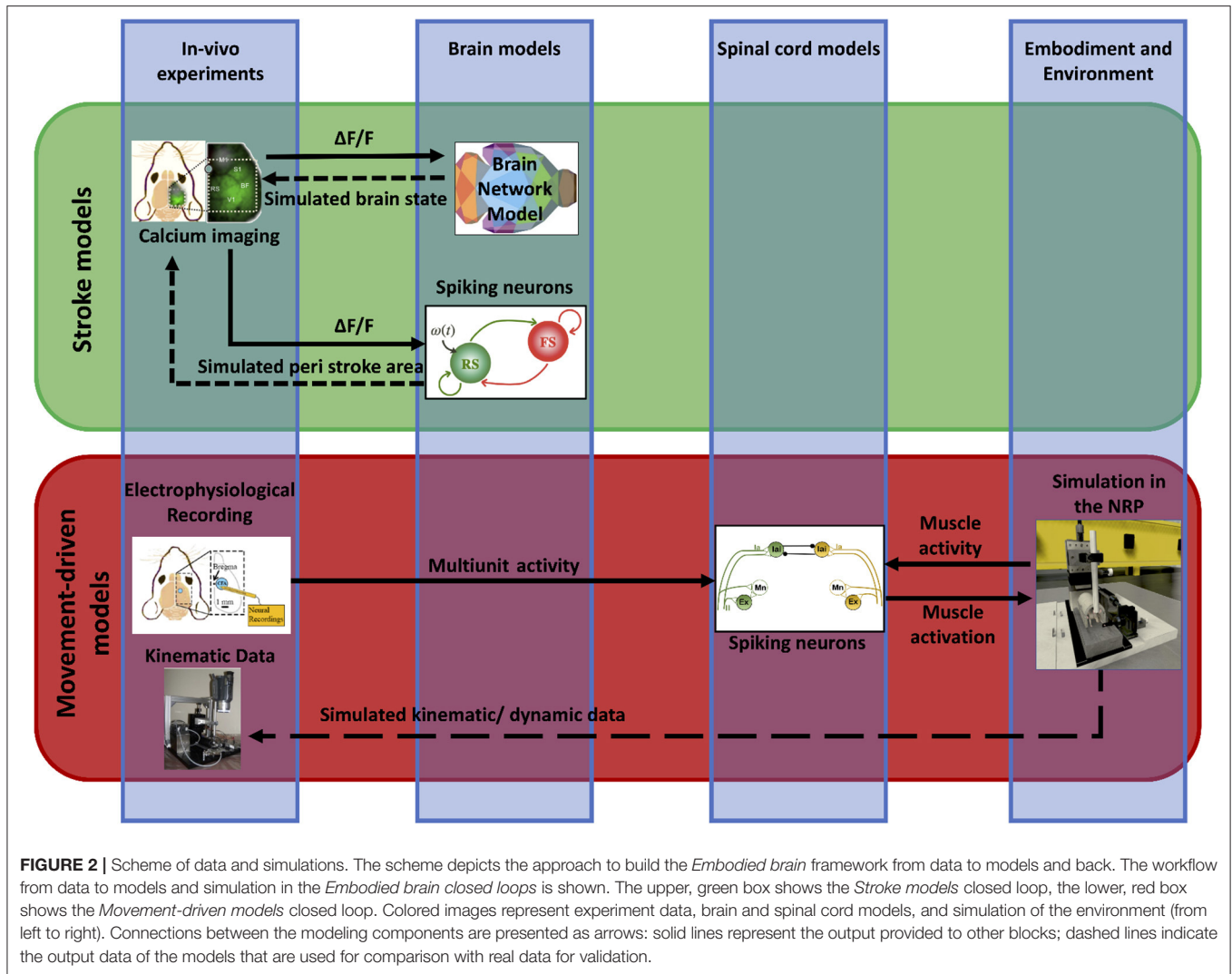
2.1. *In vivo* Experiments

On the experimental side, we performed electrophysiological recordings (**Figure 3A**) and wide-field calcium imaging (**Figure 3B**) in awake mice performing active forelimb retraction on a robotic device (M-Platform). These experiments allowed gathering simultaneous information on the neuronal activity, force applied during active forelimb retraction and position of the forelimb, as displayed in the lower panels of **Figure 3**. The electrophysiological data and the recordings of limb position were used to feed the spinal cord model, as described in section 4.1. The features of the wide-field calcium data recordings were used to build the spiking neurons brain model and to validate the BNM, section 3.3. All the procedures were in accordance with the Italian Ministry of Health for care and maintenance of laboratory animals (law 116/92) and in compliance with the European Communities Council Directive n. 2010/63/EU, under authorizations n. 183/2016-PR (imaging experiments) and n. 753/2015-PR (electrophysiology experiments).

2.1.1. Robotic Training on the M-Platform

The M-Platform is a robotic device designed to train mice to perform active forelimb retraction (Pasquini et al., 2018). Briefly, the main component of the device is a linear actuator that moves a linear slide where a custom handle is screwed. Moreover, the platform is provided with a system to control the friction on the slide and a pump for the reward. During the experiments, while the mouse has its left paw connected to the slide, first the linear actuator extends the forelimb then the animal has to perform an active pulling movement to come back to the starting point and to receive a reward. Force signal and position of the forelimb are recorded respectively by a load cell and a webcam.

In sections 2.1.3 and 2.1.4, we describe two different experiments with the robotic device. In the first one, the M-Platform is embedded with Omniplex D System (Plexon, USA) to obtain *in vivo* electrophysiological recording during the task. In the second one, the kinetic and kinematic parameters are synchronized with wide-field calcium imaging recordings (**Figure 3**).



2.1.2. Photothrombotic Stroke

To induce focal stroke in the right hemisphere, mice were injected with Rose Bengal (0.2 ml, 10 mg/ml solution in Phosphate Buffer Saline). Five minutes after intraperitoneal injection, a white light from an LED lamp was focused with a 20X objective and used to illuminate the primary motor cortex (0.5 mm anterior and 1.75 mm lateral from Bregma) for 15 min.

2.1.3. Electrophysiological Recordings on the M-Platform

Two healthy mice were used for the experiments. Animals were housed on a 12/12 h light/dark cycle. Mice were water deprived overnight before training on the platform; daily liquid supplement was given after the test. Food was available *ad libitum*. To have access to the motor cortex, a craniotomy was performed 3 days before the training to expose the Caudal Forelimb Area (CFA) of the right hemisphere. The craniotomy was filled with agarose and silicon (Kwik cast sealant, WPI) and could be opened and closed several times for acute recordings.

Mice were gradually acclimated to the platform. Then they performed the task for 2 days, fifteen trials each day. During the pulling experiment, mice were head fixed to the platform with their left wrist constrained to the slide. The friction on the slide was set at 0.3 N. The force signal was acquired by a load cell (Futek LSB200, CA, USA) along the direction of the movement at 100 Hz, at the same time a webcam recorded the position of the slide at 25 Hz and the multi-unit activity was recorded by Omniplex D System (Plexon, USA) with a frequency of 40 kHz thanks to a 16 channels linear probe (1 M Ω , ATLAS, Belgium) inserted into the CFA at 850 μ m of depth (Figure 3A).

2.1.4. Wide-Field Calcium Imaging of Cortical Activity During Training on the M-Platform

The mouse was housed in clear plastic cage under a 12 h light/dark cycle and was given *ad libitum* access to water and food. We used the following mouse line from Jackson Laboratories (Bar Harbor, Maine USA): C57BL/6J-Tg(Thy1GCaMP6f)GP5.17Dkim/J (referred to as GCaMP6f).

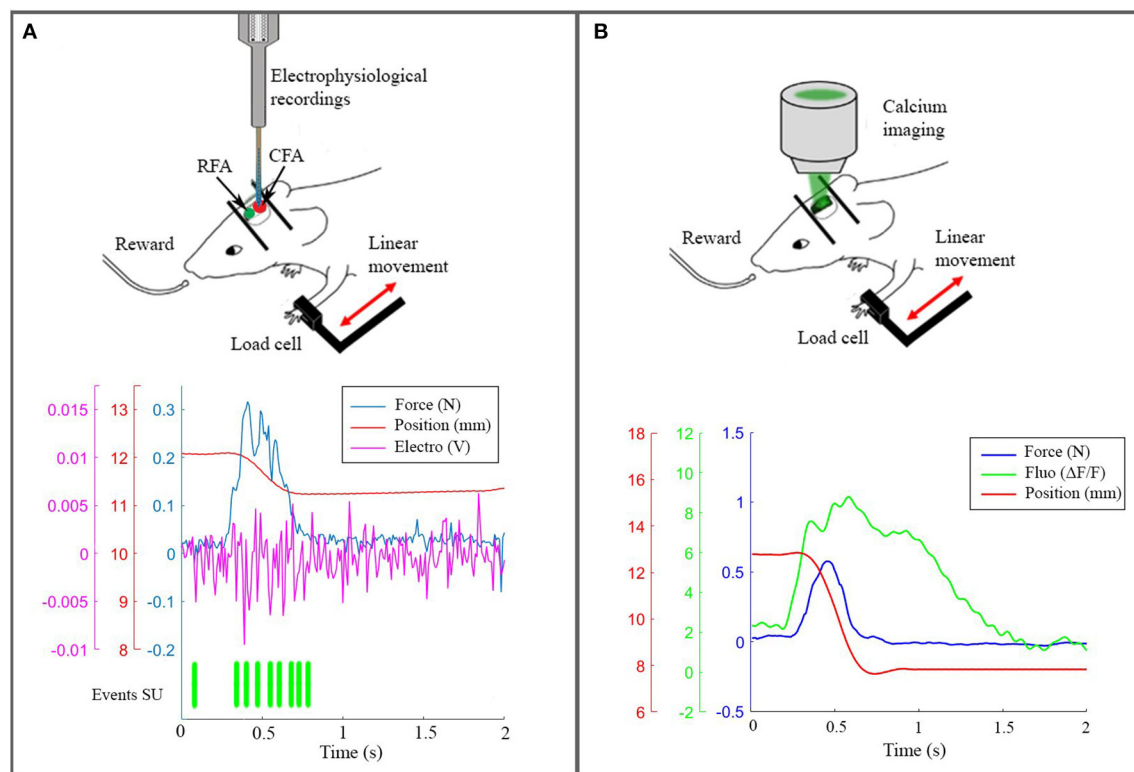


FIGURE 3 | The real robotic platform. **(A)** On the top a schematic representation of the experiment during electrophysiological recording in CFA. At the bottom the synchronized data: force peak (blue), movement of the slide (red), high frequency electrophysiological signal of a single channel (magenta) and the timestamp of a selected single unit. **(B)** On the top a scheme of the experiment with the setup to record calcium activity. At the bottom the recorded data after synchronization: force peak (blue), movement of the slide (red) and calcium response (green).

mice). In this mouse model, the fluorescence indicator GCaMP6f is mainly expressed in excitatory neurons (Dana, 2014). GCaMP6f protein is ultra-sensitive to calcium ions concentration (Chen and Kim, 2013; Dana, 2014) whose increase is associated with neuronal firing activity (Yasuda and Svoboda, 2004; Grienberger and Konnerth, 2012).

For wide-field fluorescence imaging of GCaMP6f fluorescence, we used a custom made microscope described in Conti et al. (2019). Briefly, the system is composed by a 505 nm LED (M505L3 Thorlabs, New Jersey, United States) light deflected by a dichroic filter (DC FF 495-DI02 Semrock, Rochester, New York USA) on the objective (2.5x EC Plan Neofluar, NA 0.085, Carl Zeiss Microscopy, Oberkochen, Germany). The fluorescence signal is selected by a band pass filter (525/50 Semrock, Rochester, New York USA) and collected on the sensor of a high-speed complementary metal-oxide semiconductor (CMOS) camera (Orca Flash 4.0 Hamamatsu Photonics, NJ, USA).

The experiment starts with a mouse being trained and recorded for 1 week (5 days) on the M-platform (“healthy” condition, see Figure 3). The focal stroke is then induced at the beginning of the second week by phototrombosis on the right primary motor cortex (rM1). Starting 26 days after stroke, the mouse performance and spontaneous motor remapping was

evaluated on the M-Platform for 5 days a week along 4 more weeks. The results from the first week 1 month after the injury is the so-called “stroke” condition, while the results during the last week, when the animal recovers the motor function is referred to as “rehab.”

Each day, the beginning of the wide-field imaging session was triggered by the start of the training session on the M-Platform. To detect the movement of the wrist of the animal in the low-light condition of the experiment, an infrared (IR) emitter was placed on the linear slide, and rigidly connected to the load cell and thus to the animal’s wrist. Slide displacement was recorded by an IR camera (EXIS WEBCAM #17003, Trust) that was placed perpendicular to the antero-posterior axis of the movement. Position and speed signals were subsequently extracted from the video recordings and synchronized with the force signals recorded by the load cell (sampling frequency = 100 Hz) and with the fluorescence signal recorded by the CMOS sensor (Figure 3B).

2.2. Data Analysis

2.2.1. Spikes and Force Analysis

Data were analyzed offline using custom routines in Matlab (MathWorks). First, the position signal was extracted by the video using a white squared marker on the slide as reference. The

recording frequency of the video was 25 Hz. After applying an antialiasing FIR lowpass filter, a uniform linear resample of the movement of the slide was performed, in order to synchronize the position signal with the force data, recorded at 100 Hz. To identify the timing of the voluntary activity of the animal, a threshold method was used to detect force peaks during the pulling phase of the task. For the following analysis, we picked out peaks that produced a displacement of the slide, in addition to crossing the threshold; and we calculate the onset of these peaks as the minimum of the force derivative just before the respective peak (Spalletti et al., 2014). The electrophysiological signal, recorded at 40 kHz as sampling rate, was analyzed by Offline Sorter (Plexon, Dallas, TX). First, for each channel of the probe, we sorted waveform that crossed a detection threshold of the mean ± 3 standard deviations. Then, detected spikes were clustered using an automatic process based on principal component analysis. Starting from these clusters, a manual sorting was executed to isolate all single units which could be identified in the recorded multi-units signal. The time stamp of each unit was synchronized with the data of the robot. To evaluate the temporal behavior-related spike activity, the peristimulus time histograms (PSTHs, NeuroExplorer, Plexon) was generated with bins of 20 ms in an interval of 1 s around the onset of force peaks. In addition, the resting activity of each unit was evaluated selecting intervals of at least 0.6 s with no force peaks and calculating the average of the number of spikes in bins of 20 ms. Finally, the PSTHs was used to evaluate when a single neuron was active, that is when the number of spikes for bin cross the threshold, calculated as the mean ± 2 standard deviations of the number of spikes for bin during the respective resting activity.

2.2.2. Phase Coherence and Functional Connectivity

Functional connectivity (FC) among cortical regions was inferred from phase coherence of activity measurements, and used to determine changes in brain activity in “stroke” and “rehab” condition, as compared to the healthy mice. These inferred activity changes were used to parameterize simulations of the BNM built over the Allen Brain Atlas mouse connectivity data (<http://connectivity.brain-map.org/>; Oh et al., 2014, below referred to as the Allen Mouse Brain Atlas - AMBA), incorporated in the extended virtual mouse brain (Melozzi et al., 2017). In each animal, the camera field-of-view used for activity measurements was placed in a standard position using the sagittal suture and its intersection with the coronal suture of the skull (bregma) as anatomical landmarks. To spatially correlate our activity measures with the structural connectivity data (Oh et al., 2014), the camera field-of-view (**Figure 3B**) was spatially translated to the Allen Common Coordinate Framework (CCF, v3, 2015; Wang et al., 2020). Since the CCF lacks stereotactic skull landmarks, these were introduced by spatially co-registering all diagrams from a standard stereotaxic mouse brain atlas (Franklin et al., 2008) to the CCF coordinate space with affine transformations defined using the QuickNii tool (Puchades et al., 2019). Using bregma and the sagittal suture as a reference, the four corners of the downsampled 128x128 pixels field-of-view of the recorded images were positioned in CCF, taking the 5 degree lateral tilt of the camera view into account. Delineations of layer

IV cortical regions were then projected onto the camera field-of-view, and used as a custom atlas reference for all activity maps.

The spectral content of the signals is analyzed to identify the frequency band which captures the spontaneous brain activity that occurs simultaneously with the motor-evoked events. The time-frequency analysis of the calcium recordings is limited by their sampling rate and the length. The former makes most of the activity at faster frequency bands inaccessible, but still allows analysis of the slow oscillations up to 5 Hz, which have been often associated with the spontaneous brain activity (Vanni et al., 2017; Wright et al., 2017). Even though the slowest dynamics <0.5 Hz, which has the highest power, is often a marker of the resting state (Wright et al., 2017), in this experiment it also contains the propagation of waves generated during the limb movements on the platform. The mechanisms behind stimulation propagation (Spiegler et al., 2016) are different from the spontaneous oscillations at rest (Deco and Jirsa, 2012) that we try to study and model here, and hence the lowest frequencies are excluded from the analysis. In addition, the mice heart rate is between 6 and 8 Hz, whilst the activity above 10 Hz is too close to the Nyquist frequency of 12.5 Hz, defined as half of the sampling rate of the recordings. As a consequence these bands are generally avoided in the analysis of calcium signals, which is consequently often centered at the δ band between around 1 and 5 Hz (Vanni et al., 2017; Wright et al., 2017).

The FC is characterized with the phase coherence of the analytical phases of the band-passed time-series obtained using the Hilbert transform (Pikovsky et al., 2001). For this we employ phase locking values (PLV) (Lachaux et al., 1999) that are a statistical measure for similarity between the phases of two signals, hence defined as

$$PLV_{ij} = \left| \frac{1}{M} \sum_{m=1}^M e^{i(\theta_i(m) - \theta_j(m))} \right|, \quad (1)$$

where the phase difference $\theta_i(m) - \theta_j(m)$ between the regions i and j is calculated at times $m = 1 \dots M$. The same procedure is also applied to surrogate time-series to find the level of statistically significant phase coherence (Lancaster et al., 2018).

3. MODELS

3.1. Spinal Cord Model

To develop the final model, an incremental approach was followed, starting from a circuit for a single *muscle*, adding inhibitory connections between *antagonistic* pairs and finally interneurons to modulate descending stimuli (**Figure 4**).

For a single muscle, a network with muscle spindles providing Ia and II afferent fibers activity, a pool of α -motoneurons and excitatory II-interneurons was considered (Stienen et al., 2007; Moraud et al., 2016). Ia afferents directly provide excitatory inputs to the α -motoneurons (monosynaptic stretch reflex mechanism), while the II afferents output is mediated by a set of interneurons before reaching the α -motoneurons, creating a disynaptic reflex. The muscle spindles are implemented using the model from Vannucci et al. (2017). All other neurons are modeled as leaky integrate and fire neurons. The

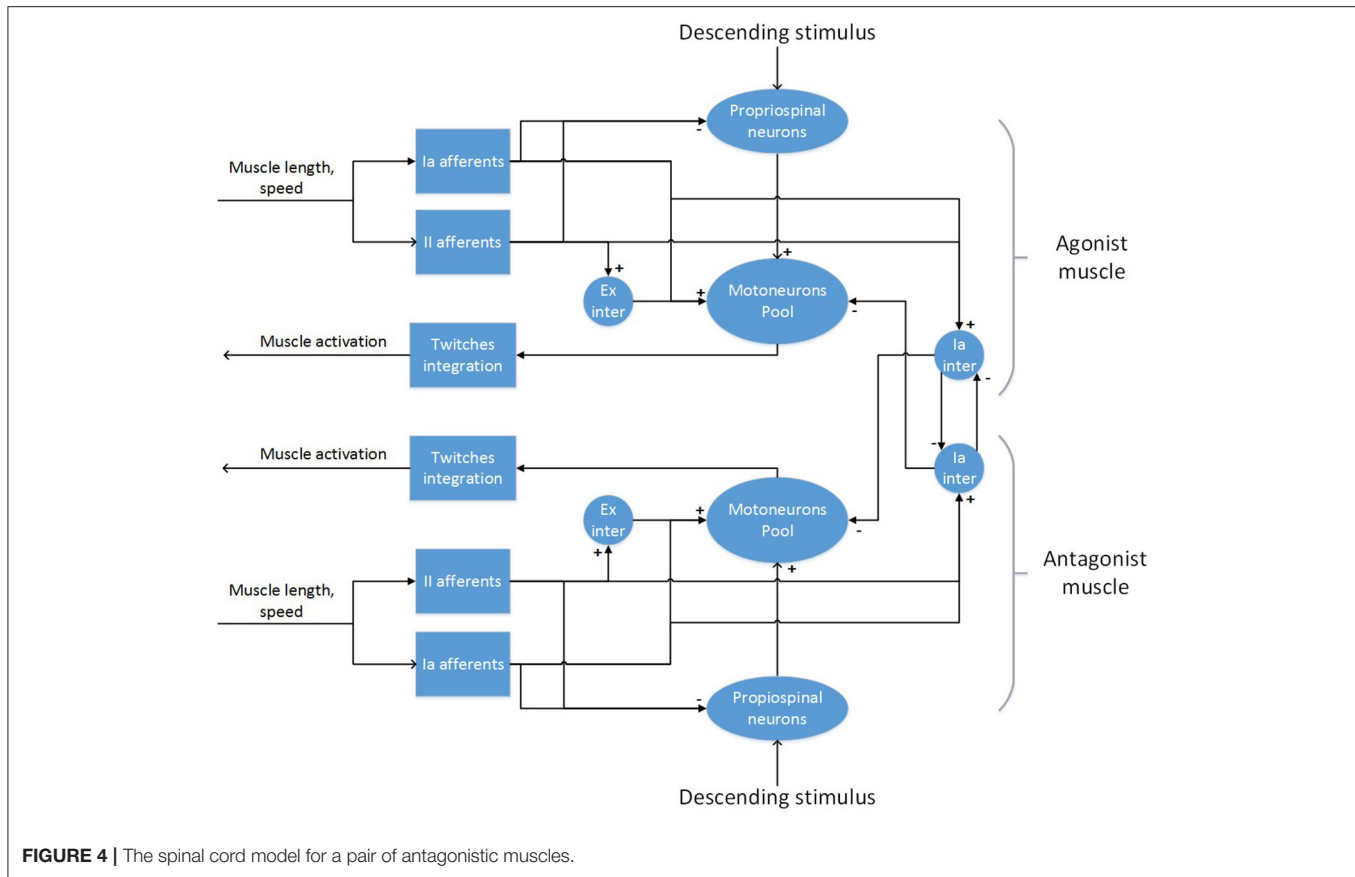


FIGURE 4 | The spinal cord model for a pair of antagonistic muscles.

number of neurons in the spinal cord populations, as well as parameters for the synaptic connections are taken from Moraud et al. (2016), with the exception of the synaptic weights of the monosynaptic connections, which have been significantly lowered (see **Supplementary Material**). The parameters from the muscle spindle models are taken from Mileusnic et al. (2006), which are tuned on neurophysiological recordings of lower mammals. Distribution of parameters for the α -motoneurons that influence the recruitment order and fiber strength (membrane capacitance, membrane time constant, maximum twitch force, time to peak force) are taken from Sreenivasa et al. (2016):

$$D_i = [d_{max} - d_{min} \cdot \log(N - i)] \cdot D_{SF} \quad (2)$$

$$C_i = \pi D_i^2 \cdot c_{spf} \quad (3)$$

$$\tau_i = \tau_{max} - (D_i - \tau_{adj}) \cdot \tau_{slp}$$

$$F_i = [p_{max} - p_{min} \cdot \log(N - i)] \cdot F_{SF}$$

$$T_i = \left[s_{min} - \frac{s_{sl}}{N} i \right] \cdot T_{SF} + s_{min}$$

where i is the index of the α -motoneuron in the pool, N is the size of the pool and the others are free parameters that can be adjusted for every muscle. In this work, the value of these parameters has not been changed from Sreenivasa et al. (2016).

In order to compute the actual muscle activation from the motoneurons activity, a special spike integration unit that sums

the fibers twitches was implemented. The spikes were integrated using the discrete time equations of Cisi and Kohn (2008) with a non-linear scaling factor from Fuglevand et al. (1993) that prevents the activation to grow indefinitely:

$$a_i(t) = 2e^{-\frac{\delta t}{T_i}} \cdot a_i(t-1) - e^{-\frac{2\delta t}{T_i}} \cdot a_i(t-2) + F_i \cdot g(t) \cdot \frac{\delta t^2}{T_i} e^{-\frac{1-\delta t}{T_i}} \cdot u(t) \quad (4)$$

where δt is the integration time, and $u(t)$ and $g(t)$ are the spike function and the non-linear scaling, defined as:

$$u(t) = \begin{cases} 1 & \text{if a spike is received at } t \\ 0 & \text{if no spikes are received at } t \end{cases} \quad (5)$$

$$g(t) = \begin{cases} 1 & \text{if } T_i/ISI_i < 0.4 \\ \frac{1 - e^{-2(T_i/ISI_i)^3}}{T_i/ISI_i} & \text{otherwise} \end{cases} \quad (6)$$

where ISI_i is the observed inter-spike interval of α -motoneuron i . Moreover, the activation can be scaled between 0 and 1 by dividing by the maximum theoretical value:

$$a_{i,max} = \lim_{t \rightarrow +\infty} a_i(t) = F_i \frac{\frac{\delta t^3}{T_i^2} \left(1 - e^{-2\left(\frac{T_i}{\delta t}\right)^3} \right) \cdot e^{-\left(1 - \frac{\delta t}{T_i}\right)}}{1 - 2e^{-\frac{\delta t}{T_i}} + e^{-2\frac{\delta t}{T_i}}} \quad (7)$$

Therefore, the output of the twitch integration module is an activation value in $[0; 1]$ that is suitable for the muscle model present on the mouse virtual embodiment. The effect of this integration is that at low frequencies the individual twitches can still be seen, while at higher stimulation the twitches fuse into a tetanic contraction. Moreover, thanks to the non-linear scaling, the activation reaches a maximum value and higher stimulation frequencies do not produce any effect, in accordance with the contractile properties of real muscle fibers.

In order to implement the polysynaptic inhibition reflex between antagonistic muscles, two populations of Ia-interneurons were added to the network. Those receive inputs from all Ia afferents of a synergistic muscle and provide inhibition to the α -motoneurons of the corresponding antagonistic muscle. Moreover, as the activation of a muscle should provoke an inhibition of its antagonist (Pierrot-Deseilligny and Burke, 2005), the Ia-interneurons also receive low-gain positive inputs from the corresponding descending pathways. Again, the number of neurons in these population and their parameters have been taken from Moraud et al. (2016). Finally, as there is lack of evidence for a direct connection between cortical neurons and motoneurons in the spinal cord of rodents (Yang and Lemon, 2003), an intermediate population of neurons mediating descending signals was added to the circuitry. This population aims at modeling propriospinal neurons, which provide an inhibitory action on the signals coming from the corticospinal tract (Alstermark, 1992). In general, the inhibition is generated from different peripheral afferents, but we included only afferents from muscle spindles as these are the only present in the model. As there is no definitive experimental evidence on the size of the population of propriospinal neurons and its parameters, we set the values of these to those of the populations of Ia-interneurons. Conversely, the synaptic weights and the number of connections between the descending inputs and the propriospinal interneurons were empirically tuned starting from experimental data.

3.2. Simulation Tools and Physical Models

This section describes simulation tools that were used to synchronize neural and physical simulations and the physical simulations models that have been developed and used. These tools and model were used in the context of the *Movement-driven models* pipeline.

3.2.1. Embodied Mouse in the Neurorobotics Platform

The full musculoskeletal model of the virtual rodent controlled by the spinal cord model was simulated in the Neurorobotics Platform (NRP) developed in the Human Brain Project (Falotico et al., 2017). The main components of the NRP are a world simulator, a brain simulator and the mechanism that enables the data flow between the two in a closed-loop. The connection between the body and the brain is specified through a domain specific language (Hinkel et al., 2015, 2017), via Python scripts called Transfer Functions. In these scripts the output of devices that read neuronal output data can be processed and passed as input for the virtual body actuators, and vice versa, the sensory information from the virtual body sensors, in this case muscle

length data, can be passed to devices that map sensory data to neural input. The brain simulation, which currently is simulating point-neurons, follows closely the paradigm of NEST (Gewaltig and Diesmann, 2007), interfaced through PyNN (Davison et al., 2009). On the other side of the closed loop the world simulator of choice is Gazebo (Koenig and Howard, 2004), extended to support muscle simulation through OpenSim (Millard et al., 2013), which provides its' own muscle simulation engine.

3.2.2. Musculoskeletal Embodiment

As described earlier the musculoskeletal system comprises of two elements, the skeletal and the muscle system respectively. Here both systems are elaborated a bit more in the context of the experiment. Developing animal skeletal systems is no trivial task. It involves many complex degrees of freedom and physical properties such as mass, center of mass and inertias. To ease this process, NRP has developed a toolkit for Blender (Open source modeling and animation tool) called RobotDesigner (HBPNeurorobotics, 2019). RobotDesigner allows to automate several steps needed to develop skeletal/robot models to be simulated in the NRP. Using the same, currently NRP hosts state-of-the-art a full skeletal model of the mouse consisting of 110 degrees of freedom. More details about the full model will be soon published following the current article. For the current experiment, the mouse skeletal model is reduced in complexity by constraining all the degrees of freedom except the left forelimb. The forelimb consists of four segments and it is further constrained to only have flexion-extension movements, enough to reproduce the passive extension-active retraction experiment on the M-Platform. The different segments and the joints of the forelimb are shown in **Figure 5**.

The physical properties of the skeletal system such as mass, center of mass and inertia are automatically estimated based on bounding objects generated for each link (segment) using the RobotDesigner. Once the skeletal system is established, muscles-tendon system can be attached to the bones. As mentioned before, NRP now supports OpenSim for integrating muscle models into physical animal bodies or even robots. In the current experiment a pair of antagonist hill-type muscles were added to each of the joints in the mouse forelimb. The muscle model in OpenSim is taken from Millard et al. (2013) (see **Supplementary Material**). Again RobotDesigner offers a unique solution to visualize attachments and easily add muscles to the body in blender. Using the same technique all the muscles for the mouse forelimb were added. Muscle parameters used in the current experiment are hand tuned to produce flexion-extension movements necessary for the experiment. **Figure 5** (right panel) shows the muscle attachments used in the current model.

3.2.3. Robotic Rehabilitation Platform Model

In the real experiment, the mouse forelimb is attached to the sliding mechanism, which is a prismatic joint, driven by a DC motor whose rotational motion is converted into a linear one. The motor that is controlled with a PID controller, whose reference can be set to a position of the joint between the minimum and maximum positions. The controller is enabled when the operator decides to replace the sled in its starting

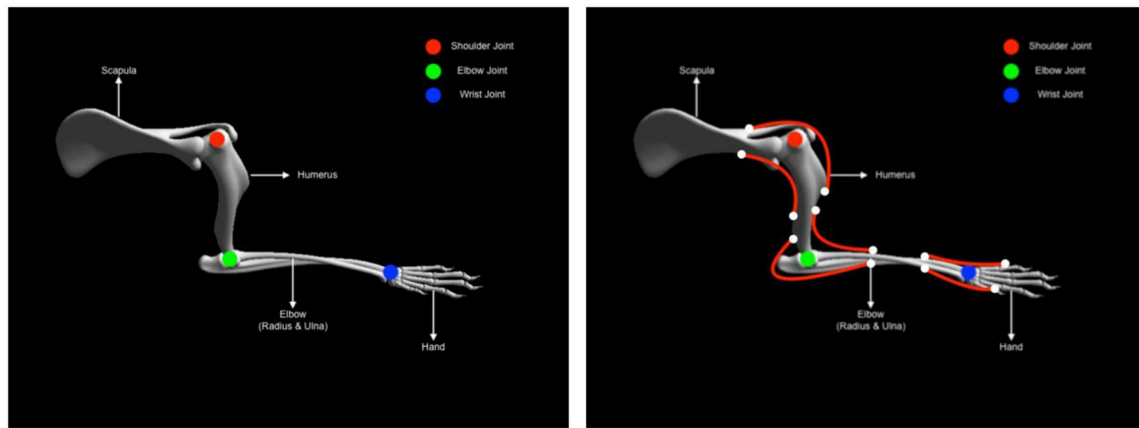


FIGURE 5 | Mouse forearm musculoskeletal system **(left)** Forelimb skeletal system with three joints (1) shoulder (Ball and socket) (2) elbow (hinge) (3) wrist (hinge). **(right)** Forelimb muscle system with six muscles (1) humerus-extension (2) humerus-flexion (1) elbow-extension (2) elbow-flexion (1) hand-extension (2) hand-flexion.

position and is disabled afterwards, so that the mouse can actually pull the sled. In simulation, the same configuration has been implemented. The musculoskeletal mouse forelimb was attached to a simulated M-platform, which has been modeled as a prismatic joint, controlled with a PID controller whose output is directly applied as a simulated force on the joint, assuming ideal actuator transfer behavior. Again, the reference to the PID controller is the position of the prismatic joint in its range, this time normalized between 0 (minimum) and 1 (maximum). To simulate the intervention of the operator that puts the slide back we employed a state machine that automatically controls the slide, by making use of the PID controller setting 1 as a reference. Inputs to this state machine are a list of times at which the slide should be put back. Conversely, to simulate the minimum amount of force that is required to move the slide in the real setup, we deactivated the PID controller in simulation only after a certain activation of the simulated muscles was reached (0.95).

3.3. Stroke Models

3.3.1. Brain Network Model With Kuramoto Oscillators

To simulate the functional network reorganization during stroke and recovery given by the phase coherence of the macroscopic brain activity reflected in calcium signals, we built our BNM based on Kuramoto oscillators for the local oscillatory dynamics and the AMBA connectome that dictates the strength of the couplings between brain regions (Melozzi et al., 2017; Choi and Mihalas, 2019), **Figure 6**, and has been validated with empirical functional data that justifies its use (Melozzi et al., 2019). The AMBA contains 86 cortical regions (43 per hemisphere), of which 18 were included in the field-of-view (**Figure 6**, bottom left). The average calcium signal of the pixels entirely located within a brain region was used to represent their mean neuronal activity.

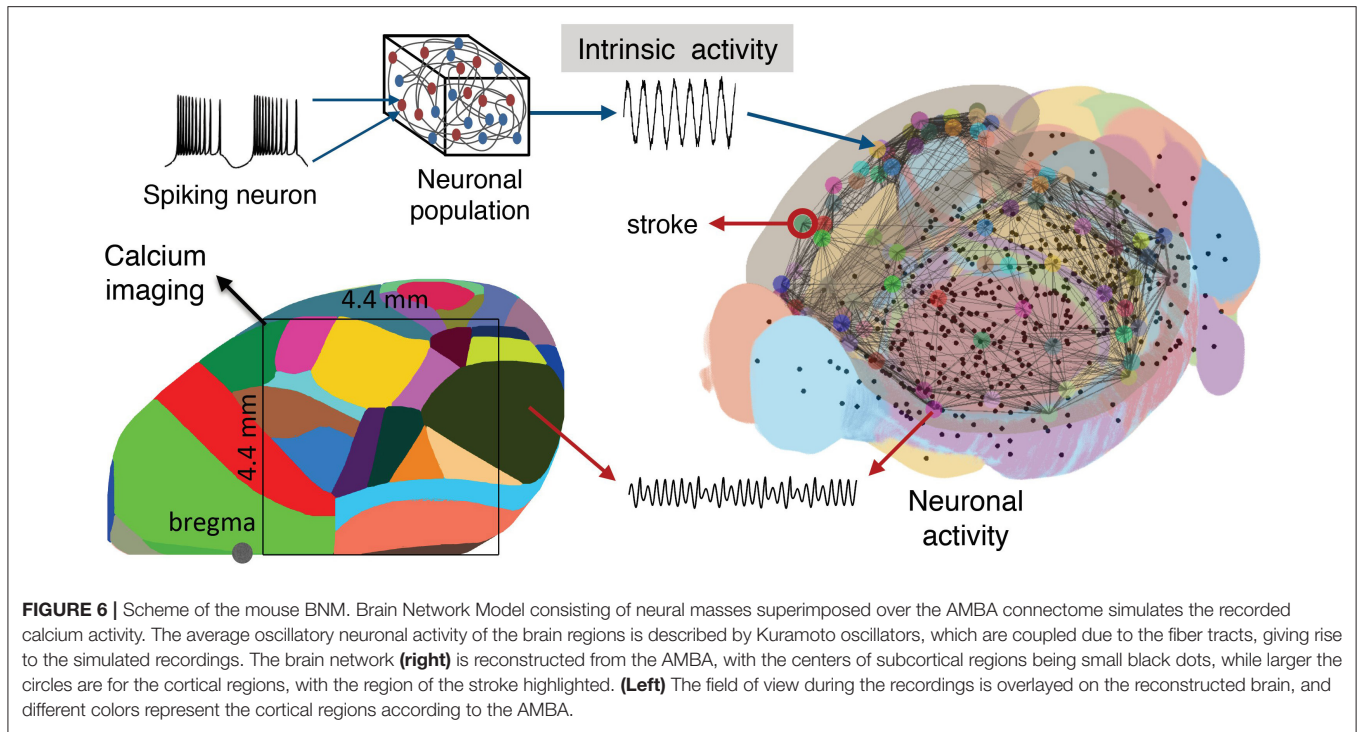
Besides their simplicity, phase models exhibit rich dynamics and a direct link to more complex biophysical models, while admitting analytic approaches (Roy et al., 2011; Sheppard et al., 2013; Ton et al., 2014; Stankovski et al., 2016). KM

(Kuramoto, 1984), as a phenomenological model for emergent group dynamics of weakly coupled oscillators (Pikovsky et al., 2001) is well-suited for assessing how the connectome governs the brain oscillatory dynamics that can be reflected in different neuroimaging modalities (Schmidt et al., 2014; Váša et al., 2015; Cabral et al., 2017; Petkoski et al., 2018). The constructed BNM is thus used to identify the structural alterations due to the stroke and the subsequent recovery, using their causal effects on the functional changes captured by the calcium recordings of the cortical brain activity.

Even though delayed interactions due to axonal transmission can be of crucial importance for the observed dynamics of the oscillatory systems (Ghosh et al., 2008; Petkoski et al., 2016, 2018), the impact of these delays is much less pronounced for low frequencies compared with them, as it is the case here. Moreover, the tracing used for obtaining the AMBA Connectome (Oh et al., 2014) does not allow tracking the length of the white fibers. Hence, we assume instantaneous couplings and the utilized model gives the following evolution of the phases for each of the N brain regions

$$\dot{\theta}_i = 2\pi f + \frac{1}{N} \sum_{j=1}^N K_{ij} \sin(\theta_j - \theta_i) + \eta_i(t), \quad i = 1 \dots N. \quad (8)$$

Here the dynamics of each region i is driven by the natural frequencies f that are assumed to be identical across the brain. A stochastic variability is introduced with additive Gaussian noise defined as $\langle \eta_i(t) \rangle = 0$ and $\langle \eta_i(t) \eta_j(t') \rangle = 2D\delta(t - t')\delta_{ij}$, where D is the noise strength and $\langle \cdot \rangle$ denotes time-averaging. The activity of the BNM is then constrained by the structural connectivity, which for every region is represented by the inputs that they receive from the other regions j through the coupling strength $K_{ij} = Kw_{ij}$. This contains the structural weight of the connectome between these areas, w_{ij} , scaled with the same global coupling K for every link.



3.3.2. Spiking Network Model for Simulation of Slow Wave Activity in Peri-Infarct Cortex

Besides the phenomenological neural mass model for the oscillatory activity that we have used in the BNM, we also show an alternative spiking neural model to reproduce local brain activity in the acute phase after stroke. In future, this model should be integrated in the BNM and therefore in the *Embodied brain* closed-loop simulation, either by deriving its mean-field representation, e.g., see Zerlaut et al. (2017), or by co-simulation. This model aims to reproduce the two-photon calcium signals of a population of spiking neurons located at the peri-infarct area, since it is known that slow frequency patterns of synchronized activity emerge from the damaged areas after an ischemic stroke (Carmichael and Chesselet, 2002; Butz et al., 2004; Rijdsdijk et al., 2008; Rabiller et al., 2015).

Network of adaptive exponential integrate and fire (adex) neurons

The network consists of an excitatory (regular spiking, RS) and inhibitory (fast spiking, FS) population of neurons (**Figure 7A**). All cells are modeled as adex neurons, which can be described by the following equations:

$$\begin{cases} C_m \frac{dV(t)}{dt} = G_l(E_l - V(t)) + G_l \Delta_V e^{\left(\frac{V(t) - V_{thre}}{\Delta_V}\right)} + I_{syn}(t, V(t)) \\ \quad - w(t) + \sigma \xi(t) \\ \frac{dw(t)}{dt} = -\frac{G_l}{C_m} w(t) + b \sum_k \delta(t - t_k) + a(V(t) - E_l) \end{cases} \quad (9)$$

where the synaptic input I_{syn} is defined as

$$I_{syn}(t, V(t)) = \sum_i g_i^{syn}(t)(V(t) - E_i^{syn}) \quad (10)$$

with

$$\frac{dg_i^{syn}(t)}{dt} = -g_i^{syn}(t)/\tau_{syn} \quad (11)$$

Here, $G_l = 10 \text{ nS}$ is the leak conductance and $C_m = 150 \text{ pF}$ is the membrane capacitance. The resting potential, E_l , is -60 mV or -65 mV , for excitatory or inhibitory cells, respectively. Similarly, the steepness of the exponential approach to threshold, Δ_V is 2.0 mV or 0.5 mV , for excitatory or inhibitory cells, respectively. When the membrane potential V reaches the threshold, $V_{thre} = -50 \text{ mV}$, a spike is emitted and V is instantaneously reset and clamped to $V_{reset} = -65 \text{ mV}$ during a refractory period of $T_{refrac} = 5 \text{ ms}$. The membrane potential of excitatory neurons is also affected by the adaptation variable, w , with time constant $\tau_w = 500 \text{ ms}$, and the dynamics of adaptation is given by parameter $a = 4 \text{ nS}$. At each spike, w is incremented by a value b , which regulates the strength of adaptation. $b = 60 \text{ pA}$ was used to model deep anesthesia, and $b = 20 \text{ pA}$ for light anesthesia simulations.

From spikes to fluorescence of two photon signal

In order to model the two-photon calcium signal the spikes and the values of the membrane potential V_m (**Figure 7C**, gray trace) of each neuron were recorded during the simulation, for each level of adaptation. Increases in the V_m lead to an inward calcium current through voltage-dependent channels. We characterized the L-type high voltage activated calcium current I_{Ca} (**Figure 7C**, red trace) as in Rahmati et al. (2016):

$$I_{Ca} = g_{Ca}s(V_m - E_{Ca}) \quad (12)$$

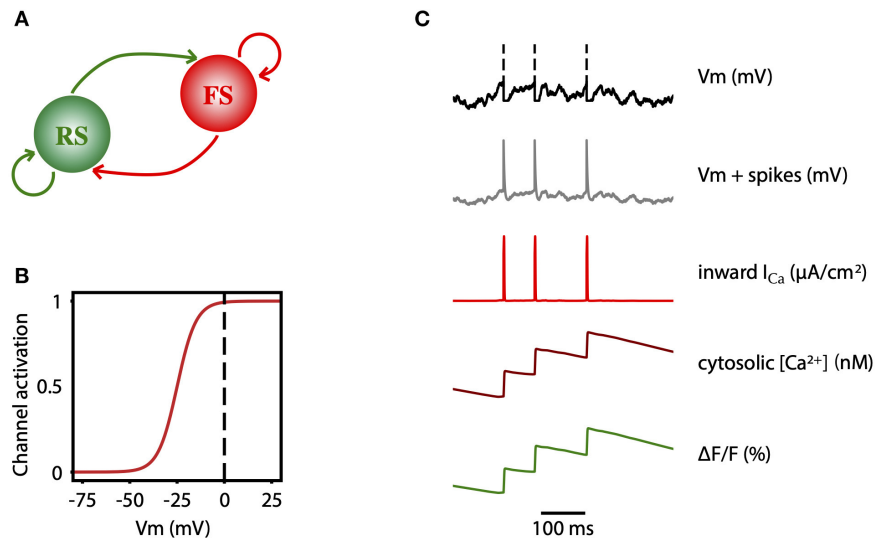


FIGURE 7 | Two-photon calcium signal model from a spiking network model. **(A)** Schematic connectivity between the excitatory population of regular spiking (RS) neurons and the inhibitory population of fast spiking (FS) neurons of the modeled cortical network. **(B)** Activation curve of a high-voltage activated calcium channel that is used to compute the inward calcium current (I_{Ca}) from the changes in the V_m . **(C)** From top to bottom, simulated membrane potential of a neuron emitting three spikes which are represented by dashed lines, membrane potential with reconstructed spikes, inward calcium current associated with changes in the membrane potential, cytosolic calcium concentration, and fluorescence emitted by the calcium indicator due to the intracellular concentration of calcium.

where $E_{Ca} = 120$ mV and $g_{Ca} = 5$ mS/cm² are the reversal potential and the maximal conductance of this current, respectively. The steady-state voltage dependent activation of the channel (**Figure 7B**), is defined by the Boltzmann function:

$$s = \frac{1}{1 + \exp\left(-\frac{(V_m - V_{1/2})}{\rho}\right)} \quad (13)$$

with a half-activation voltage $V_{1/2} = -25$ mV and a slope factor $\rho = 5$ mV (Ermentrout, 1998; Helton et al., 2005). The intracellular concentration of calcium (cytosolic $[Ca^{2+}]$, **Figure 7C**, dark red trace) increases proportionally to the I_{Ca} current, and then it slowly decays back to a basal value $[Ca^{2+}]_i$ (Traub, 1982):

$$\frac{d[Ca^{2+}]}{dt} = -k_{Ca}I_{Ca} - \frac{[Ca^{2+}] - [Ca^{2+}]_i}{\tau_{Ca}} \quad (14)$$

with $K_{Ca} = 0.002$ (nM/ms)(μA/cm²)⁻¹, $\tau_{Ca} = 760$ ms and $[Ca^{2+}]_i = 0$ nM.

Finally, the fluorescence $F(t)$ associated with the intracellular calcium concentration (**Figure 7C**, green trace) is then computed following the equation:

$$F(t) = dF + K_F \frac{[Ca^{2+}]^{n_H}}{[Ca^{2+}]^{n_H} + K_d} \quad (15)$$

where $dF = 0$ and $K_F = 10$ are the offset and the scaling of $F(t)$, $K_d = 375$ nM is the dissociation time constant for GCaMP6f (Chen and Kim, 2013), a measure of the affinity of the fluorescent indicator to the calcium ion, and $n_H = 2.3$ is the Hill coefficient (Chen and Kim, 2013).

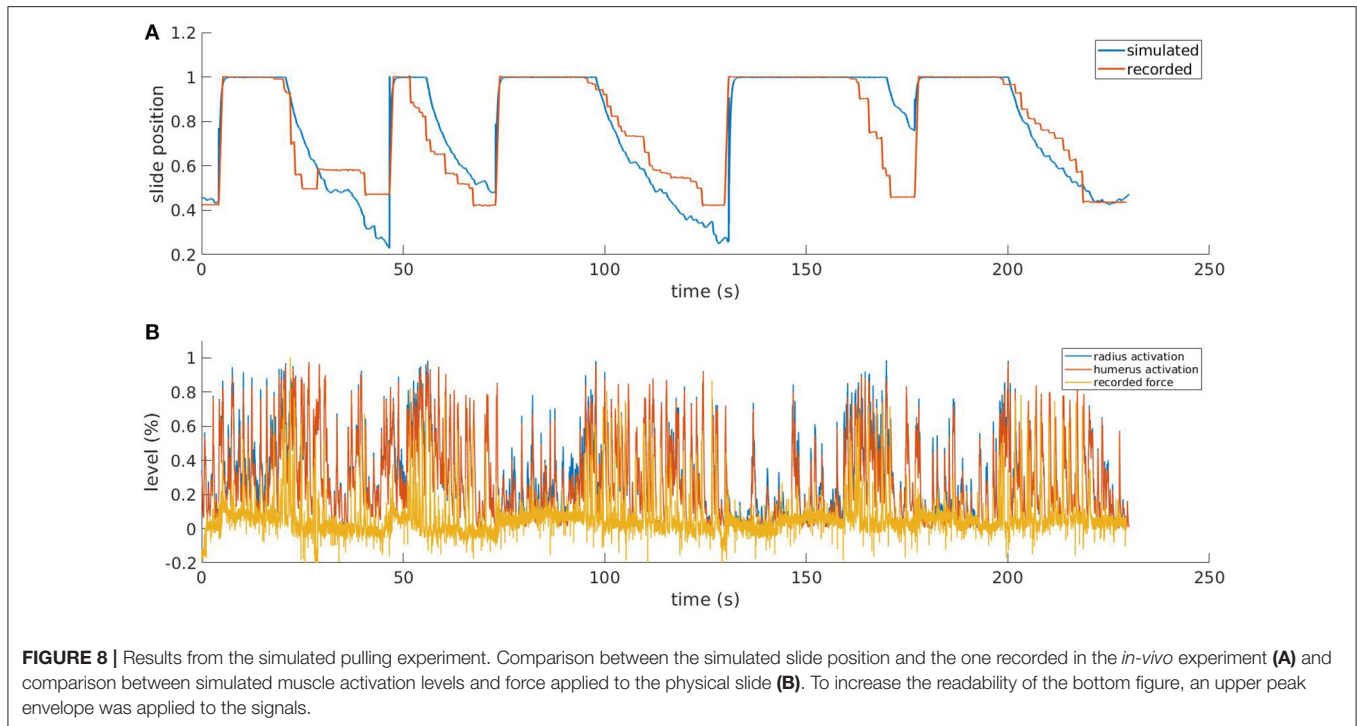
4. RESULTS

Here we show first the results we obtained on the simulation of goal-directed movements ("Movement-driven models" pipeline) and then on the modeling of brain alterations after stroke ("Stroke models" pipeline).

4.1. Simulation of the Experiment on Goal-Directed Movements

As the first component of the proposed framework ("Movement-driven models"), we simulated the experiment on goal-directed forelimb pulling in the virtual environment and validated the simulation on experimental data.

In the *in-vivo* experiment, two healthy mice were trained on the M-Platform to perform active pulling of the forelimb. As we expected, the contralateral motor cortex showed a highly coherent activation with the kinetic data. The coherence between the force applied by the animal and the signal recorded in the CFA was evident both in the low and in the high frequencies band (**Figure 8**). For the data that were later used in simulations we focused on the high band (300 to 40k Hz); in particular we found an high activation of the motor cortex around the force peaks for both multi-unit activity and single units analysis. This result proves that for each recording the SUs were successfully extracted by the multi-units. The PSTHs was used to evaluate the temporal behavior-related spike activity of every single unit. The behavior around the force peaks was different according to the single units selected, but all of them showed that the activity began to increase before onset of force peaks and came back to the resting value after 0.4 s from the onset. In order to simulate the descending signal from the motor cortex generating the movement of the



forelimb, we employed these neurophysiological recordings. In particular, the events resulting from the single unit spike sorting were given as spike times for static spike generator in the neural simulation. As the number of recorded neurons was low, the spike generators were copied 100 times, while also adding gaussian noise (with mean = 0ms and standard deviation = 5ms) to the spike times of the copies to avoid synchronicity. As the neural recording originates mainly from neurons that control the pulling, we decided to connect the descending stimuli only to interneural populations associated with muscles that are active during the pulling, i.e. the flexors of the two actuated joints. Therefore, the antagonistic muscles would only actuate thanks to spinal reflexes. To tune the parameters of these connections, and to produce a muscular activation that was similar in amplitude to the force recorded in the *in-vivo* experiments, we performed a preliminary set of experiments, without the simulated embodiment, in which we empirically tuned the synaptic weights and number of connections. Due to the absence of the embodiment, at this stage there is no muscle spindle activity and thus no sensory feedback enabling reflexes. Then, the spinal cord model described in section 3.1 was connected to the mouse forelimb. In principle, the musculoskeletal embodiment has three pairs of muscles, but the one controlling the paw is not significantly involved in the pulling of the limb. We did not consider those when building the neural network to decrease simulation times. Thus, we replicated the same spinal cord circuitry two times and connected it to the four muscles controlling the elbow and shoulder joints, named *humerus* and *radius* in the simulation. In the closed loop simulated by the Neurorobotics Platform, the output of the spinal cord model (muscles activation between 0 and 1) could be directly given to the simulated mouse actuators, while the muscle lengths and

contraction speed had to be normalized before sending them to the muscle spindles models.

In **Figure 8**, we show the results for a simulation trial and a comparison with data recorded from a physical experiment. We employed kinematic data recorded alongside neural activity in the same *in-vivo* experiment: position of the slide and force applied to the slide through the trial. As expected, by comparing the activation levels with the normalized force applied by the mouse to the slide we can observe that the flexor muscles are active when there is also a force recorded, and conversely, there is low activation when the slide is still. It is also worth mentioning that, although the two muscles receive the same inputs from the descending stimuli, their activation levels are different due to the feedback circuitry of the spinal cord and the activity of muscle spindles, which are different for the two muscles. Thus, the output of the spinal cord circuitry is not a mere filtering of the input signals, but it also takes into account the feedback from the embodiment, which can change during the experiment. This effect underlies the importance of embedding neural circuits in a proper, realistic embodiment. The comparison between the simulated slide position and the recorded one shows that, thanks to the recorded neural activity, the muscles are able to overcome the force threshold and release the slide, and that an actual pulling is performed. Every pulling episode in the trial is reproduced, even if with different degrees of accuracy. Overall, the mean absolute error between simulated and recorded slide positions is 13%. The main discrepancies between the simulated and the recorded data come from the fact that in the simulated mode, the muscular activity is mostly directly proportional to the neural activity, while in the recorded data this is not always the case. While there is clearly a correlation between presence of neural activity and motion, the intensity

of such activity sometimes does not match the intensity of the motion.

4.2. Local and Global Brain Simulations After Stroke and Rehabilitation

4.2.1. BNM for Brain Connectivity Changes After Stroke and Rehabilitation

Within the second pipeline of the framework (“Stroke models”), in this subsection we simulated different extents of the brain injury and rehabilitation-induced plasticity after stroke. The results from the simulations are compared with the experimental data (Figure 2), allowing us to find the best fit with the empirical functional reorganization in the parameters space of the structural changes in the white matter connectivity.

The averaged calcium activity has a local peak in the power spectrum at around 3.5 Hz (Figure 9A), which is within the band relevant for the resting state activity. We hence focused the analysis of the experimental data on the upper delta band, 2.5 – 5 Hz, where we consequently band-pass filter the signals. From these we calculated the pair-wise PLV in each condition, thus constructing the FC matrices for the cortical regions of interest. Finally, to remove one condition, we calculate the changes of the FC during stroke and recovery compared with the healthy state, and this is the data feature that is then compared with the simulated data. For this we use the model described in section 3.3.1, to identify which scenarios of structural alterations cause the best agreement with the data in the modeled FC alterations (Figure 9). To minimize the effect of tissue displacement after stroke (Brown et al., 2007), the analysis includes only 12 ipsilesional regions located outside the stroke core (Figures 9B,D).

The stroke affects not only the inherent activity of the rM1, but all the connected regions. However, the precise breadth and magnitude of the structural damage, namely which links and to what extent are they disabled over time, is unknown. Similarly, it is not known which new links are created or reinforced during the spontaneous recovery or what is modulated by the rehabilitation. On the other hand, the stroke was shown to consistently change the alignment of dendrites and axons toward the core *in vivo* (Brown et al., 2007), possibly meaning altered SC, confirming previous works on structural rewiring after stroke (Dancause, 2005; Nudo, 2013). Hence a numerical exploration of the different possibilities of the stroke and rewiring in the large-scale BNM is used to unveil the most probable structural alterations associated with stroke and recovery. The calcium activity in the upper delta band that was chosen for the analysis shows highly coherent co-activation of different parts of the cortex, compared with the surrogate time-series (Figure 9A). We compared the functional reorganization associated with spontaneous recovery after stroke (“stroke” group) to rehabilitation-supported recovery (“rehab” group). The changes in the functional connectivity in “stroke” compared to “healthy” mice (Figure 9B, left matrix) indicate an increased co-activation of all but one somatosensory areas in the chronic phase after stroke, while visual areas have increased connectivity with all the regions, and reduced with the retrosplenial cortex. In the rehabilitated mice (Figure 9B, right matrix), the increase in connectivity of the somatosensory is

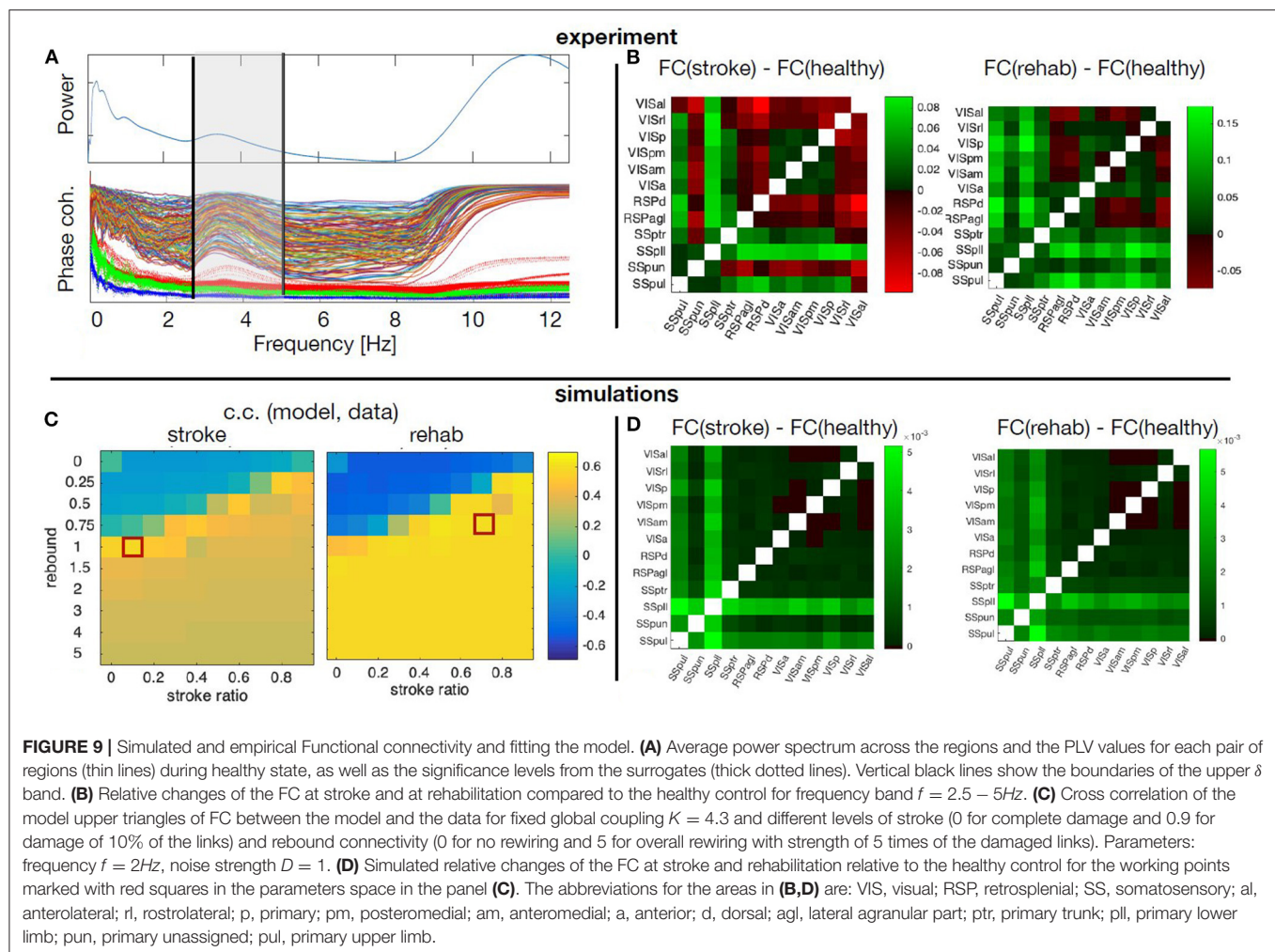
even higher across all the areas, and there is also an increased FC of the visual areas between each other and with the somatosensory regions.

A phenomenological neural mass is used to simulate how ipsilesional FC is changed by stroke and rehabilitation based on the modifications of the SC. For this, we systematically modified the SC to account for various impacts of stroke and subsequent recovery, in order to find the best match with the patterns observed in the data. The damage due to the stroke is assumed to be homogeneous across the links connecting rM1, but their magnitude is varied from 10 to 100%. Similarly, after the recovery it is assumed that 0 to 500% of the lost connectivity due to stroke is restored homogeneously across the regions with preexisting links toward rM1, proportionally to the initial strength of their link to rM1. We thus explore the possibility of up to 5 times of weights of the damaged links to be redistributed along the rest of the links of the nodes directly connected with the infarct area, in order to also allow for over-compensation of the lost direct connectivity. The absence of time-delays and the focus on the phase locking, makes the model insensitive on the chosen frequencies (Petkoski et al., 2016), which are therefore fixed in the simulations. The natural time-variability of parameters (Petkoski and Stefanovska, 2012) is assumed to be stochastic (Petkoski et al., 2018). We hence fix the level of the noise and we explore the impact of the global coupling K and the described strategies of the stroke and recovery. For each combination of parameters we obtain the same metric of FC as for the empirical data. The parameters space for the agreement between the modeled and the experimental data about the changes in the FC for the two parameters of the stroke-induced structural changes are shown in Figure 9C.

Figure 9D illustrates the simulated FC for spontaneously recovered “stroke” and “rehab” mice compared with pre-stroke conditions (“healthy” group), for fixed global coupling and for points in the parameters space of the stroke damage and rebound connectivity that show the best fitting with the empirical data. Comparing the simulated (Figure 9B), with the empirical FC (Figure 9D), we see that the best agreement is achieved for the FC of the somatosensory areas, while that of the visual cortex areas could be improved by testing different damage and rewiring strategies for those regions. From the model fitting for different parameters, it is also visible that generally better fit is achieved if the extent of damaged links is decreased due to rehabilitation-induced remapping. There is also a similar tendency for the rebound connectivity to be decreased due to recovery training, although there are other possible recovery paths that keep roughly the same level of rebound connectivity. In conclusion, the systematic exploration of the model parameters to best fit the empirical data, allows us to obtain the sufficient structural changes that can reproduce the modulation in FC after stroke and rehabilitation.

4.2.2. Simulation of the Calcium Activity of the Peri-Infarct Network After Stroke

Stroke profoundly alters the functionality at the local level in addition to long-range connections. The local network next to the stroke core switches to slow wave activity (Butz et al., 2004), a type of brain oscillation that is observed during deep



sleep, but also during anesthesia and other pathological brain states (Sanchez-Vives et al., 2017). Understanding the changes in the activity patterns at the level of the peri-stroke region is necessary to get insight on the possible mechanisms that underlie functional recovery. In order to explore the mechanisms that drive the neuronal networks of the peri-stroke areas to oscillate, we developed a model that reproduces the spiking activity of a local network during slow oscillations and extended the model to provide the two-photon calcium signal that one would record from that network. We compared the simulated calcium data with that of the two-photon experiments conducted in anesthetized mice (see the “Stroke models” box in Figure 2).

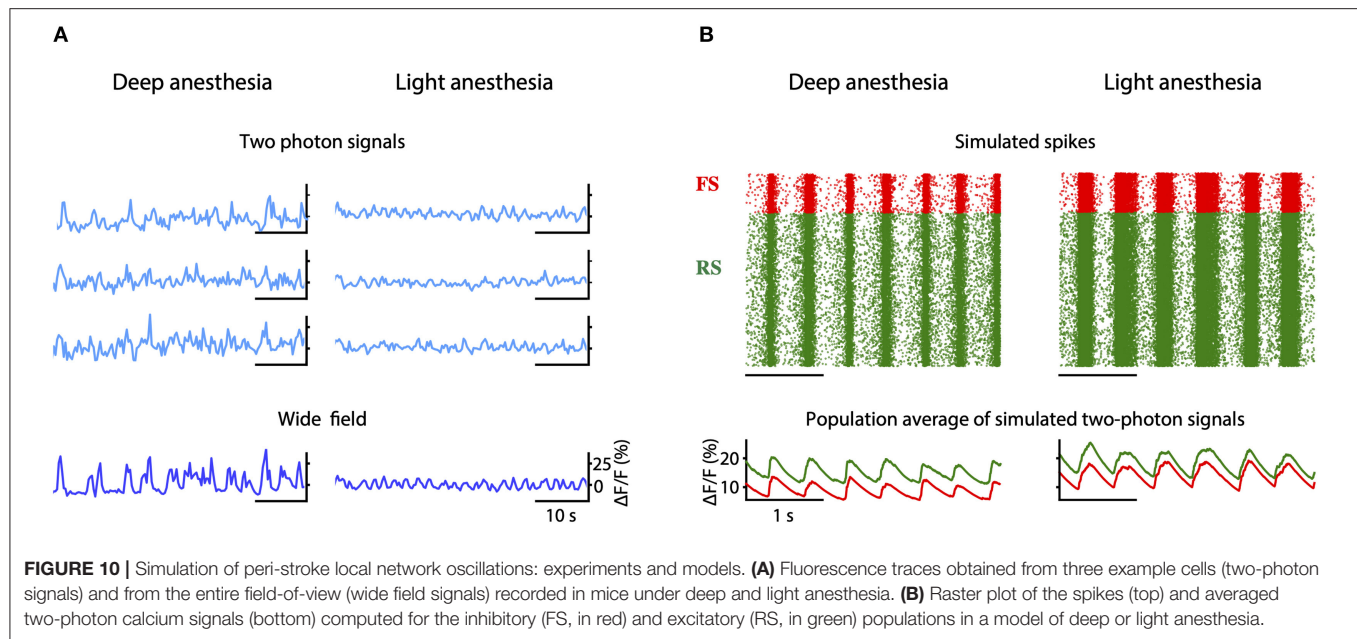
We propose that a deficit in neuromodulation produced by the decreased cerebral blood flow in the periphery of the region affected by the stroke could be responsible for the emergence of slow oscillations and the general flattening of the EEG through an increase on the level of adaptation of the neurons (Nghiem et al., 2020). To test this hypothesis, we developed a spiking network model capable of reproducing the spontaneous activity of a cortical network during different depths of anesthesia (Figure 7). The strength of adaptation in the model can be regulated to produce different types of oscillations. In particular,

we aimed at reproducing the changes in frequency of the slow oscillation observed both in the two-photon and in the wide field calcium imaging *in vivo* experiments when the anesthesia is reduced (Figure 10A). When adaptation is strong, the model of deep anesthesia produces slow oscillations at 2.1 Hz, while decreasing the strength of adaptation (model of light anesthesia) leads to slightly faster slow oscillations, at a frequency of 2.31 Hz (Figure 10B). Thus, we show that varying the strength of adaptation allows to reproduce the increase in frequency of the slow oscillations observed in the calcium imaging experiments when decreasing the level of anesthesia.

5. DISCUSSION

In this paper, we explored the steps and methods that are needed to develop a simulation model of a complete experiment. We designed, validated and combined many components, even though they are obviously not exhaustive to encounter for the complexity of the real world.

We first simulated execution of goal-directed movements with displacement of objects in the virtual reality setting. Results show that we could replicate with high accuracy the displacement of an



object by a virtual (healthy) mouse in the simulated environment (Figure 2). While this simulation is only an approximation of brain-body interactions, it showcases the capability to simulate large scale neural networks as well as body dynamics in a virtual environment.

The second pipeline of the framework involved modeling brain injury. We validated the potential of a brain network model to predict the long-range stroke-induced connectivity changes measured in a real experiment. We also tested an oscillatory spiking network model to simulate local peri-infarct activity after stroke. In addition, this model could simulate the fluctuation in calcium concentration due to spiking activity of homogeneous neuronal network, thus allowing modeling of calcium imaging data.

Toward the mechanistic understanding of behavior, few studies already provide tools for closed loop neuroscience (Mulas et al., 2010; Tessadori et al., 2012; Weidel et al., 2016, for a review see Potter et al., 2014). In addition, recent studies took advantage of virtual reality (VR) experiments conducted under controlled environment, where behavioral strategies could be isolated and tested (Dombeck and Reiser, 2012). In a VR experiment, a simulated environment is updated based on the animal's actions (Ritter et al., 2001; Chronis et al., 2007; Reiser and Dickinson, 2008; Dombeck et al., 2010). The main drawback of this approach is that the activity of animals dictates not only the response of the VR but also the properties of the neurons being measured. As a consequence, the closed-loop VR system shall then be optimized on-line based on the animal's behavior, which is very challenging. The approach we propose here instead is based on an off-line simulation, that allows exploring multiple dimensions in the parameter space of the dynamical model of mouse brain and the environment. Anyway, both strategies are synergistic with the research of effective functional brain machine interfaces (Santhanam et al., 2006).

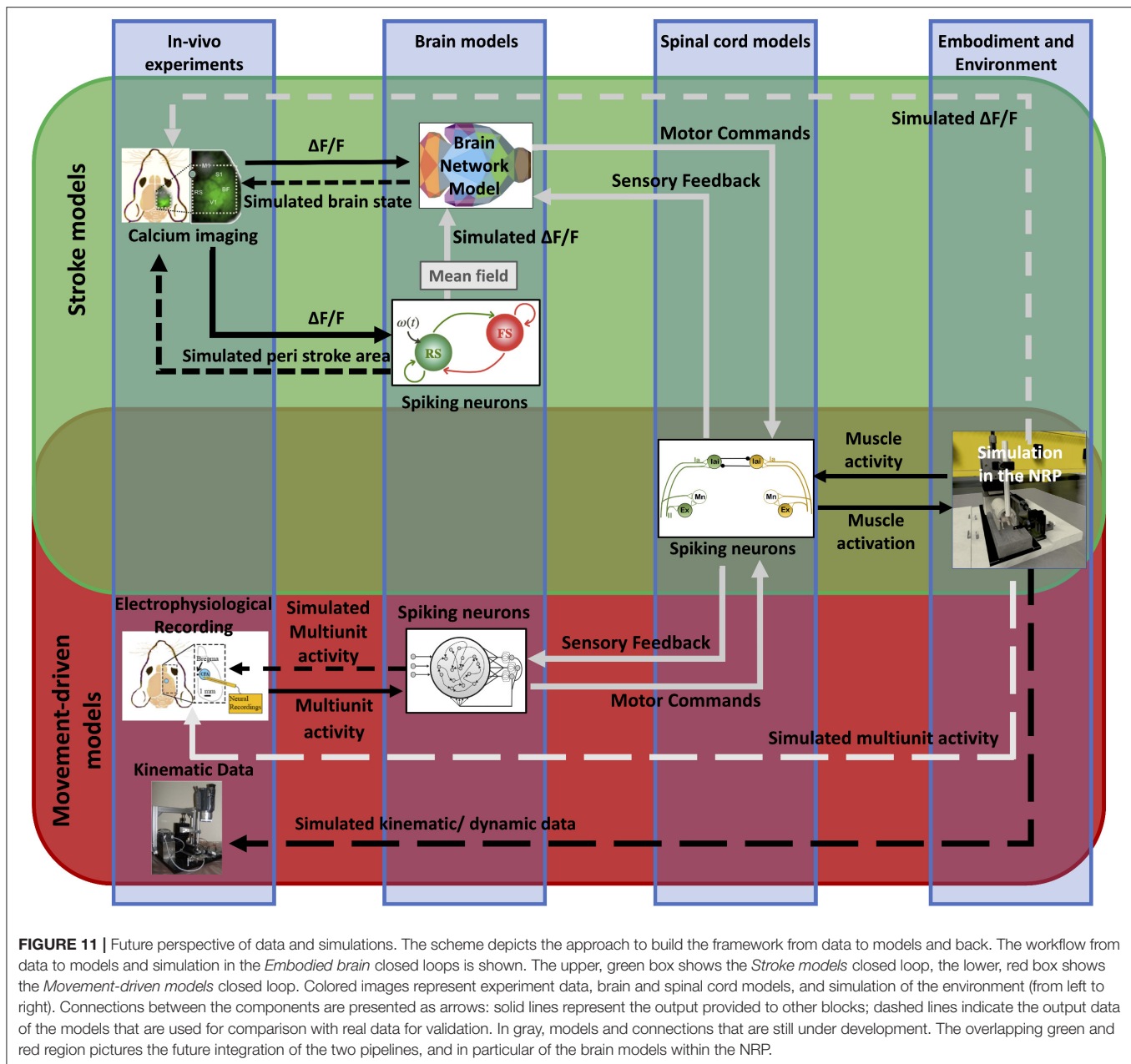
5.1. Movement-Driven Models Closed-Loop

The results showed in section 4.1 demonstrate that is possible to achieve realistic simulations by integrating some of the components described previously. Accuracy of the closed-loop simulation could be increased by removing some simplifications that are currently in place. Some of them are related to the physical models of the slide and of the musculoskeletal embodiment. Regarding the former, a more accurate slide simulation will allow to introduce friction effects that are occurring in the real setup, thus we could avoid putting a muscle activation level threshold for the release of the slide. Moreover, more detailed spinal cord and musculoskeletal models will be essential to simulate finer movements.

Results shown in Figure 8 demonstrate the system is able to simulate the pulling task, albeit with some inaccuracies on some pulling trials. A presumable cause of this inaccuracy can be identified in the low number of neurons (less than 20) that is possible to record during an experiment on the platform with the 16 channels linear probe. For this reason, it is possible that the selected units do not encompass the entire population of neurons involved in the movement. This issue could be mitigated by employing a multi-unit analysis, however, this will add to the inputs a significant background activity which may not be useful to generate the pulling movement.

Many parameters of the spinal cord circuitry can be adjusted, depending on the inputs, to accurately reproduce the movements recorded in the *in-vivo* experiments. While in this work the tuning was done manually, a more effective and generalized way would be to use different recordings, both neurophysiological and kinematic, and employ an optimization similar to what has been done in Sreenivasa et al. (2016).

The level of detail of the spinal cord circuitry can clearly be improved. In this work we modeled a minimal set of



components that were capable of replicating experimental data with a certain degree of realism. To achieve this, it was decided not to arbitrarily increase the complexity of the models by adding subcircuits whose impact cannot be clearly measured from a comparison with experimental data. Among these, it is worth mentioning the inclusion of proprioceptive feedback from Golgi tendon organs, which could be potentially implemented with computational models such as Mileusnic and Loeb (2006) or the one already included in a spinal cord model in Mugge et al. (2010). Perhaps more interesting is the modulation of muscle spindle sensitivity from γ -motoneurons, as this is crucial in the control of both voluntary and involuntary movements. While including a population of γ -motoneurons could be done by replicating populations of α -motoneurons, measuring the impact

of adding this component is not trivial, especially considered that there is no experimental data measured, in the rehabilitation setup, that can be used to validate the addition. As such, we decided not include γ -motoneurons in the spinal cord circuitry.

5.2. Stroke Models Closed-Loop

AMBA was previously tested and demonstrated to have a predictive value for the resting state dynamics in healthy conditions, compared with the gold standard individualized diffusion tensor imaging connectome (Melozzi et al., 2019). One of the main aims of the stroke modeling pipeline in this study is to validate the use of AMBA in the cases when there are significant changes in SC as compared to the healthy state for which it was obtained (Oh et al., 2014). This requires finding

the most probable structural alterations corresponding to the stroke and recovery. From the perspective of the integrative neuroscience, this is especially important as it will allow further application of these altered connectomes validated from the resting state FC, to generate the particular brain dynamics associated with active forelimb pulling on the M-platform by stroke and rehabilitated mice.

To this aim, the present study suggests that rehabilitative training could reinforce the connectivity between motor and visual areas. The iterative loop between experiments and modeling goes toward the confirmation of this hypothesis via stimulation experiments. New experiments shall verify the necessity of this feature in promoting the recovery by stimulating the connections between motor and visual cortex, and modulation of FC could be achieved via optogenetic stimulation, which recently showed to be a promising approach in stroke recovery (e.g., Cheng et al., 2014; Pendharkar et al., 2016; Conti et al., 2020).

The results from the model identify routes from the stroke to the recovery in the parameter space that can be related to neurophysiological quantities, such as the white matter tracts. We could thus determine links that need to be restored, or prevented from being established, for a successful recovery. One such a recovery path proposed by the model is the rebound in SC after rehabilitative training, and this is especially true for the links involving the visual-associated areas. The proposed rebound is due to newly established links from the regions afferent to the site of the stroke. This can lead to overall overcompensation for the SC, and some of these scenarios could be possible paths for recovery. However, it remains to be seen whether the structural changes of such magnitude can be achieved. Several studies previously showed that axonal growth is stimulated by neurorehabilitative activities after stroke, and that sprouting can extend to widespread brain systems (Carmichael et al., 2017). New experiments aimed at verifying the SC modifications shall verify the hypothesis on the importance of modified connectivity in the visual areas for recovery. In addition, stimulation experiments can also strengthen certain links, and with our modeling framework we can virtually compare the effects of each such modification to the observed dynamics.

For the best fitting of the data one would also need to allow different levels of the global coupling, which governs the global level of synchronization and that is already shown to be increased during stroke (Falcon et al., 2016; Corbetta et al., 2018), thus decreasing integration and information capacity (Adhikari et al., 2017) and modularity (Falcon et al., 2015). Thus, one could more precisely identify the path from stroke to recovery for a wider parameters range. This also includes numerically testing different scenarios for heterogeneous connectivity reinforcing (Nudo, 2013) such as reinforcing of contralateral links in general, or those to contralateral stroke region only, or toward the nodes (ipsi-, contra-lateral, or both) that were connected to the damaged region prior stroke.

Possible problems could arise from the alignment of the experimental data, especially after the stroke, due to the shrinkage and the movement of the tissue (Brown et al., 2007; Allegra Mascaro et al., 2019). We have tried to avoid this by

excluding from the analysis the regions adjacent to the stroke, but this reduces the predictive value of the model due to smaller number of analyzed regions.

Finally, these experiments provide a picture of the ipsilesional functionality after stroke and rehabilitation, but many other regions are involved, including the contralesional hemisphere (see, for instance, Dodd et al., 2017). In the next experiments, the focus shall be on recording with a higher sampling rate to capture wider spectrum of brain dynamics, and on enlarging the field-of-view of the wide-field imaging setup to provide longitudinal pictures of cortical functionality over both hemispheres. The latter should also refine the fitting across parameters, which now contains large areas or similar level of predictability, thus offering more precise recovery path. Individualized connectome data by Diffusion Tensor Imaging during the recovery process is another aspect of the future experiments that should test the predicted changes in the structure that we propose to be the cause of the observed functional alterations of different conditions. In addition, higher resolution SC performed with light-sheet microscopy on individual mice (Allegra Mascaro et al., 2015) could test the model prediction at the final time point of the experiment. As a final step, an individualized therapy could be proposed targeting specific parts of the brain (Spalletti et al., 2017; Allegra Mascaro et al., 2019; Conti et al., 2020), depending on the location and the size of the stroke.

5.3. Integration

We propose viable strategies to integrate the brain models described here and to embed them within the *Embodied brain* framework on the NRP (Falotico et al., 2017) (pictured by the overlapping green and red boxes in **Figure 11**). Before applying it to the simulation of the whole-brain dynamics, the spiking neurons model shall be extended to include the heterogeneous long-range connections either via mean-field approximation or by means of co-simulation with other neural masses (see the spiking neurons model that receives calcium imaging data in **Figure 11**). In addition, a model for embedding spiking model modules into the whole brain model is currently under development (displayed as a gray arrow from the spiking neurons to BNM in **Figure 11**). This work includes validating neuronal mass models against high-dimensional neuronal networks. Once available, this tool will allow bridging the scales of brain models with different levels of description, and they will be then implemented in the NRP and integrated into the *Embodied brain* framework (gray arrows in upper box of **Figure 11**).

To integrate the large-scale BNM with the proposed spinal cord model, we propose to modulate the activity of the spiking neurons in the spinal cord by the output of the cortical regions, mainly those related to the motor activity (displayed by gray arrows in the *Stroke models* closed loop, green upper box in **Figure 11**). In particular, the firing rate of the neurons in the spinal cord that triggers the movements on the NRP can be driven by the mean activity of the cortical motor regions, or by some specific patterns of their co-activation, such as a high-level activity propagation, similar as the one observed during the movements. In this way, the mean

neuronal activity of the brain regions at different conditions would trigger movements at the NRP using the activity of the spinal cord. For the feedback link of the sensory activity, we envision the information about muscle activity and limb displacement, which is encoded into the firing patterns of the spinal cord spiking neurons, to directly modulate the mean activity of the sensory motor regions (displayed as a dashed gray arrow in the *Stroke models* closed loop, upper box in **Figure 11**). This on the other hand would impact the overall brain network dynamics, including the activation patterns of the motor regions.

To allow the flow of information from the brain to the virtual environment, we anticipate that the next step will be the integration of a spiking network model of motor areas upstream to the spinal cord model. This data-driven model of the motor cortex will include populations of pyramidal neurons and interneurons that can be functionally attached to different lower circuits (displayed by gray arrows in the *Movement-driven* closed loop, red lower box in **Figure 11**). This integration in the proposed framework can be an effective strategy to effectively close the *Embodied brain* loop.

6. CONCLUSIONS

To summarize, in this study we proposed a methodological framework (named *Embodied brain*) to investigate a “brain in the loop” by a constructive refinement of experiments and simulation of an embodied mouse.

Our findings suggest that simulation of real experiments within the proposed framework will help better understand the complex mechanism that underlies the generation of behavior. Nevertheless, the actual advantages of the “Embodiment” approach, still under construction, are largely unexplored. Even though some aspects of complex animal behavior may be represented with good accuracy by modeling single neural components, without embedding the neural simulations in a physical embodiment it is impossible to show the effect of such neural systems on the body and the surrounding environment. In our study, it would be impossible to assess whether or not the neural models are capable of performing the pulling task with any degree of accuracy, computed on the kinematic data. Furthermore, we believe that new features [e.g., activation of different brain regions for performing the same task due to degeneracy (Price and Friston, 2002) and its impact for stroke and recovery] will be disclosed by the simulation of the entire experiment. In conclusion, the framework shown in this study will advance the field by formulating new hypothesis on the mechanism underlying goal-directed voluntary movements, to be validated on *ad hoc* designed experiments. In general, the framework could simulate new types of experiments that cannot be run in the real world. Last but not least, the virtual environment will be an essential tool to reduce the number of animals used in the experiments, thus making the “Reduction” rule on animal experimentation a feasible goal.

DATA AVAILABILITY STATEMENT

The datasets generated for this study are openly available online at: <http://doi.org/10.25493/Z9J0-ZZQ>. The raw datasets on neurophysiological recordings for this study can be found online at: <http://doi.org/10.5281/zenodo.3546068>. Processed data and source code that can be used to reproduce the neurophysiological recordings experiment can be found in the following repository: <https://gitlab.com/lore.ucci/closed-loop-mouse-stroke-simulation>. The BNM model and all the relevant information to reproduce the simulated BNM data can be found in the following repository: https://github.com/esaps/AllenMouse_strokeKuramoto. Data sharing license: This work is shared under a Creative Commons Attribution CC BY 4.0 license (<https://creativecommons.org>).

ETHICS STATEMENT

All the procedures were in accordance with the Italian Ministry of Health for care and maintenance of laboratory animals (law 116/92) and in compliance with the European Communities Council Directive no. 2010/63/EU, under authorizations no. 183/2016-PR (imaging experiments), and no. 753/2015-PR (electrophysiology experiments).

AUTHOR CONTRIBUTIONS

ALAM, EFa, SP, LV, M-OG, and FP conceived the study. ALAM, MP, EC, FR, and CS performed the experiments. ALAM, EFa, SP, LV, MP, NT-C, EC, SR, EA, CB, and TL wrote the paper. All authors agreed with the manuscript. SP, VJ, NT-C, and AD developed the brain models. LV and EFa developed the spinal cord model. LV, AA, EA, and SR developed the simulation in the NRP.

FUNDING

This project was supported by the European Union's Horizon 2020 research and innovation programme under grant agreement nos. 720270 (SGA1), 785907 (SGA2), and 945539 (SGA3) Human Brain Project.

ACKNOWLEDGMENTS

We thank Krister Andersson, Oliver Schmid, Martin Øvsthus, Ingrid Reiten, and Jan G. Bjaalie of the Human Brain Project curation team for expert assistance to share data via the Human Brain Project Neuroinformatics Platform.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fnsys.2020.00031/full#supplementary-material>

REFERENCES

- Adhikari, M. H., Hacker, C. D., Siegel, J. S., Griffo, A., Hagmann, P., Deco, G., et al. (2017). Decreased integration and information capacity in stroke measured by whole brain models of resting state activity. *Brain* 140, 1068–1085. doi: 10.1093/brain/awx021
- Allegra Mascaro, A. L., Conti, E., Lai, S., Di Giovanna, A. P., Spalletti, C., Alia, C., et al. (2019). Combined rehabilitation promotes the recovery of structural and functional features of healthy neuronal networks after stroke. *Cell Rep.* 28, 3474–3485.e6. doi: 10.1016/j.celrep.2019.08.062
- Allegra Mascaro, A. L., Silvestri, L., Sacconi, L., and Pavone, F. S. (2015). Towards a comprehensive understanding of brain machinery by correlative microscopy. *J. Biomed. Optics* 20:061105. doi: 10.1117/1.JBO.20.6.061105
- Alstermark, B. (1992). “The C3-C4 propriospinal system: target-reaching and food-taking,” in *Muscle Afferents and Spinal Control of Movement* (Elsevier Ltd), 327–354.
- Bashor, D. P. (1998). A large-scale model of some spinal reflex circuits. *Biol. Cybernet.* 78, 147–157. doi: 10.1007/s004220050421
- Batterman, R. W., and Rice, C. C. (2014). Minimal model explanations. *Philos. Sci.* 81, 349–376. doi: 10.1086/676667
- Breakspear, M., Heitmann, S., and Daffertshofer, A. (2010). Generative models of cortical oscillations: neurobiological implications of the Kuramoto model. *Front. Hum. Neurosci.* 4:190. doi: 10.3389/fnhum.2010.00190
- Brette, R., and Gerstner, W. (2005). Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *J. Neurophysiol.* 94, 3637–3642. doi: 10.1152/jn.00686.2005
- Brown, C. E., Li, P., Boyd, J. D., Delaney, K. R., and Murphy, T. H. (2007). Extensive turnover of dendritic spines and vascular remodeling in cortical tissues recovering from stroke. *J. Neurosci.* 27, 4101–4109. doi: 10.1523/JNEUROSCI.4295-06.2007
- Butz, M., Gross, J., Timmermann, L., Moll, M., Freund, H.-J., Witte, O. W., et al. (2004). Perilesional pathological oscillatory activity in the magnetoencephalogram of patients with cortical brain lesions. *Neurosci. Lett.* 355, 93–96. doi: 10.1016/j.neulet.2003.10.065
- Cabral, J., Hugues, E., Kringelbach, M. L., and Deco, G. (2012). Modeling the outcome of structural disconnection on resting-state functional connectivity. *Neuroimage* 62, 1342–1353. doi: 10.1016/j.neuroimage.2012.06.007
- Cabral, J., Hugues, E., Sporns, O., and Deco, G. (2011). Role of local network oscillations in resting-state functional connectivity. *Neuroimage* 57, 130–139. doi: 10.1016/j.neuroimage.2011.04.010
- Cabral, J., Vidaurre, D., Marques, P., Magalhães, R., Silva Moreira, P., Miguel Soares, J., et al. (2017). Cognitive performance in healthy older adults relates to spontaneous switching between states of functional connectivity during rest. *Sci. Rep.* 7, 1–13. doi: 10.1038/s41598-017-05425-7
- Carmichael, S. T., and Chesselet, M.-F. (2002). Synchronous neuronal activity is a signal for axonal sprouting after cortical lesions in the adult. *J. Neurosci.* 22, 6062–6070. doi: 10.1523/JNEUROSCI.22-14-06062.2002
- Carmichael, S. T., Kathirvelu, B., Schweppe, C. A., and Nie, E. H. (2017). Molecular, cellular and functional events in axonal sprouting after stroke. *Exp. Neurol.* 287, 384–394. doi: 10.1016/j.expneurol.2016.02.007
- Chen, T. W., Wardill, T. J., Sun, Y., Pulver, S. R., Renninger, S. L., Baohan, A., et al. (2013). Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature* 499, 295–300. doi: 10.1038/nature12354
- Cheng, M. Y., Wang, E. H., Woodson, W. J., Wang, S., Sun, G., Lee, A. G., et al. (2014). Optogenetic neuronal stimulation promotes functional recovery after stroke. *Proc. Natl. Acad. Sci. U.S.A.* 111, 12913–12918. doi: 10.1073/pnas.1404109111
- Choi, H., and Mihalas, S. (2019). Synchronization dependent on spatial structures of a mesoscopic whole-brain network. *PLoS Comput. Biol.* 15:e1006978. doi: 10.1371/journal.pcbi.1006978
- Chronis, N., Zimmer, M., and Bargmann, C. I. (2007). Microfluidics for *in vivo* imaging of neuronal and behavioral activity in *Caenorhabditis elegans*. *Nat. Methods* 4:727. doi: 10.1038/nmeth1075
- Cisi, R. R., and Kohn, A. F. (2008). Simulation system of spinal cord motor nuclei and associated nerves and muscles, in a web-based architecture. *J. Comput. Neurosci.* 25, 520–542. doi: 10.1007/s10827-008-0092-8
- Conti, E., Allegra Mascaro, A. L., and Pavone, F. S. (2019). Large scale double-path illumination system with split field of view for the all-optical study of inter- and intra-hemispheric functional connectivity on mice. *Methods Protoc.* 2:11. doi: 10.3390/mps2010011
- Conti, E., Allegra Mascaro, A. L., Scaglione, A., de Vito, G., Calugi, F., Pasquini, M., et al. (2020). Synergic effect of optogenetic stimulation and motor training boosts recovery of motor functionality after stroke supported by segregation of motor representation. *bioRxiv*. doi: 10.1101/2020.03.05.974972
- Corbetta, M., Siegel, J. S., and Shulman, G. L. (2018). On the low dimensionality of behavioral deficits and alterations of brain network connectivity after focal injury. *Cortex* 107, 1–9. doi: 10.1016/j.cortex.2017.12.017
- Dana, H., Chen, T.-W., Hu, A., Shields, B. C., Guo, C., Looger, L. L., et al. (2014). Thy1-GCaMP6 transgenic mice for neuronal population imaging *in vivo*. *PLoS ONE* 9:108697. doi: 10.1371/journal.pone.0108697
- Dancuse, N. (2005). Extensive cortical rewiring after brain injury. *J. Neurosci.* 25, 10167–10179. doi: 10.1523/JNEUROSCI.3256-05.2005
- Davison, A. P., Brüderle, D., Eppler, J., Kremkow, J., Müller, E., Pecevski, D., et al. (2009). PyNN: a common interface for neuronal network simulators. *Front. Neuroinform.* 2:11. doi: 10.3389/neuro.11.011.2008
- Deco, G., and Jirsa, V. K. (2012). Ongoing cortical activity at rest: criticality, multistability, and ghost attractors. *J. Neurosci.* 32, 3366–3375. doi: 10.1523/JNEUROSCI.2523-11.2012
- Deco, G., Jirsa, V. K., and McIntosh, A. R. (2011). Emerging concepts for the dynamical organization of resting-state activity in the brain. *Nat. Rev. Neurosci.* 12, 43–56. doi: 10.1038/nrn2961
- Deco, G., Jirsa, V. K., McIntosh, A. R., Sporns, O., and Kötter, R. (2009). Key role of coupling, delay, and noise in resting brain fluctuations. *Proc. Natl. Acad. Sci. U.S.A.* 106, 10302–10307. doi: 10.1073/pnas.0901831106
- Deco, G., Jirsa, V. K., Robinson, P. A., Breakspear, M., and Friston, K. (2008). The dynamic brain: from spiking neurons to neural masses and cortical fields. *PLoS Comput. Biol.* 4:e1000092. doi: 10.1371/journal.pcbi.1000092
- Deco, G., Kringelbach, M. L., Jirsa, V., and Ritter, P. (2016). The dynamics of resting fluctuations in the brain: metastability and its dynamical cortical core. *Sci. Rep.* 7:3095. doi: 10.1101/065284
- DeMarse, T. B., Wagenaar, D. A., Blau, A. W., and Potter, S. M. (2001). The neurally controlled animat: biological brains acting with simulated bodies. *Auton. Robots* 11, 305–310. doi: 10.1023/A:1012407611130
- Destexhe, A. (2009). Self-sustained asynchronous irregular states and up-down states in thalamic, cortical and thalamocortical networks of nonlinear integrate-and-fire neurons. *J. Comput. Neurosci.* 27:493. doi: 10.1007/s10827-009-0164-4
- Dodd, K. C., Nair, V. A., and Prabhakaran, V. (2017). Role of the contralesional vs. ipsilesional hemisphere in stroke recovery. *Front. Hum. Neurosci.* 11:469. doi: 10.3389/fnhum.2017.00469
- Dombeck, D. A., Harvey, C. D., Tian, L., Looger, L. L., and Tank, D. W. (2010). Functional imaging of hippocampal place cells at cellular resolution during virtual navigation. *Nat. Neurosci.* 13:1433. doi: 10.1038/nn.2648
- Dombeck, D. A., and Reiser, M. B. (2012). Real neuroscience in virtual worlds. *Curr. Opin. Neurobiol.* 22, 3–10. doi: 10.1016/j.conb.2011.10.015
- Ermmentrout, B. (1998). Linearization of F-I curves by adaptation. *Neural Comput.* 10, 1721–1729. doi: 10.1162/089976698300017106
- Falcon, M. I., Riley, J. D., Jirsa, V., McIntosh, A. R., Chen, E. E., and Solodkin, A. (2016). Functional mechanisms of recovery after chronic stroke: modeling with the virtual brain. *eNeuro* 3, 202–208. doi: 10.1523/ENEURO.0158-15.2016
- Falcon, M. I., Riley, J. D., Jirsa, V., McIntosh, A. R., Shereen, A. D., Elinor Chen, E., et al. (2015). The virtual brain: Modeling biological correlates of recovery after chronic stroke. *Front. Neurol.* 6:228. doi: 10.3389/fneur.2015.00228
- Falotico, E., Vannucci, L., Ambrosano, A., Albanese, U., Ulbrich, S., Vasquez Tieck, J. C., et al. (2017). Connecting artificial brains to robots in a comprehensive simulation framework: the neurobotics platform. *Front. Neurobot.* 11:2. doi: 10.3389/fnbot.2017.00002
- Franklin, K. B., and Paxinos, G. (2008). *The Mouse Brain in Stereotaxic Coordinates*, Vol. 3. New York, NY: Academic Press.
- Fuglevand, A. J., Winter, D. A., and Patla, A. E. (1993). Models of recruitment and rate coding organization in motor-unit pools. *J. Neurophysiol.* 70, 2470–2488. doi: 10.1152/jn.1993.70.6.2470
- Gewaltig, M., and Diesmann, M. (2007). NEST (NEural Simulation Tool). *Scholarpedia* 2:1430. doi: 10.4249/scholarpedia.1430
- Ghosh, A., Rho, Y., McIntosh, A. R., Kötter, R., and Jirsa, V. K. (2008). Noise during rest enables the exploration of the brain’s dynamic repertoire. *PLoS Comput. Biol.* 4:e1000196. doi: 10.1371/journal.pcbi.1000196
- Grienberger, C., and Konnerth, A. (2012). Imaging calcium in neurons. *Neuron* 73, 862–885. doi: 10.1016/j.neuron.2012.02.011

- HBPNeuroRobotics (2019). *Robot Designer*. Available online at: <https://github.com/HBPNeuroRobotics/BlenderRobotDesigner>
- Helton, T. D., Xu, W., and Lipscombe, D. (2005). Neuronal I-type calcium channels open quickly and are inhibited slowly. *J. Neurosci.* 25, 10247–10251. doi: 10.1523/JNEUROSCI.1089-05.2005
- Hinkel, G., Groenda, H., Krach, S., Vannucci, L., Denninger, O., Cauli, N., et al. (2017). A framework for coupled simulations of robots and spiking neuronal networks. *J. Intell. Robot. Syst.* 85, 71–91. doi: 10.1007/s10846-016-0412-6
- Hinkel, G., Groenda, H., Vannucci, L., Denninger, O., Cauli, N., and Ulbrich, S. (2015). “A domain-specific language (DSL) for integrating neuronal networks in robot control,” in *Proceedings of the 2015 Joint MORSE/VAO Workshop on Model-Driven Robot Software Engineering and View-based Software-Engineering*. MORSE/VAO '15 (New York, NY: ACM), 9–15. doi: 10.1145/2802059.2802060
- Hodgkin, A. L., and Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.* 117, 500–544. doi: 10.1113/jphysiol.1952.sp004764
- Honey, C. J., Kötter, R., Breakspear, M., and Sporns, O. (2007). Network structure of cerebral cortex shapes functional connectivity on multiple time scales. *Proc. Natl. Acad. Sci. U.S.A.* 104, 10240–10245. doi: 10.1073/pnas.0701519104
- Izhikevich, E. M. (1998). Phase models with explicit time delays. *Phys. Rev. E* 58, 905–908. doi: 10.1103/PhysRevE.58.905
- Izhikevich, E. M., and Edelman, G. M. (2008). Large-scale model of mammalian thalamocortical systems. *Proc. Natl. Acad. Sci. U.S.A.* 105, 3593–3598. doi: 10.1073/pnas.0712231105
- Jirsa, V. K., Stacey, W. C., Quilichini, P. P., Ivanov, A. I., and Bernard, C. (2014). On the nature of seizure dynamics. *Brain* 137, 2210–2230. doi: 10.1093/brain/awu133
- Johansen-Berg, H., and Rushworth, M. F. S. (2009). Using diffusion imaging to study human connective anatomy. *Annu. Rev. Neurosci.* 32, 75–94. doi: 10.1146/annurev.neuro.051508.135735
- Koenig, N., and Howard, A. (2004). “Design and use paradigms for gazebo, an open-source multi-robot simulator,” in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE Cat. No.04CH37566), Vol. 3 (Sendai), 2149–2154. doi: 10.1109/IROS.2004.1389727
- Kuramoto, Y. (1984). *Chemical Oscillations, Waves, and Turbulence*. Berlin: Springer-Verlag. doi: 10.1007/978-3-642-69689-3
- Lachaux, J. P., Rodriguez, E., Martinerie, J., and Varela, F. J. (1999). Measuring phase synchrony in brain signals. *Hum. Brain Mapp.* 8, 194–208. doi: 10.1002/(SICI)1097-0193(1999)8:4<194::AID-HBM4>3.0.CO;2-C
- Lancaster, G., Iatsenko, D., Pidde, A., Ticcinelli, V., and Stefanovska, A. (2018). Surrogate data for hypothesis testing of dynamical systems. *Phys. Rep.* 748, 1–60. doi: 10.1016/j.physrep.2018.06.001
- Markram, H., Müller, E., Ramaswamy, S., Reimann, M. W., Abdellah, M., Sanchez, C. A., et al. (2015). Reconstruction and simulation of neocortical microcircuitry. *Cell* 163, 456–492. doi: 10.1016/j.cell.2015.09.029
- Melozzi, F., Bergmann, E., Harris, J. A., Kahn, I., Jirsa, V., and Bernard, C. (2019). Individual structural features constrain the mouse functional connectome. *Proc. Natl. Acad. Sci. USA* 116, 26961–26969. doi: 10.1073/pnas.1906694116
- Melozzi, F., Woodman, M. M., Jirsa, V. K., and Bernard, C. (2017). The virtual mouse brain: a computational neuroinformatics platform to study whole mouse brain dynamics. *eNeuro* 4, 1–14. doi: 10.1523/ENEURO.0111-17.2017
- Mileusnic, M. P., Brown, I. E., Lan, N., and Loeb, G. E. (2006). Mathematical models of proprioceptors. I. control and transduction in the muscle spindle. *J. Neurophysiol.* 96, 1772–1788. doi: 10.1152/jn.00868.2005
- Mileusnic, M. P., and Loeb, G. E. (2006). Mathematical models of proprioceptors. II. structure and function of the golgi tendon organ. *J. Neurophysiol.* 96, 1789–1802. doi: 10.1152/jn.00869.2005
- Millard, M., Uchida, T., Seth, A., and Delp, S. L. (2013). Flexing computational muscle: modeling and simulation of musculotendon dynamics. *J. Biomech. Eng.* 135, 021005–021005. doi: 10.1115/1.4023390
- Moraud, E. M., Capogrosso, M., Formento, E., Wenger, N., DiGiovanna, J., Courtine, G., et al. (2016). Mechanisms underlying the neuromodulation of spinal circuits for correcting gait and balance deficits after spinal cord injury. *Neuron* 89, 814–828. doi: 10.1016/j.neuron.2016.01.009
- Mugge, W., Abbink, D. A., Schouten, A. C., Dewald, J. P., and Van Der Helm, F. C. (2010). A rigorous model of reflex function indicates that position and force feedback are flexibly tuned to position and force tasks. *Exp. Brain Res.* 200, 325–340. doi: 10.1007/s00221-009-1985-0
- Mulas, M., Massobrio, P., Martinoia, S., and Chiappalone, M. (2010). A simulated neuro-robotic environment for bi-directional closed-loop experiments. *Paladyn* 1, 179–186. doi: 10.2478/s13230-011-0004-x
- Nakagawa, T. T., Jirsa, V. K., Spiegler, A., McIntosh, A. R., and Deco, G. (2013). Bottom up modeling of the connectome: linking structure and function in the resting brain and their changes in aging. *Neuroimage* 80, 318–329. doi: 10.1016/j.neuroimage.2013.04.055
- Nghiem, T.-A. E., Tort-Colet, N., Górski, T., Ferrari, U., Moghimyfiroozabad, S., Goldman, J. S., et al. (2020). Cholinergic switch between two types of slow waves in cerebral cortex. *Cereb. Cortex* 30, 3451–3466. doi: 10.1093/cercor/bhz320
- Nudo, R. J. (2013). Recovery after brain injury: mechanisms and principles. *Front. Hum. Neurosci.* 7:887. doi: 10.3389/fnhum.2013.00887
- Oh, S. W., Harris, J. A., Ng, L., Winslow, B., Cain, N., Mihalas, S., et al. (2014). A mesoscale connectome of the mouse brain. *Nature* 508, 207–214. doi: 10.1038/nature13186
- Pasquini, M., Lai, S., Spalletti, C., Cracchiolo, M., Conti, S., Panarese, A., et al. (2018). A robotic system for adaptive training and function assessment of forelimb retraction in mice. *IEEE Trans. Neural Syst. Rehabil. Eng.* 26, 1803–1812. doi: 10.1109/TNSRE.2018.2864279
- Pendharkar, A. V., Levy, S. L., Ho, A. L., Sussman, E. S., Cheng, M. Y., and Steinberg, G. K. (2016). Optogenetic modulation in stroke recovery. *Neurosurg. Focus* 40:E6. doi: 10.3171/2016.2.FOCUS163
- Petkoski, S., and Jirsa, V. K. (2019). Transmission time delays organize the brain network synchronization. *Philos. Trans. R. Soc. A* 377:20180132. doi: 10.1098/rsta.2018.0132
- Petkoski, S., Palva, J. M., and Jirsa, V. K. (2018). Phase-lags in large scale brain synchronization: methodological considerations and *in-silico* analysis. *PLoS Comput. Biol.* 14:e1006160. doi: 10.1371/journal.pcbi.1006160
- Petkoski, S., Spiegler, A., Proix, T., Aram, P., Temprado, J.-J., and Jirsa, V. K. (2016). Heterogeneity of time delays determines synchronization of coupled oscillators. *Phys. Rev. E* 94:012209. doi: 10.1103/PhysRevE.94.012209
- Petkoski, S., and Stefanovska, A. (2012). Kuramoto model with time-varying parameters. *Phys. Rev. E* 86:046212. doi: 10.1103/PhysRevE.86.046212
- Pierrot-Deseilligny, E., and Burke, D. (2005). *The Circuitry of the Human Spinal Cord: Its Role in Motor Control and Movement Disorders*. Cambridge: Cambridge University Press. doi: 10.1017/CBO9780511545047
- Pikovsky, A., Rosenblum, M., and Kurths, J. (2001). *Synchronization-A Universal Concept in Nonlinear Sciences*. Cambridge: Cambridge University Press. doi: 10.1017/CBO9780511755743
- Ponce-Alvarez, A., Deco, G., Hagmann, P., Romani, G. L., Mantini, D., and Corbetta, M. (2015). Resting-state temporal synchronization networks emerge from connectivity topology and heterogeneity. *PLoS Comput. Biol.* 11:e1004100. doi: 10.1371/journal.pcbi.1004100
- Potter, S. M., El Hady, A., and Fetzi, E. E. (2014). Closed-loop neuroscience and neuroengineering. *Front. Neural Circ.* 8:115. doi: 10.3389/fncir.2014.00115
- Price, C. J., and Friston, K. J. (2002). Degeneracy and cognitive anatomy. *Trends Cogn. Sci.* 6, 416–421. doi: 10.1016/S1364-6613(02)01976-9
- Puchades, M. A., Csucs, G., Ledergerber, D., Leergaard, T. B., and Bjaalie, J. G. (2019). Spatial registration of serial microscopic brain images to three-dimensional reference atlases with the QUICKNII tool. *PLoS ONE* 14:e0216796. doi: 10.1371/journal.pone.0216796
- Rabiller, G., He, J.-W., Nishijima, Y., Wong, A., and Liu, J. (2015). Perturbation of brain oscillations after ischemic stroke: a potential biomarker for post-stroke function and therapy. *Int. J. Mol. Sci.* 16, 25605–25640. doi: 10.3390/ijms161025605
- Rahmati, V., Kirmse, K., Marković, D., Holthoff, K., and Kiebel, S. J. (2016). Inferring neuronal dynamics from calcium imaging data using biophysical models and Bayesian inference. *PLoS Comput. Biol.* 12:e1004736. doi: 10.1371/journal.pcbi.1004736
- Ramanathan, D. S., Guo, L., Gulati, T., Davidson, G., Hishinuma, A. K., Won, S.-J., et al. (2018). Low-frequency cortical activity is a neuromodulatory target that tracks recovery after stroke. *Nat. Med.* 24:1257. doi: 10.1038/s41591-018-0058-y
- Reger, B. D., Fleming, K. M., Sanguineti, V., Alford, S., and Mussa-Ivaldi, F. A. (2000). Connecting brains to robots: an artificial body for studying the computational properties of neural tissues. *Artif. Life* 6, 307–324. doi: 10.1162/106454600300103656
- Reiser, M. B., and Dickinson, M. H. (2008). A modular display system for insect behavioral neuroscience. *J. Neurosci. Methods* 167, 127–139. doi: 10.1016/j.jneumeth.2007.07.019

- Rijdsdijk, M., Leijten, F., and Slooter, A. (2008). Continuous EEG monitoring in the intensive care unit. *Neth. J. Crit Care* 12, 157–62.
- Ritter, D. A., Bhatt, D. H., and Fetcho, J. R. (2001). *In vivo* imaging of zebrafish reveals differences in the spinal networks for escape and swimming movements. *J. Neurosci.* 21, 8956–8965. doi: 10.1523/JNEUROSCI.21-22-08956.2001
- Roy, D., Ghosh, A., and Jirsa, V. K. (2011). Phase description of spiking neuron networks with global electric and synaptic coupling. *Phys. Rev. E* 83:051909. doi: 10.1103/PhysRevE.83.051909
- Saenger, V. M., Ponce-Alvarez, A., Adhikari, M., Hagmann, P., Deco, G., and Corbetta, M. (2018). Linking entropy at rest with the underlying structural connectivity in the healthy and lesioned brain. *Cereb. Cortex* 28, 2948–2958. doi: 10.1093/cercor/bhx176
- Saggio, M. L., Spiegler, A., Bernard, C., and Jirsa, V. K. (2017). Fast-slow bursters in the unfolding of a high codimension singularity and the ultra-slow transitions of classes. *J. Math. Neurosci.* 7:7. doi: 10.1186/s13408-017-0050-8
- Sanchez-Vives, M. V., Massimini, M., and Mattia, M. (2017). Shaping the default activity pattern of the cortical network. *Neuron* 94, 993–1001. doi: 10.1016/j.neuron.2017.05.015
- Santhanam, G., Ryu, S. I., Byron, M. Y., Afshar, A., and Shenoy, K. V. (2006). A high-performance brain-computer interface. *Nature* 442:195. doi: 10.1038/nature04968
- Sanz Leon, P., Knock, S. A., Woodman, M. M., Domide, L., Mersmann, J., McIntosh, A. R., et al. (2013). The Virtual Brain: a simulator of primate brain network dynamics. *Front. Neuroinform.* 7:10. doi: 10.3389/fninf.2013.00010
- Sanz-Leon, P., Knock, S. A., Spiegler, A., and Jirsa, V. K. (2015). Mathematical framework for large-scale brain network modeling in the virtual brain. *Neuroimage* 111, 385–430. doi: 10.1016/j.neuroimage.2015.01.002
- Schmidt, H., Petkov, G., Richardson, M. P., and Terry, J. R. (2014). Dynamics on networks: the role of local dynamics and global networks on the emergence of hypersynchronous neural activity. *PLoS Comput. Biol.* 10:e1003947. doi: 10.1371/journal.pcbi.1003947
- Sheppard, L. W., Hale, A. C., Petkoski, S., McClintock, P. V. E., and Stefanovska, A. (2013). Characterizing an ensemble of interacting oscillators: the mean-field variability index. *Phys. Rev. E* 87:012905. doi: 10.1103/PhysRevE.87.012905
- Spalletti, C., Alia, C., Lai, S., Panarese, A., Conti, S., Micera, S., et al. (2017). Combining robotic training and inactivation of the healthy hemisphere restores pre-stroke motor patterns in mice. *eLife* 6, 1–31. doi: 10.7554/eLife.28662
- Spalletti, C., Lai, S., Mainardi, M., Panarese, A., Ghionzoli, A., Alia, C., et al. (2014). A robotic system for quantitative assessment and poststroke training of forelimb retraction in mice. *Neurorehabil. Neural Repair* 28, 188–196. doi: 10.1177/1545968313506520
- Spiegler, A., Hansen, E. C. A., Bernard, C., McIntosh, A. R., and Jirsa, V. K. (2016). Selective activation of resting-state networks following focal stimulation in a connectome-based network model of the human brain. *eNeuro* 3, 1–17. doi: 10.1523/ENEURO.0068-16.2016
- Sreenivasa, M., Ayusawa, K., and Nakamura, Y. (2016). Modeling and identification of a realistic spiking neural network and musculoskeletal model of the human arm, and an application to the stretch reflex. *IEEE Trans. Neural Syst. Rehabil. Eng.* 24, 591–602. doi: 10.1109/TNSRE.2015.2478858
- Stankovski, T., Petkoski, S., Smith, A. F., McClintock, P. V. E., Stefanovska, A., Raeder, J., et al. (2016). Alterations in the coupling functions between cortical and cardio-respiratory oscillations due to anaesthesia with propofol and sevoflurane. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* 374:20150186. doi: 10.1098/rsta.2015.0186
- Stefanescu, R. A., and Jirsa, V. K. (2008). A low dimensional description of globally coupled heterogeneous neural networks of excitatory and inhibitory neurons. *PLoS Comput. Biol.* 4:e1000219. doi: 10.1371/journal.pcbi.1000219
- Stienen, A. H., Schouten, A. C., Schuurmans, J., and Van Der Helm, F. C. (2007). Analysis of reflex modulation with a biologically realistic neural network. *J. Comput. Neurosci.* 23:333. doi: 10.1007/s10827-007-0037-7
- Tessadori, J., Bisio, M., Martinoia, S., and Chiappalone, M. (2012). Modular neuronal assemblies embodied in a closed-loop environment: toward future integration of brains and machines. *Front. Neural Circ.* 6:99. doi: 10.3389/fncir.2012.00099
- Ton, R., Deco, G., and Daffertshofer, A. (2014). Structure-function discrepancy: inhomogeneity and delays in synchronized neural networks. *PLoS Comput. Biol.* 10:e1003736. doi: 10.1371/journal.pcbi.1003736
- Traub, R. (1982). Simulation of intrinsic bursting in ca3 hippocampal neurons. *Neuroscience* 7, 1233–1242. doi: 10.1016/0306-4522(82)91130-7
- Vanni, M. P., Chan, A. W., Balbi, M., Silasi, G., and Murphy, T. H. (2017). Mesoscale mapping of mouse cortex reveals frequency-dependent cycling between distinct macroscale functional modules. *J. Neurosci.* 37, 7513–7533. doi: 10.1523/JNEUROSCI.3560-16.2017
- Vannucci, L., Falotico, E., and Laschi, C. (2017). Proprioceptive feedback through a neuromorphic muscle spindle model. *Front. Neurosci.* 11:341. doi: 10.3389/fnins.2017.00341
- Váša, F., Shanahan, M., Hellyer, P. J., Scott, G., Cabral, J., and Leech, R. (2015). Effects of lesions on synchrony and metastability in cortical networks. *Neuroimage* 118, 456–467. doi: 10.1016/j.neuroimage.2015.05.042
- Wang, Q., Ding, S.-L., Li, Y., Zeng, H., Harris, J. A., Ng, L. et al. (2020). The Allen mouse brain common coordinate framework: a 3D reference atlas. *Cell* 181, 936–953. doi: 10.1016/j.cell.2020.04.007
- Weidel, P., Djurfeldt, M., Duarte, R. C., and Morrison, A. (2016). Closed loop interactions between spiking neural network and robotic simulators based on music and ROS. *Front. Neuroinform.* 10:31. doi: 10.3389/fninf.2016.00031
- Wilson, H. R., and Cowan, J. D. (1972). Excitatory and inhibitory interactions in localized populations of model neurons. *Biophys. J.* 12, 1–24. doi: 10.1016/S0006-3495(72)86068-5
- Wong, K.-F., and Wang, X.-J. (2006). A recurrent network mechanism of time integration in perceptual decisions. *J. Neurosci.* 26, 1314–1328. doi: 10.1523/JNEUROSCI.3733-05.2006
- Wright, P. W., Brier, L. M., Bauer, A. Q., Baxter, G. A., Kraft, A. W., Reisman, M. D., et al. (2017). Functional connectivity structure of cortical calcium dynamics in anesthetized and awake mice. *PLoS ONE* 12:e0185759. doi: 10.1371/journal.pone.0185759
- Yang, H.-W., and Lemon, R. (2003). An electron microscopic examination of the corticospinal projection to the cervical spinal cord in the rat: lack of evidence for cortico-motoneuronal synapses. *Exp. Brain Res.* 149, 458–469. doi: 10.1007/s00221-003-1393-9
- Yasuda, R., Nimchinski, E. A., Scheuss, V., Pologruto, T. A., Oertner, T. G., Sabatini, B. L., et al. (2004). Imaging calcium concentration dynamics in small neuronal compartments. *Sci. STKE* 2004:p15. doi: 10.1126/stke.2192004p15
- Yilmaz, O., Birbaumer, N., and Ramos-Murguialday, A. (2015). Movement related slow cortical potentials in severely paralyzed chronic stroke patients. *Front. Hum. Neurosci.* 8:1033. doi: 10.3389/fnhum.2014.01033
- Zerlaut, Y., and Destexhe, A. (2017). Enhanced responsiveness and low-level awareness in stochastic network states. *Neuron* 94, 1002–1009. doi: 10.1016/j.neuron.2017.04.001
- Zimmermann, J., Ritter, P., Shen, K., Rothmeier, S., Schirner, M., and McIntosh, A. R. (2016). Structural architecture supports functional organization in the human aging brain at a regionwise and network level. *Hum. Brain Mapp.* 37, 2645–2661. doi: 10.1002/hbm.23200
- Zrenner, C., Belardinelli, P., Muller-Dahlhaus, F., and Ziemann, U. (2016). Closed-loop neuroscience and non-invasive brain stimulation: a tale of two loops. *Front. Cell. Neurosci.* 10:92. doi: 10.3389/fncel.2016.00092

Conflict of Interest: AA and EA was employed by company Fortiss GmbH.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2020 Allegra Mascaro, Falotico, Petkoski, Pasquini, Vannucci, Tort-Colet, Conti, Resta, Spalletti, Ramalingasetty, von Arnim, Formento, Angelidis, Blixhavn, Leergaard, Caleo, Destexhe, Ijspeert, Micera, Laschi, Jirsa, Gewaltig and Pavone. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Advantages of publishing in Frontiers



OPEN ACCESS

Articles are free to read
for greatest visibility
and readership



FAST PUBLICATION

Around 90 days
from submission
to decision



HIGH QUALITY PEER-REVIEW

Rigorous, collaborative,
and constructive
peer-review



TRANSPARENT PEER-REVIEW

Editors and reviewers
acknowledged by name
on published articles

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne | Switzerland

Visit us: www.frontiersin.org

Contact us: info@frontiersin.org | +41 21 510 17 00



REPRODUCIBILITY OF RESEARCH

Support open data
and methods to enhance
research reproducibility



DIGITAL PUBLISHING

Articles designed
for optimal readership
across devices



FOLLOW US

@frontiersin



IMPACT METRICS

Advanced article metrics
track visibility across
digital media



EXTENSIVE PROMOTION

Marketing
and promotion
of impactful research



LOOP RESEARCH NETWORK

Our network
increases your
article's readership