



Credibility, Replicability, and Reproducibility in Simulation for Biomedicine and Clinical Applications in Neuroscience

Lealem Mulugeta^{1*}, Andrew Drach², Ahmet Erdemir³, C. A. Hunt⁴, Marc Horner⁵, Joy P. Ku⁶, Jerry G. Myers Jr.⁷, Rajanikanth Vadigepalli⁸ and William W. Lytton^{9,10,11*}

¹ InSilico Labs LLC, Houston, TX, United States, ² The Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX, United States, ³ Department of Biomedical Engineering and Computational Biomodeling (CoBi) Core, Lerner Research Institute, Cleveland Clinic, Cleveland, OH, United States, ⁴ Department of Bioengineering and Therapeutic Sciences, University of California, San Francisco, San Francisco, CA, United States, ⁵ ANSYS, Inc., Evanston, IL, United States, ⁶ Department of Bioengineering, Stanford University, Stanford, CA, United States, ⁷ NASA Glenn Research Center, Cleveland, OH, United States, ⁸ Department of Pathology, Anatomy and Cell Biology, Daniel Baugh Institute for Functional Genomics and Computational Biology, Thomas Jefferson University, Philadelphia, PA, United States, ⁹ Department of Neurology, SUNY Downstate Medical Center, The State University of New York, New York, NY, United States, ¹⁰ Department of Physiology and Pharmacology, SUNY Downstate Medical Center, The State University of New York, New York, NY, United States, ¹¹ Department of Neurology, Kings County Hospital Center, New York, NY, United States

OPEN ACCESS

Edited by:

Sharon Crook,
Arizona State University, United States

Reviewed by:

Georg Hinkel,
FZI Forschungszentrum Informatik,
Germany
Upinder S. Bhalla,
National Centre for Biological
Sciences, India

*Correspondence:

Lealem Mulugeta
lealem@insilico-labs.com
William W. Lytton
bill.lytton@downstate.edu

Received: 01 February 2018

Accepted: 29 March 2018

Published: 16 April 2018

Citation:

Mulugeta L, Drach A, Erdemir A, Hunt CA, Horner M, Ku JP, Myers JG Jr., Vadigepalli R and Lytton WW (2018) Credibility, Replicability, and Reproducibility in Simulation for Biomedicine and Clinical Applications in Neuroscience. *Front. Neuroinform.* 12:18. doi: 10.3389/fninf.2018.00018

Modeling and simulation in computational neuroscience is currently a research enterprise to better understand neural systems. It is not yet directly applicable to the problems of patients with brain disease. To be used for clinical applications, there must not only be considerable progress in the field but also a concerted effort to use best practices in order to demonstrate model credibility to regulatory bodies, to clinics and hospitals, to doctors, and to patients. In doing this for neuroscience, we can learn lessons from long-standing practices in other areas of simulation (aircraft, computer chips), from software engineering, and from other biomedical disciplines. In this manuscript, we introduce some basic concepts that will be important in the development of credible clinical neuroscience models: reproducibility and replicability; verification and validation; model configuration; and procedures and processes for credible mechanistic multiscale modeling. We also discuss how garnering strong community involvement can promote model credibility. Finally, in addition to direct usage with patients, we note the potential for simulation usage in the area of Simulation-Based Medical Education, an area which to date has been primarily reliant on physical models (mannequins) and scenario-based simulations rather than on numerical simulations.

Keywords: computational neuroscience, verification and validation, model sharing, modeling and simulations, Simulation-Based Medical Education, multiscale modeling, personalized and precision medicine, mechanistic modeling

INTRODUCTION

One hallmark of science is reproducibility. An experiment that cannot be reproduced by others may result from statistical aberration, artifact, or fraud. Such an experiment is not credible. Therefore, reproducibility is the first stage to ensure the credibility of an experiment. However, reproducibility alone is not sufficient. For example, an *in vitro* experiment is performed to advance or aid the understanding of *in vivo* conditions. However, the applicability of *in vitro* results to the living tissue may be limited due to the artifact of isolation: a single cell or a tissue slice extracted from its environment will not function in precisely the way it functioned *in situ*. In medicine, animal models of a disease or treatment are frequently used, but may not be credible due to the many differences between the human and the monkey, rat, or another animal that may limit the transfer of findings from one species to another.

In the computational world, the credibility of a simulation, model or theory depends strongly on the projected model use. This is particularly true when translating a model from research usage to clinical usage. In research, innovation and exploration are desirable. Computer models are used to introduce or explore new hypotheses, ideally providing a new paradigm for experimentation. In this setting, the most important models may in some cases be the less credible ones – these are the models that stretch understanding by challenging the common view of how a particular system works, in order to offer a paradigm shift. Here, *prima facie* credibility is in the eye of the beholder, who is more likely representing the views of the community – the dominant paradigm.

In the clinical realm, by contrast, establishing credibility is of paramount importance. For pharmaceuticals, credibility is currently established through evidence-based medicine (EBM), ideally through double-blind trials with large numbers of patients. The downside of this statistical approach is that it necessarily aggregates a large number of disparate patients to achieve statistical significance. In some cases, this has resulted in tragedy, as a subgroup with particular genetics has a fatal response to a drug that is beneficial in the overall group (e.g., rofecoxib, brand-name Vioxx). As EBM gives way to precision medicine, pharmaceutical credibility will be established in each subgroup, or even at an individual level, to enhance safety. However, to establish pharmaceutical reliability for personalized medicine (precision with a subgroup of $n = 1$), a lack of comparator precludes the use of data-mining by definition. Simulations based on patient genetics and various levels of epigenetics up through brain connectomics will then be the only way to predict the response of an individual patient to a particular treatment. Such patient simulation would provide a prediction of the response of that patient's unique physiodynamics to a therapy. The credibility of such models will be essential.

In addition to pharmacotherapy, brain disease treatment also utilizes other therapeutic modalities, ranging from neurosurgery to the talk therapy of psychiatry and clinical psychology. While the latter will likely remain beyond the range of our modeling efforts, neurosurgery has already begun to benefit from modeling efforts to identify locations and pathways for

epilepsy ablation surgery (Jirsa et al., 2017; Lytton, 2017; Proix et al., 2017). Another set of therapeutic approaches that are likely to benefit from modeling are the electrostimulation therapies that are finding increased use in both neurology and psychiatry, e.g., deep brain stimulation (DBS), transcranial magnetic stimulation (TMS), transcranial direct/alternating current stimulation (tDCS/tACS), and electroconvulsive therapy (ECT) (Kellner, 2012). Neurorehabilitation will also benefit from modeling to help identify procedures to encourage neuroplasticity at locations where change will have the greatest effect for relief of deficit.

In most respects, the issues of credibility, replicability, reproducibility for computational neuroscience simulation are comparable to those faced by researchers using simulation to study other organ systems. However, the complexity of the brain and several special features provide unique challenges. As with modeling of other organ systems, the relevant length scales range from molecular (angstrom) level up through tissue (centimeter) levels. Many experiments extend from the lowest to highest scales, for example evaluating performance on a memory task as a drug modifies ion channel activity at the molecular level. Compared to other organ systems, there is a particularly broad range of temporal scales of interest: 100 microseconds of sodium channel activation up to years of brain maturation and learning. In addition to physical scales of the central nervous system (CNS) itself, brain research includes further investigative levels of cognition, information, and behavior. An additional aspect of nervous system organization involves overlap between scales, which prevents encapsulation of one scale for inclusion in the higher scale. For example, spatiotemporal activity in apical dendrites of neocortical pyramidal cells (subcellular level) are co-extensive in both spatial and temporal scales with the scales of the local network (supracellular level).

In this paper, we will focus on the many issues of model credibility from a biomedical and clinical perspective. We will use *model* here forward to mean a mathematical model, primarily analyzed *in silico* via a *simulation*, which is the numerical instantiation of a mathematical model on a computer. We will identify explicitly in cases where we are discussing other types of models: verbal models, animal models, physical models, etc. Currently, there are still relatively few models of brain disease, and those remain in the research domain, rather than being practical clinical tools. Therefore, we are considering policy and practice for a future *clinical computational neuroscience*, based on the current uses of computational neuroscience in basic biomedical research, and on the clinical usage of simulations in other domains of medicine. In doing this, we will introduce some basic concepts that are important in the development of credible models: Reproducibility and Replicability; Verification and Validation.

REPRODUCIBILITY AND REPLICABILITY

Replicability, here subsuming repeatability, is the ability to achieve a fully identical result (Plessner, 2017). For example, in the case of neuronal network simulation, a simulation has been

fully replicated when one can show that spike times, as well as all state variables (voltages, calcium levels), are identical to those in the original simulation. Replicability is a design desideratum in engineering when one wants to ensure that a system being distributed runs identically to a prototype system (Drummond, 2009; Crook et al., 2013; McDougal et al., 2016a).

Reproducibility, in contrast, is the ability of a simulation to be copied by others to provide a simulation that provides the same *results* (Crook et al., 2013; McDougal et al., 2016a). This will then depend on what one considers to be a result, which will reflect the purposes of the simulation and will involve some measures taken on the output. Taking again the case of the neuronal network simulation, a result might involve some direct statistical measures of spiking (e.g., population rates), perhaps complemented by the summary statistics provided by field potential spectra, likely ignoring state-variable trajectories or their statistics.

Replicability and reproducibility are inversely related. A turnkey system, provided on dedicated hardware or a dedicated virtual machine, will run identically every time and therefore be fully replicable. However, such a system will not be reproducible by outsiders, and may in some cases have been encrypted to make it difficult to reverse engineer. Generally, the higher level the representation is, the more readily other groups understand a simulation and reproduce the results, but the less likely it is that they obtain an identical result – lower replicability. Representations using equations – algebra, linear algebra, calculus – are identical worldwide and therefore can provide the greatest degree of reproducibility by any group anywhere. However, reproducing a simulation from equations is a laborious process, more difficult than reproducing from algorithms, which is, in turn, more laborious than reproducing from a declarative (e.g., XML) representation. A dedicated package in the domain which provides a declarative description of a simulation will be more easily ported than general mathematical declarative description for dynamical systems. Even at the level of reproduction from software code, differences in numerical algorithms used in computer implementations will lead to somewhat different results.

Different software representations also provide different levels of difficulty in the task of reproducing a simulation by porting to another language. The major innovation in this respects has been the development and adoption of object-oriented languages such as Java and Python. In particular, the use of object inheritance allows the definition of a type (e.g., a cell) with a subsequent definition of particular subtypes, where each type has parameter differences that are easily found and can be clearly documented by provenance.

Some difficulties with precise replicability of simulations are common to many different simulation systems. In particular, a model that uses pseudo-random numbers will not replicate precisely if a different randomizer is used, if seeds are not provided, or if randomizers are not handled properly when going from serial to parallel implementations. One difficulty peculiar to neural simulation is related to the strong non-linearities (and numerical stiffness) associated with action potential spike thresholding. Spiking networks are sensitive to round-off error; a

single time-step change in spike time will propagate to produce changes in the rest of the network (London et al., 2010).

In general, specific simulation programs have enhanced reproducibility by providing purpose-built software to solve the particular problems of the computational neuroscience domain. Typically, these packages couple ordinary differential equation (ODE) solvers for simulating individual neurons with an event-driven queuing system to manage spike event transmission to other neurons in neuronal networks. These facilities are provided by neuronal network simulators such as BRIAN (Goodman and Brette, 2008), PyNN (Davison et al., 2008), and NEST (Plesser et al., 2015), which allow spiking neurons to be connected in networks of varying size. It should be noted that these simulators are very different from the artificial neural networks used in deep learning, which do not implement spiking neurons. Some other packages also add the ability to do more detailed cellular simulation for the individual neurons by adding the partial differential equations (PDE) needed to simulate the internal chemophysiology that complements the electrophysiology of spiking – NEURON (Carnevale and Hines, 2006) and MOOSE (Dudani et al., 2009) provide this additional functionality. Many computational neuroscience simulations are still carried out using general-purpose mathematical software, e.g., Matlab (The Mathworks, Inc., Natick, MA, United States); or by more general computer languages such as FORTRAN, C, or C++, limiting their reproducibility, reusability, and extensibility. However, it should also be noted that the popularity of software also plays a role. A language or package that is widely used will increase the number of people that can easily contribute to developing or utilizing the simulations.

Since computational neuroscience simulations are often very large, extensibility to high-performance computing (HPC) resources is also desirable and will enhance reproducibility. Some current simulator tools offer a direct path to these larger machines. NetPyNE, a declarative language built on top of NEURON, provides a general description of a network that can be run directly either on a serial processor or, via MPI (message passing interface), on an HPC resource (Lytton et al., 2016). The Neuroscience Gateway Portal provides a shared, NSF-supported graphical resource that simplifies HPC for a variety of simulators, including NetPyNE, NEURON, NEST, BRIAN, and others, avoiding the need for the user to know MPI usage (Carnevale et al., 2014).

Hypothetically, journal articles permit reproducibility using equations given in the Methods section. Often, however, full details are not provided due to the enormous complexity of information associated with many different cell types, and complex network connectivity (McDougal et al., 2016a). Furthermore, parameters and equations may be given for one figure, but not for all figures shown. Journal articles may also have typographical or omission errors. And even when the document is complete and entirely without error, errors are likely to creep in when reproduction is attempted, and the model is typed or scanned back into a computer. For all of these reasons, an electronic version of a model is more accurate and more immediately usable than a paper copy. Some journals, and many individual editors or reviewers, require software sharing as part

of the publication process. However, some authors resist this mandate, desiring to retain exclusive access to their intellectual property.

Databases of models and parts of models have become important and have encouraged reproducibility in computational neuroscience (Gleeson et al., 2017). Additional value is added by utilizing formal model definitions such as ModelML, CellML, NeuroML, VCML, SBML (Hucka et al., 2003; Zhang et al., 2007; Lloyd et al., 2008; Moraru et al., 2008; Gleeson et al., 2010; Cannon et al., 2014). Major databases are being provided by the Human Brain Project, the Allen Brain Institute, the International Neuroinformatics Coordinating Facility, and others. Other databases include the NeuroMorpho.Org database of neuronal morphologies, the ModelDB database of computational neuroscience models (see this issue) in a variety of simulation packages and languages, and the Open Source Brain database for collaborative research (Gleeson et al., 2012, 2015; Gleeson et al., unpublished). We discuss these resources further in the section discussing the “Role of the Community.”

A recent initiative to encourage reproducibility in science is the new *ReScience Journal* (rescience.github.io), which publishes papers that replicate prior computational studies. By hosting the journal on GitHub, new implementations are directly available alongside the paper, and alongside any ancillary materials identifying provenance or providing documentation. Recently, the classic Potjans-Diesmann cortical microcircuit model was ported from NEST to BRIAN, reproducing and confirming the primary results (Cordeiro et al., unpublished).

GOOD PRACTICES CONTRIBUTING TO SIMULATION CREDIBILITY

Verification and Validation (V&V)

Verification

Verification and validation (V&V) help users demonstrate the credibility of a computational model within the context of its intended use. This is accomplished by assessing the quantitative and qualitative aspects of the model that influence model credibility. The process of establishing the model’s correctness and its capability to represent the real system is accomplished through the processes of verification, validation, uncertainty propagation, and sensitivity analysis. Of these, V&V represent the most well-known, and potentially confused, aspects of model assessment.

Computational models may be implemented using open-source or commercial (off-the-shelf) software, custom (in-house) code, or a combination of the two (modified off-the-shelf software). Verification assures that a computational model accurately represents the underlying mathematical equations and their solution on a specific code platform. Verification also emphasizes confirmation of parameter specification and the accurate translation of model data from model data sources into the model application. Full verification for all possible inputs is technically not possible due to the halting problem. Nonetheless, software implementation

of model concepts should always include some level of code verification and, to the extent possible, follow best management practices and established quality-assurance processes. In the case of commonly used simulation packages, verification will be the responsibility of the platform developer and not of the user, but unanticipated usage can bring verification concerns back to the fore in particular cases.

Verification can be divided into two sequential steps: code verification and calculation verification. *Code verification*, initially performed by the software platform developer provides evidence that the numerical algorithms implemented in the code are faithful representations of the underlying physical or conceptual model of the phenomenon. Code verification should be repeated by the user in the case of novel usage of the simulator. Code verification establishes the reliability of the source code in representing the conceptual model, including relevant physics, for the problem. Ideally, benchmark problems with analytical solutions are employed to ensure that the computer code generates the correct solution at the specified order of accuracy. For example, a common reference in computational fluid dynamics (CFD) is laminar flow in a straight pipe with a circular cross-section, whose analytical solution is well-known (Bird et al., 1960; Stern et al., 2006). CFD modeling techniques are being used to provide insight into flow patterns and rupture-risk of cerebral artery aneurysms (Campo-Deaño et al., 2015; Chung and Cebal, 2015) and for interventional radiology planning (Babiker et al., 2013). Unfortunately, no analytic solutions are available for most computational neuroscience applications. Therefore, one is restricted to comparing results from one numerical approximation to that of another, without reference to any ground truth (Brette et al., 2007).

Next, *calculation verification* aims to estimate numerical and logical errors associated with the conceptual model implementation, i.e., the computational model representing the target application. Going back to the laminar pipe flow example, one would specify geometry, material properties, and loading conditions to match the problem at hand. This typically results in a problem that no longer has an analytical solution but must be solved numerically. Various aspects of the numerical representations, particularly discretization, are investigated and refined until the model is deemed to be accurate within a pre-specified tolerance. Upon completion of the calculation verification step, the developer has established (and should document) a bug-free implementation of the model concepts with reasonable parameter values.

One of aspect of verification that is often overlooked in the computational science community is the testing of model scripts and binary codes. Researchers tend to focus their attention on V&V in the context of overall program performance, but omit testing the functionality of individual software modules. Module verification can be implemented as a suite of tests which verify the functionality of individual functions and modules (unit tests) and their integration within the system (integration tests). It is common practice to automate the testing procedures using an automated testing framework to perform these tests after each version update.

Validation

Everyday vernacular often equates or confuses the terms “verification” and “validation.” As described in the previous section, *verification* seeks to ensure the quality of the implementation of the conceptual model. Meanwhile, *validation* seeks to assess how well the conceptual model and its implementation on a specified code platform represent the real-world system for a defined application. More rigorously stated, validation is an assessment of the degree to which the model and simulation is an accurate representation of the response of the real system within the intended context of use. In this case, validation is a comparative process that defines a qualitative or quantitative measure of how the model differs from an appropriate experimental or other data source – the *referent*, generally a series of experiments in our context. Validation also helps to ensure that the computational model has *sufficient* rigor for the context of use, although a more rigorous or more precise model is not necessarily more credible.

The definition of an appropriate referent is a critical aspect of model validation. Ideally, a validation referent consists of data obtained from a system with high similarity to the system being modeled in an environment similar to that of the target system. In clinical computational neuroscience, this is often difficult since data is obtained from a rodent (not high similarity to human), and sometimes from a slice (environment not similar to *in vivo* situation). The data used should be considered to be high quality by the model end-user community, and should represent data not used in model development. This separates *design data* from *testing data* (also called fit vs. benchmark data, or calibration vs. validation data). Model limitations due to inadequacies of the validation referent should be communicated. Practitioners should also keep in mind that a model validation, or the understanding of the variability of the model in predicting real world response, is only valid in the area of the referents used in the validation process, as illustrated by the Validation Domain (**Figure 1**). The Application Domain may be larger than the Validation Domain; it establishes the range of input and output parameters relevant to the context of use of the computational model. As the application of the model deviates from the situational context described by the referent, the influence of the model validation information will also change.

In most cases, the more quantitative the comparison, the stronger the case that a model and simulation is contextually valid. Organizations such as the American Society of Mechanical Engineers have developed standards to guide the model practitioner in performing successful model validation activities for specific application domains¹. These comparisons range from qualitative, graphical comparative measures to quantitative comparative measures relying on statistical analysis of referent and model variances, the latter obtained over a wide range of input parameter variation (Oberkampf and Roy, 2010). Ideally, the end-user community and regulatory agencies will play a role in assessing the adequacy of the validation based on the context

of an expected application and the influence the model has on critical decision-making.

Aspects of Good Practice for Credibility Software Aspects

The credibility of simulation results requires the reliability of simulation protocols and software tools. Good software practices include version control; clear, extensive documentation and testing; use of standard (thoroughly tested) software platforms; and use of standard algorithms for particular types of numerical solutions. It is also desirable to rely on existing industry standards and guidelines and to compare simulation results against competing implementations. To maintain the highest level of credibility, one would establish and follow these practices at every step of the development, verification, validation, and utilization of the simulation tools.

Version control is an approach for preserving model development and use histories, which can also be useful for tracing the provenance of model parameters and scope of applicability. There exists a large number of version control systems (VCS) which provide on-site, remote, or distributed solutions (e.g., Git, SVN, and Mercurial). In general terms, these systems provide tools for easy traceability of changes in individual files, attribution of modifications and updates to the responsible author, and versioning of specific snapshots of the complete system. Use of a VCS is recommended for both development (troubleshooting of bugs) as well as the day-to-day use of the modeling tools (monitoring of modeling progress).

Good documentation can be aided by the rigorous use of a detailed, dated electronic laboratory notebook (e-notebook). The e-notebook can provide automatic coordination with software versions and data output. E-notebooks are supported through various software packages, notably the Python Jupyter notebook and the Emacs org-mode notebook (Schulte and Davison, 2011; Stanic et al., 2015; Kluyver et al., 2016). A major advantage over the traditional paper notebook is that the e-notebook can be directly integrated into simulation workflow, and will also provide direct pointers to simulation code, output, figures, data, parameter provenance, etc. This then allows later reviewers to identify all these links unambiguously. However, compared with a paper notebook, the e-notebook is at greater risk of falsification due to later rewriting. This risk can be reduced by including the e-notebook in the VCS, and can be eliminated by using blockchain technology (Furlanello et al., 2017). An e-notebook will also include informative records of model development and implemented assumptions; hypotheses and approaches to testing the hypotheses; model mark-up; detailed descriptions of the input and output formats; and simulation testing procedures. Going beyond the e-notebook, but also linked through it, the developer may add case studies, verification problems, and tutorials to ensure that other researchers and practitioners can learn to use the model.

It should be noted that even models developed following the aforementioned guidelines will have application bugs and usability issues. Thus, it is valuable to also cross-verify simulation results using alternative execution strategies and competing

¹<https://cstools.asme.org/csconnect/CommitteePages.cfm?Committee=100108782>

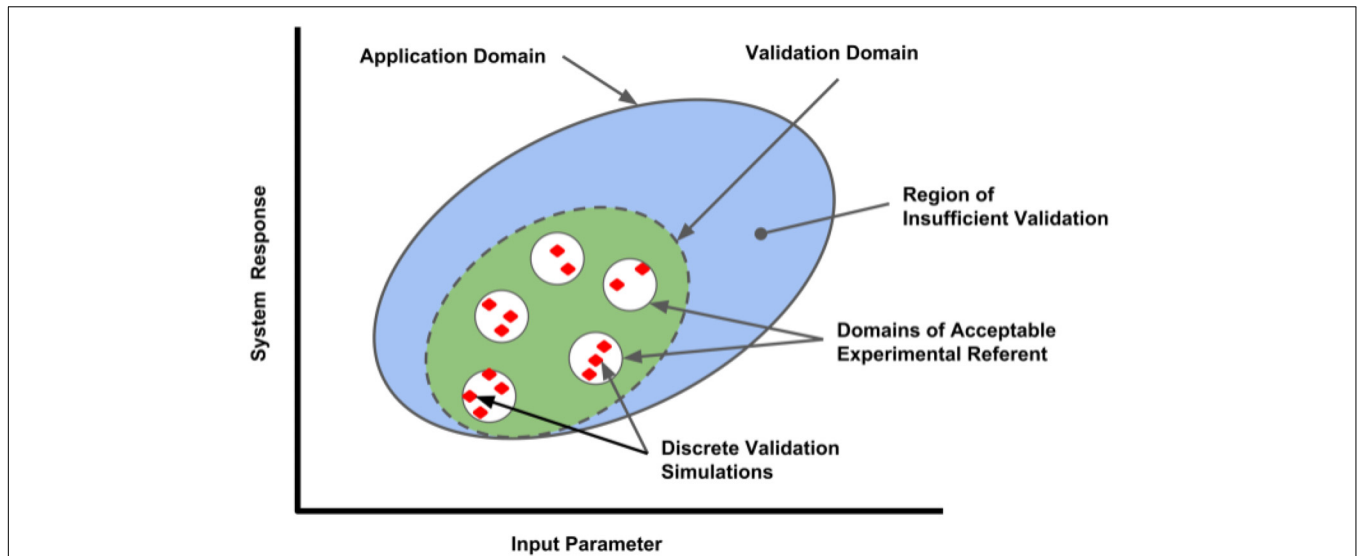


FIGURE 1 | Validation domain of a computational model for a range of input parameters and outputs (system responses). Red diamonds represent the validation set points, where comparisons between the computational model and applicable referents were conducted by discrete simulations performed in each referent sub-domain. The range of parameters evaluated establishes the Validation Domain (green ellipse), which will define the extent of the Application Domain (large blue ellipse) where model performance has established credibility. Applications of the model outside of the Application Domain lack validation and have lesser credibility.

model implementations – e.g., run a simulation using BRIAN and NEST (Cordeiro et al., unpublished) – to reduce the chance of obtaining spurious simulation results. In addition to the inter-model verification, simulation process should be governed by generally applicable or discipline-specific standard operating procedures, guidelines, and regulations.

Developing Credible Mechanism-Oriented Multiscale Models: Procedure and Process

In science, an explanation can be inductive, proceeding from repeated observation. Ideally, however, explanation precedes prediction, permitting deductive reasoning (Hunt et al., 2018). Simulation of a mechanistic multiscale model provides an explicit way of connecting a putative explanatory mechanism to a phenomenon of interest.

Credibility and reproducibility can be enhanced by taking note of the many factors and workflows required to build a credible simulation to be used in a clinical application. One of these, often overlooked, is the role of exploratory (non-credible) simulations in building credible simulations. We would argue that most of the simulations that have been done in computational neuroscience are exploratory, and that we can now begin winnowing and consolidating these to create credible simulations for clinical application.

Unfortunately, the problems in biology and particularly in neuroscience are characterized by (1) imprecise, limited measures – for example, EEG measures 6 cm of cortex at once (Nunez and Srinivasan, 2005); (2) complex observations whose relevance is sometimes unclear – there is no broad agreement on the relevance of particular bands of brain oscillations (Weiss and Mueller, 2012); (3) sparse, incommensurable and sometimes contradictory supporting information – for example,

the difficulties of connecting microconnectomic (<1 mm; axon tracing, dual impalement, etc.) with macroconnectomic (1 cm; functional magnetic resonance imaging, fMRI) measures; and (4) high degree of uncertainty – parameter values obtained from brain slice work may differ from *in vivo* values. These limitations (left of ranges in **Figure 2**) contrast with the more solid information, concepts, observables and lower uncertainty associated with “classical” engineering of man-made devices such as computers, cars, and aircraft.

Model development is difficult, involving the need to consider and consolidate a large variety of factors from

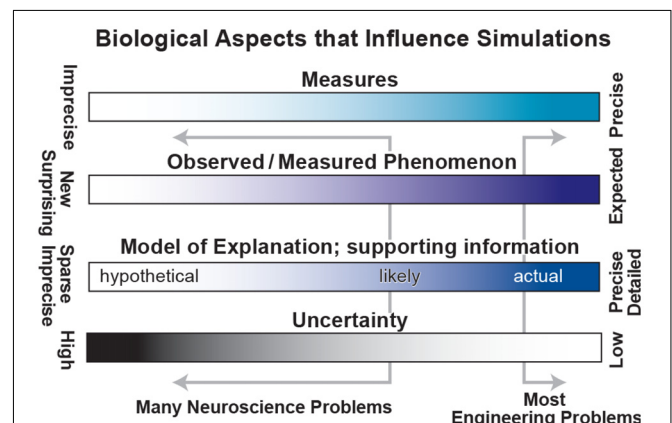


FIGURE 2 | Spectra to characterize biological aspects of interest. Classical engineering problems lie at the right side of each range, working with precise measures, strong expectations, detailed information and low uncertainty. Unfortunately, most neuroscience problems lie far to the left with weak measures, unclear phenomenology, sparse information and high uncertainty.

biology, engineering, mathematics, and design under the many constraints due to the limitations mentioned above. We have identified a set of procedures for the building of credible simulations, breaking out the many sub-workflows and processes involved (**Figure 3**). One can spend years building tools and running exploratory simulations that only lay the groundwork for future credible models. Nonetheless, it is important not to lose sight of the goal, which is to explain brain phenomenology at one or more levels of description: electrical rhythms, movement, behavior, cognition, etc.

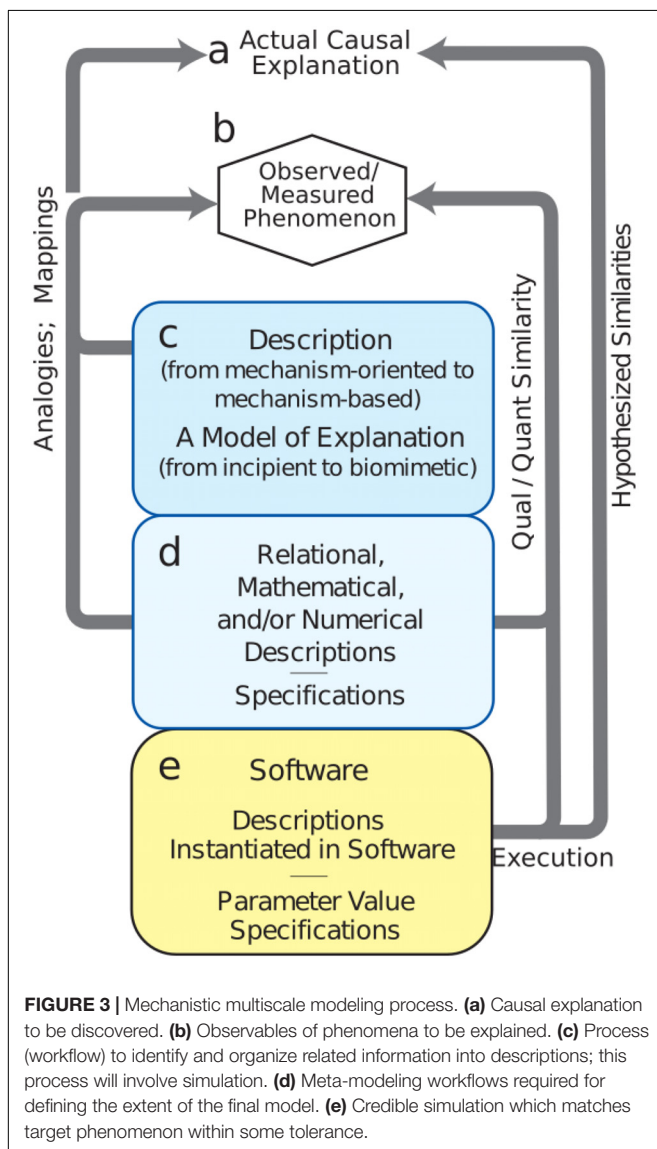
The first task is to specify phenomena to be explained (**Figure 3b** – Observed Phenomenon). From this perspective, potentially relevant biological aspects are then organized together with relevant information and data into incipient explanations (**Figure 3c** – Descriptions and Explanations). In the computational neuroscience community, there are multiple perspectives regarding what information is to be considered

relevant. For example, some argue that dendritic morphology and the details of ion channels are critical for understanding cortical networks (Amunts et al., 2017), while others consider that one need only consider simplified spiking cells (Diesmann et al., 1999; Potjans and Diesmann, 2014; Cain et al., 2016), and still others that it is best to work with high-level dynamical representations of populations (Shenoy et al., 2013) or mean-field theory (Robinson et al., 2002). Indeed all of these perspectives can be regarded as part of the explanatory modeling that will find its way into new concepts of (1) what is considered to be a relevant observation or measurement (**c to b in Figure 3**) and (2) what will be considered to be the form of an eventual causal explanation (**c to a**). Development of this incipient explanation will involve establishing mappings and drawing analogies between features of the explanation and particular measurements. These mappings and analogies may then be extended to provide working hypotheses and to actual preliminary biomimetic simulations for the eventual causal explanations.

A set of additional considerations (**Figure 3d** – Numerics and Specifications) provide the bridge from the exploratory activities (**Figure 3c** – Descriptions) to final credible models of (**Figure 3e** – Software). Although illustrated toward the bottom, these aspects of project formulation should be considered from the start as well. In computational neuroscience, potential use cases are still being developed and differ considerably across the four major clinical neuroscience specialties. Use cases, and data availability, will identify phenomena to be considered. For example, access to electroencephalography (EEG), but not electrocorticography, changes not only the type of software to be developed, but also the types of explanations to be sought. Use cases will also need to be organized based on expectations, separating near-term and long-term needs. New computational neuroscience users, use cases, and applications will arise in other specialties as they consider the innervation of other organs (Barth et al., 2017; Samineni et al., 2017; Vaseghi et al., 2017; Ross et al., 2018).

While identification of users and use cases remains relatively underdeveloped in computational neuroscience, the development of simulation tools is quite sophisticated. The need for multialgorithmic as well as multiphysics simulation has required that simulation platforms combine a variety of numerical and conceptual techniques: ODEs, PDEs, event-driven, graph theory, information theory, etc. Simulation techniques have been developed over more than a half-century, starting with the pioneering work of Hodgkin and Huxley (1952), Fitzhugh (1961), Rall (1962), and others. Today, we have a large variety of simulators with different strengths (Carnevale and Hines, 2006; Brette et al., 2007; Davison et al., 2008; Goodman and Brette, 2008; Bower and Beeman, 2012; Plesser et al., 2015; Tikidji-Hamburyan et al., 2017) that can be used individually or in combination, cf. MUSIC (Djurfeldt et al., 2010).

During the final stage of credible model development (**Figure 3e**), we demonstrate that the model provides the desired outputs to represent observations (**e to b**). A typical requirement is that simulation outputs agree with target phenomenon measurements within some tolerance. Since the simulation system will be multi-attribute and multiscale, it will at the very least begin providing mechanism-based, causal understanding



across measures. To the extent that the model is truly biomimetic, direct mappings will exist to specific biological counterparts – voltages, spike times, field potential spectra, calcium levels, or others.

Although credible software appears to be the end of the road, actual practice will require that the resulting software system undergoes many rounds of verification, validation, refinement, and revision before even being released to users. From there, continued credibility requires continuing work on documentation, tutorials, courses, bug reports and bug fixes, requested front-end enhancements, and identified backend enhancements.

ROLE OF THE COMMUNITY

Community involvement is important to establish the credibility of the modeling and simulation process, and of the models themselves. The peer-review publication process serves as a traditional starting point for engaging the community. However, sharing of models, along with the data used to build and evaluate them, provides greater opportunities for the community to directly assess credibility. The role of sharing models and related resources has been acknowledged in the neurosciences community (McDougal et al., 2016a; Callahan et al., 2017). Related resources for sharing include documentation of the model (commonly provided in English) and implementation encoded in its original markup or code. With access to the model and its associated data and documentation, interested parties can then assess the reproducibility and replicability of the models and the simulation workflow first-hand. Further, reuse of the models and extensions of the modeling and simulation strategies for different applications can reveal previously unknown limitations. These various community contributions help build up the credibility of a model.

It is also important to note the value of cross-community fertilization to establish a common ground for credible modeling and simulation practices, to conform to and evolve standards that promote a unified understanding of model quality. Such standards enable individuals to easily exchange and reuse a given model on its own, or in combination with other models. Especially in multi-scale modeling, the ability to trust and build upon existing models can accelerate the development of these larger-scale efforts. Organizations such as the Committee on Credible Practice for Modeling and Simulation in Healthcare are leading efforts to establish such standards (Mulugeta and Erdemir, 2013).

Most community involvement in computational neuroscience has come through the establishment of databases and other resources that have encouraged submissions from the overall community of researchers. For example, the Scholarpedia resource established by Izhikevich and collaborators has hosted articles on computational neuroscience concepts and techniques that have been used to share concepts, modeling techniques, and information about particular modeling tools (Gewaltig and Diesmann, 2007; Wilson, 2008; Seidenstein et al., 2015). The Open Source Brain project provides a central location

for collaborators working on modeling the nervous system of particular brain areas or of whole organisms, notably the OpenWorm project for modeling the full nematode nervous system (Szigeti et al., 2014). The Human Brain Project, an EU effort, has established a number of “Collaboratories,” web-accessible platforms to curate models and conduct simulations, to encourage community involvement in coordinated projects (Amunts et al., 2017). One of these projects aims to identify the parameters underlying individual synaptic events recorded in voltage clamp experiments (Lupascu et al., 2016). The Allen Brain Institute, another large modeling and data collection center, shares all results with the community, even before publication. Most of the major simulator projects encourage contributions from the community to provide either simulator extensions or additional analytic tools. For example, the SenseLab project hosts a SimToolDB alongside ModelDB for sharing general simulation code (Nadkarni et al., 2002). ModelDB itself is a widely used resource which specifically solicits model contributions and then provides a starting point for many new modeling projects that are extensions or ports of existing models (Peterson et al., 1996; McDougal et al., 2016b).

The above databases are used to provide completed models that are designed to be stand-alone but can also be used as components of larger models. By contrast, detailed neuron morphologies and ion channel models are generally only used as starting points for other models to build models at higher scales. Examples of these databases include the NeuroMorpho.Org database of neuronal morphologies and the Channelpedia database of voltage-sensitive ion channels (Ascoli et al., 2007; Ranjan et al., 2011).

The availability of these valuable resources is a testament to the successful engagement of the computational neuroscience community. A coming challenge will be to provide mechanisms for the discovery and selection of appropriate models for defined contexts among the existing hundreds of models. Here again, community involvement can play a critical role by providing the feedback and assessments of a model and its credibility to aid others in deciding whether to, or how to, re-use a model or part of a model.

USE OF SIMULATION IN MEDICAL EDUCATION

Simulation-Based Medical Education (SBME) is rapidly growing, with applications for training medical students, and residents (Jones et al., 2015; Yamada et al., 2017). However, the use of the word *simulation* in SBME differs from our usage above. SBME is referring to *simulated reality*: paper exercises based on protocols; mannequins; re-enactments with live actors for physical exams or major disasters; detailed computer-based virtual reality training, similar to video games. Currently, even the most advanced mannequins and computer-generated simulations have very limited capacity to produce a realistic focal neurological deficit or combination of signs and symptoms. Design and implementation of SBME tools is especially challenging in brain disease due to the complexity of the brain, and the variety of its responses to

insult (Chitkara et al., 2013; Ermak et al., 2013; Fuerch et al., 2015; Micieli et al., 2015; Konakondla et al., 2017). This complexity suggests that back-end multiscale modeling would be particularly valuable in SBME development for brain disease.

Mechanistic multi-scale modeling has been used in both pre-clinical and clinical education to explain disease causality. For example, a demyelination simulation, relevant to multiple sclerosis and Guillain-Barre syndrome, is available that demonstrates both the effects of demyelination on action potential propagation and the consequent increase in temperature sensitivity (Moore and Stuart, 2011). Although this model has not been linked directly to a hands-on SBME activity, it could be linked to a training activity for Nerve Conduction Studies. Another promising area of recent research interest is in the area of neurorobotics, where robotic arms have been linked to realistic biomimetic simulations in the context of neuroprosthetics (Dura-Bernal et al., 2014; Falotico et al., 2017). In the clinical neurosciences, epilepsy is one of the most successfully modeled disorders in neurology and neurosurgery (Lytton, 2008). Jirsa et al. (2017) have pioneered models of individual patients prior to epilepsy surgery – see Section “Personalized Medicine Simulation for Epilepsy.” These simulations could now be extended into training protocols for neurology and neurosurgery residents, offering an opportunity for melding educational simulation with computational neuroscience.

Current SBME efforts are focused on mannequins. A generic mannequin was used to train medical students to manage differential diagnoses and emergency procedures for status epilepticus and acute stroke (Ermak et al., 2013). The mannequin used was not capable of mimicking visible, or electrographic, signs of stroke or seizures. Instead, students were given chart data and simulation actors playing family member. Such simulations necessarily fall short on one aspect of effective implementation of SBME (Issenberg et al., 2005): the degree to which the simulation has a “real-life” feel.

In the future, life-like and highly immersive SBME will facilitate the learning of dangerous medical procedures, including emergencies and recovery from mistakes, in the way that is currently done with simulators used to train pilots, astronauts and flight controllers. Ideally, neurology and neurosurgery (and eventually psychiatry and physiatry) SBME systems will produce learned skills that are transferable to patient care. The predictive validity of the simulation could then be assessed by comparing performance measured under simulation conditions with corresponding measurements made on real patients (Konakondla et al., 2017). However, the skill level of a learner in the simulator should not be taken as a direct indication of real-life skill performance. In one example from flight training, a particular technique (full rudder), which served to “game” the simulator, resulted in a fatal accident due to tail separation when used in real life (Wrigley, 2013).

An example of SBME in neurosurgical training is the Neuro-Touch surgical training system developed by the National Research Council of Canada (Konakondla et al., 2017). The systems is built around a stereoscope with bimanual procedure

tools that provide haptic feedback and a real-time computer-generated virtual tissue that responds to manipulation. As the surgeon is working through a surgical scenario, the simulator records metrics for detailed analysis to develop benchmarks for practitioners at different stages of training.

The Neurological Exam Rehearsal Virtual Environment (NERVE) virtual-patient SBME tool was developed to teach 1st- and 2nd-year medical students how to diagnose cranial nerve deficits (Reyes, 2016). Educational results were validated using questionnaires designed for virtual patient simulators (Huwendiek et al., 2014). An interesting future extension would be to provide an underlying simulator that would take account of the many complex neurological deficits found in patients due to the anatomical confluence of tracts in the brainstem. Such a simulator would be useful for neurology residents as well as for medical students.

To advance both the technologies and methodologies applied in SBME, the Society for Simulation in Healthcare (SSH) recently established the Healthcare Systems Modeling and Simulation Affinity Group. The Committee on Credible Practice of Modeling and Simulation in Healthcare (Mulugeta and Erdemir, 2013) has been collaborating with the SSH community by providing guidance on how to design and implement explicit multiscale computational models into traditional SBME systems. Additionally, the Congress of Neurological Surgeons has formed a Simulation Committee to create simulations for resident education (Konakondla et al., 2017). The US Food and Drug Administration (FDA) and the Association for the Study of Medical Education have also been working to publish standards and guidelines for regulatory submissions involving computational models and simulations for healthcare applications (Hariharan et al., 2017).

USE OF MODELING IN CLINICAL DOMAINS OF BRAIN DISEASE

Simulations in the clinical neuroscience domain have largely focused on accounting for the neural activity patterns underlying brain disease. Testing the predictions arising from the simulation is dependent on technological advances in neuromodulation, pharmacology, electrical stimulation, optogenetic stimulation, etc. Neuropharmacological treatment is systemic, with effects wherever receptors are found, often peripherally as well as centrally. Although targeted treatment with an implanted cannula is possible, it is not widely used clinically. By contrast, electrical stimulation can be highly targeted with a local placement of electrodes. Development of closed-loop systems and devices for brain stimulation (Dura-Bernal et al., 2015), are currently being used and show promise in treating a wide range of neurological diseases and disorders including Parkinson, depression, and other disorders (Johansen-Berg et al., 2008; Shils et al., 2008; Choi et al., 2015). Non-targeted electrostimulation using transcranial electrodes is also being widely used but remains controversial, and lacks precise clinical indications (Lefaucheur et al., 2014; Esmaeilpour et al., 2017; Huang et al., 2017; Lafon et al., 2017; Santos et al., 2017). Consideration is also

being given to future use of optogenetic stimulation therapies that would offer still greater precision compared to electrical stimulation – targeting not only a particular area but a particular cell type or set of cell types within that area (Vierling-Claassen et al., 2010; Kerr et al., 2014; Samineni et al., 2017). All such stimulation protocols require identification of suitable, if not necessarily optimal, ranges of parameters, e.g., strength, duration, frequency, waveform, location, iterations, schedule, reference activity, etc. It is not generally feasible to evaluate this large and high-dimensional stimulus parameter space empirically through experimentation. Hence, stimulation development would benefit from the extensive explorations possible using simulation of the response of micro- and macro-scale neural circuits in the brain, potentially in patient-specific fashion.

Successful application of models in the clinical domain will in future depend on the use of credible practices to develop, evaluate, document and disseminate models and simulations, using the principles outlined above. Nonetheless, clinical applications will also always need to be verified and validated by in a clinical before being utilized. At present, studies have been done in the absence of standards for computational neuroscience models. The current manuscript is designed in part to continue the discussion about the development of such standards. Meanwhile, clinically relevant studies have been performed with varying degrees of adherence to engineering best-practices. Here, we briefly discuss two such studies, noting adherence to such practices and where investigators may have fallen short.

Simulation of Multi-Target Pharmacology

We first consider a mechanistic multiscale model of multi-target pharmacotherapy for disorders of cortical hyperexcitability (Neymotin et al., 2016). The study assessed hyperexcitability in an exploratory manner. They did not identify a single clinical context. Rather, they left the study open for exploration of cortical activation in both dystonia and seizures.

The multiscale model included molecular, cellular, and network scales, containing 1715 compartmental model neurons with multiple ion channels and intracellular molecular dynamics. Data used to develop the model was taken from a large number of sources including different species, different preparations (slice, cell culture, *in vivo*, *ex vivo*), different age animals, different states, different conditions. None of the data was taken from the clinical disorders in question due to limitations of human experimentation. As the model lacked a description of the motor output, the simulations could not be systematically evaluated in the context of dystonia. Beta activation in the cortex was used as a surrogate biomarker to evaluate whether simulations could account for activity patterns relevant to dystonia. However, as with many brain diseases, there is no established, clinically validated biomarker for dystonia. Additionally, the model lacked representations of spinal cord or limb, as well as many pharmacological parameters, particularly with respect to the role of neuromodulators (known unknown), brain states (less known unknown) and metabolic parameters.

The simulations were able to reproduce the target patterns of heightened cortical activity. The corresponding pathological

parameter sets were identified by independent random variations in parameters. These simulations demonstrated degeneracy, meaning that there were many combinations of parameters that yielded the pathological syndrome. The primary result was that no individual parameter alteration could consistently distinguish between pathological and physiological dynamics. A support vector machine (SVM), a machine learning approach, separated the physiological from pathological patterns in different regions of high-dimensional space, suggesting multi-target routes from dystonic to physiological dynamics. This suggested the potential need for a multi-target drug-cocktail approach to intervening in dystonia.

Several aspects of best-practices were utilized in this study. Dissemination: The model was disseminated via publication and meeting presentations with the code made available via ModelDB resource (reference #189154). Documentation: Limited documentation was also made available at a level conforming to ModelDB requirements, consistent with practices accepted by the computational neuroscience community. Provenance: Due to the nature of the clinical domain, parameter provenance was partial; details of parameter sources were included in the paper. Replicability: Model replicability was tested by ModelDB curators, but was not tested directly by the manuscript reviewers. Reproducibility: There have not yet been any third-party studies reproducing the model. There are unexploited opportunities to compare the model with alternative implementations, for example by considering simpler modeling formalisms for single-neuron activity. The credibility of the simulations along with insights derived from the results would be enhanced by follow-on work that reproduces the simulations using similar or alternative implementations. Validation: The current lack of an adequate biomarker for dystonia limits the ability to validate this study in the future. Verification: The NEURON simulation platform was used in this study. It has been vetted both internally and in comparison to other simulators (Brette et al., 2007). Versioning: The Mercurial VCS was used to track parameter variations and match to corresponding simulations.

Personalized Medicine Simulation for Epilepsy

As a contrast to above examination of the credibility practices as applicable to a mechanistic multiscale model, we considered these issues in the context an individualized phenomenological model of seizure propagation by Proix et al. (2017), implemented using The Virtual Brain platform (Sanz-Leon et al., 2013). The study was aimed at demonstrating that patient-specific virtual brain models derived based on information from diffusion MRI technique have sufficient predictive power to improve diagnosis and surgery outcome in cases of drug-resistant epilepsy. Data from individual patient tractography and EEG was utilized to parameterize each individual model separately. The diffusion MRI-based connectivity observed between the parcellated brain regions in each individual was used to create patient-specific connectivity matrices that related distinct autonomous oscillators (“Epileptors”) at each brain region. The resultant patient-specific virtual brain model was evaluated for

its consistency in predicting seizure-like activity patterns in that patient.

Dissemination: The model and the results were disseminated as part of the published manuscript as well as through conference presentations and posters. **Documentation:** The platform is well documented. Internal documentation of individual models for use by neurosurgeons may be available but was not publically available, perhaps for reasons of patient confidentiality. **Provenance:** Provenance of connectivity data from individual patients was made clear. The Epileptor model is a phenomenological description of oscillatory patterns of activity in a bulk tissue (neural mass model); hence, there are no explicit parameters or variables that directly arise from specific molecular, cellular and metabolic pathways. **Replicability:** While the manuscript was peer-reviewed, it is not readily apparent if the model was tested directly in simulations by the manuscript reviewers. **Reproducibility:** Virtual Brain provides a particular dynamical formalism based on bulk activity. It would be interesting to see if alternative model formalisms that incorporate details at cellular scales would produce similar results. **Validation:** Alternative connectivity matrices and weightings were considered based on data from control subjects, shuffling the data, changing the weights while preserving the topology of the connectivity, and log-transformation. The authors demonstrated that prediction of seizure patterns was best when the patient-specific topology of the connectivity matrix was utilized. **Verification:** The study considered alternative models based on fast coupling, no time-scale separation, and a generic model that shows saddle-node bifurcation. Based on the simulations considering these alternative models, the authors concluded that weak coupling is necessary for the predictions on the recruited networks. The Virtual Brain platform has undergone considerable code testing over the years. **Versioning:** The platform is made available in a public repository using Git.

ACTIONABLE RECOMMENDATIONS AND CONCLUSIONS

The potential of modeling and simulation in clinical application and medical education are promising. However, this potential is mainly being tapped in the areas that are close to traditional engineering domains, such as CFD and stress analysis. For this reason areas of medicine that are related to blood flow, biomechanics and orthopedics have benefited most. By contrast, the brain has an idiosyncratic evolved set of mechanisms that are extremely difficult to reverse engineer and which draw on many areas of engineering, some of which have not been invented yet. Therefore, computational neuroscience remains primarily in the research domain, with only fragmented translations from computational neuroscience to clinical use or to medical education. As medical practice moves toward precision, and then personalized, healthcare, multiscale modeling will be necessary for simulating the individual patient's response to disease and treatment. To move toward this goal, we must cultivate credible modeling and simulation practices taken from traditional areas of engineering.

Model Configuration Management

Since many models will be built within the context of a simulation platform, we refer here to "Model configuration." However, all of these points apply *a fortiori* to models being built from scratch in a general-purpose programming language.

- (1) Use *version control*: Git or Mercurial (hg) are preferred. GitHub can be used to host projects (Dabbish et al., 2012). In shared projects, version control establishes who is responsible for which pieces of code.
- (2) Use an *electronic notebook* (e-notebook) with clear documentation of every stage of model development (including mistakes).
- (3) Include *provenance* of parameters in e-notebook and via version control – parameters may be changed due to new experimental data, and it is valuable to have a clear record of when and why the change was made and what the consequence was for the model.
- (4) Perform testing of model components: for example, demonstrate that the cell-types show proper firing characteristics before incorporating them into networks.
- (5) Later in the process, develop a test suite for further testing. Ideally, model testing should be performed at every version update. A test suite can be linked to standard testing frameworks to automate this testing. Commonly used frameworks include Travis CI (Continuous Integration), Circle CI, Jenkins, and AppVeyor.
- (6) Whenever possible, use reliable model development platforms such as NEST, BRIAN, NEURON, MOOSE, NENGO, PyNN, etc. This will increase the likelihood of accurate simulation and will enhance sharing. Similarly, model components should be taken from reliable databases of morphologies, channels and other components.
- (7) Later in the process, encourage other groups to compare simulation results on alternative platforms or with different implementations.

Verify and Validate Models

- (1) Simulation platform developers generally verify the adequacy of the numerical analysis. Some simulators offer alternative numerical solvers which can be tested to assess the qualitative similarity of results. For example, one problematic area is the handling of fixed and variable time steps for spike handling in networks (Lytton and Hines, 2005).
- (2) Verify algorithms you develop. For example, when developing a neuronal network, make sure that the network design is correct before moving to actual simulation.
- (3) Verify that the simulation is a reasonable implementation of the conceptual model, ideally by comparing a graphical output of basic phenomenology with target phenomena. It is tragically easy to move on to the analysis phase of the study without first looking at the raw output to make sure it is reasonable.
- (4) Validate based on data from a real-world system. In some cases, it may be important to distinguish simulation results

from the same model for different situations, e.g., *in vitro* slice vs. *in vivo* background activation.

- (5) Test robustness of a model to parameter changes to ascertain whether a particular result is only seen with a particular set of parameter choices.
- (6) Document and propagate V&V testing processes and findings to the community.

Although all of these steps are valuable, it is important to understand that higher model fidelity and V&V rigor do not automatically translate to higher credibility.

Share Models and Data

- (1) Submit models to shared databases such as ModelDB or via GitHub. Share widely – you can submit the same model to various databases, and submit components such as cells and ion channel definitions to component databases.
- (2) Document thoroughly – the Methods section of a paper will provide an overall gloss but, typically, will not provide sufficient detail about software design and use. Comment code. Provide a README on how to launch the simulation.
- (3) Disseminate – publish, of course, and present posters and talks to various audiences. In a multidisciplinary area such as computational neuroscience, the same model will say different things to different audiences – physiologists, anatomists, cognitive neuroscientists, neurologists, psychiatrists, other modelers.
- (4) Join communities via organizations such as Society for Neuroscience (SFN), Computational and Systems Neuroscience (CoSyne), Organization for Computational Neuroscience (CNS).
- (5) Obtain independent reviews of models. This is difficult and time-consuming but some grants are now providing funding explicitly to pay for consultants to review models (NIH, 2015).

Define Context of Use and Simulation Requirements

We distinguished above between *exploratory models*, done by an individual researcher in order to provide ideas and new hypotheses, and *context models*, purpose-built models for external users in a given environment, for example, clinical or medical education use. In the latter case, it is essential to be clear about who the users are and which usage patterns (sets of use cases) are to be targeted and which ones are to be excluded or to be left as part of longer-term goals. However, even exploratory models can benefit from these considerations, envisioning yourself, your team, and perhaps an experimental collaborator as users.

- (1) Identify the *users*. Even if you are the only user at the beginning of the project, you will be sharing the model later, so you may want to take account of other users who are not as familiar with your domain. For clinical use, a clinical assistant for epilepsy would be different for neurosurgeons vs. neurologists.

- (2) Identify the *context of use*. For example, will this model primarily be used to study dynamics, or will it be extended into information theory or will it be expected to perform a memory function, etc.
- (3) Identify *intended uses*. You may have one intended use for yourself as a modeler to generate new theoretical hypotheses and another for your experimental colleague. In the context of an educational application, an SBME for medical students will be very different than an application for residents or continuing medical education.
- (4) Attempt to identify *usage patterns* – it is often the case that underprepared users who have not read the manual use a program in unintended, and sometimes dangerous, ways. Platforms and programs can produce warnings when it detects that the user is trying to use unsuitable combinations of parameters, etc.

Translation of Computational Neuroscience Models for Clinical and Medical Education Use

In the preceding sections, we have given examples of particular clinical and educational/training scenarios that may be ripe for the introduction of simulation technology. Here we list both those already mentioned and others that have potential for future applications. This list is by no means complete.

- (1) Education: Integration of modeling into mannequins and online virtual patients to reproduce neurological deficits in SBME. Initial versions of this would not require mechanistic multiscale modeling but could be done with phenomenological modeling. Future versions would incorporate mechanistic modeling to also incorporate the time-course of signs and symptoms (dynamics at multiple timescales).
- (2) Training: Virtual reality simulators with haptic feedback for neurosurgery training.
- (3) Personalized patient simulations to decide on surgery vs. interventional radiology (coiling) for aneurysms.
- (4) Clinical decision making: Personalized patient simulations to decide on surgical approach for epilepsy surgery (Jirsa et al., 2017).
- (5) Simulation for seizure prediction in Epilepsy Monitoring Unit (EMU).
- (6) Personalized patient simulation to determine therapies for Parkinson disease to include combinations of surgical, electrical and pharmacological therapy (Grill and McIntyre, 2001; Hammond et al., 2007; Shils et al., 2008; Van Albada et al., 2009; Kerr et al., 2013; Holt and Netoff, 2017).
- (7) Head, brain and neural modeling for understanding effects of different kinds of electrical stimulation including transcranial stimulation (Esmailpour et al., 2017; Huang et al., 2017; Lafon et al., 2017).
- (8) Modeling vagal and peripheral nerve stimulation for treatment of systemic disorders (NIH SPARC program²).

²<https://commonfund.nih.gov/sparc>

- (9) Psychiatry: identifying the varying roles of disorder in dopaminergic, glutamatergic, inhibitory and other deficits in schizophrenia to develop new multi-target therapies (Lisman et al., 2010).
- (10) Neurorehabilitation (physiatry) Modeling the interface between neural and musculoskeletal models to treat spasticity or dystonia (van der Krogt et al., 2016).

AUTHOR CONTRIBUTIONS

LM and WL worked collaboratively to layout the contents of the manuscript, coordinate the team of co-authors, as well as edited the final draft of the manuscript. All of the co-authors drafted the specific section listed below beside their initials, as well as helped with the review and editing of the entire manuscript. LM: Abstract, Use of Simulation in Medical Education, and Actionable Recommendations and Conclusions, rewriting manuscript. AD: Software Aspects. AE and JK: Role of the Community. CH: Developing Credible Mechanism-Oriented Multiscale Models: Procedure and Process section. JM and MH: Verification and Validation (V&V). RV: Use of Modeling in Clinical Domains of Brain Disease. WL: Introduction, Reproducibility and Replicability, and

Actionable Recommendations and Conclusions, rewriting manuscript.

FUNDING

This research was supported by NIH R01GM104139, R01EB024573, and USAMRMC W81XWH-15-1-0232 (AE); NIH R01 GM107340, U54 EB020405, P2C HD065690; USAMRMC W81XWH-15-1-0232 (JK); NIBIB U01 EB023224, NHLBI U01 HL133360, NIH OT2 OD023848 (RV); and NIH R01-EB022903, U01-EB017695, R01-MH086638, NYS DOH01-C32250GG-3450000 (WL).

ACKNOWLEDGMENTS

The authors would like to thank all members of the Committee on Credible Practice of Modeling & Simulation in Healthcare who have been dedicated to the advancement of the Committee's mission, but were not able to directly contribute to the development of this manuscript. The authors would also like to extend their sincerest gratitude to the Interagency Modeling and Analysis Group (IMAG), and the Multiscale Modeling Consortium (MSM) for enabling the Committee's work.

REFERENCES

- Amunts, K., Ebell, C., Muller, J., Telefont, M., Knoll, A., and Lippert, T. (2017). The human brain project: creating a European research infrastructure to decode the human brain. *Neuron* 92, 574–581. doi: 10.1016/j.neuron.2016.10.046
- Ascoli, G. A., Donohue, D. E., and Halavi, M. (2007). NeuroMorpho.Org: a central resource for neuronal morphologies. *J. Neurosci.* 27, 9247–9251. doi: 10.1523/JNEUROSCI.2055-07.2007
- Babiker, M. H., Chong, B., Gonzalez, L. F., Cheema, S., and Frakes, D. H. (2013). Finite element modeling of embolic coil deployment: multifactor characterization of treatment effects on cerebral aneurysm hemodynamics. *J. Biomech.* 46, 2809–2816. doi: 10.1016/j.jbiomech.2013.08.021
- Barth, B. B., Henriquez, C. S., Grill, W. M., and Shen, X. (2017). Electrical stimulation of gut motility guided by an in silico model. *J. Neural Eng.* 14:066010. doi: 10.1088/1741-2552/aa86c8
- Bird, R. B., Stewart, W. E., and Lightfoot, E. N. (1960). *Transport Phenomena*, 1st Edn. Hoboken, NJ: John Wiley & Sons, 808.
- Bower, J. M., and Beeman, D. (2012). *The Book of GENESIS: Exploring Realistic Neural Models with the General NEural Simulation System*. Berlin: Springer Science & Business Media.
- Brette, R., Rudolph, M., Carnevale, T., Hines, M., Beeman, D., Bower, J. M., et al. (2007). Simulation of networks of spiking neurons: a review of tools and strategies. *J. Comput. Neurosci.* 23, 349–398. doi: 10.1007/s10827-007-0038-6
- Cain, N., Iyer, R., Koch, C., and Mihalas, S. (2016). The computational properties of a simplified cortical column model. *PLoS Comput. Biol.* 12:e1005045. doi: 10.1371/journal.pcbi.1005045
- Callahan, A., Anderson, K. D., Beattie, M. S., Bixby, J. L., Ferguson, A. R., Fouad, K., et al. (2017). Developing a data sharing community for spinal cord injury research. *Exp. Neurol.* 295, 135–143. doi: 10.1016/j.expneurol.2017.05.012
- Campo-Deaño, L., Oliveira, M. S. N., and Pinho, F. T. (2015). A review of computational hemodynamics in middle cerebral aneurysms and rheological models for blood flow. *Appl. Mech. Rev.* 67:030801. doi: 10.1115/1.4028946
- Cannon, R. C., Gleeson, P., Crook, S., Ganapathy, G., Marin, B., Piasini, E., et al. (2014). LEMS: a language for expressing complex biological models in concise and hierarchical form and its use in underpinning NeuroML 2. *Front. Neuroinform.* 8:79. doi: 10.3389/fninf.2014.00079
- Carnevale, T., and Hines, M. (2006). *The NEURON Book*. New York, NY: Cambridge University Press. doi: 10.1017/CBO9780511541612
- Carnevale, T., Majumdar, A., Sivagnanam, S., Yoshimoto, K., Astakhov, V., Bandrowski, A., et al. (2014). The neuroscience gateway portal: high performance computing made easy. *BMC Neurosci.* 15(Suppl. 1):101. doi: 10.1186/1471-2202-15-S1-P101
- Chitkara, R., Rajani, A. K., Oehlert, J. W., Lee, H. C., Epi, M. S., and Halamek, L. P. (2013). The accuracy of human senses in the detection of neonatal heart rate during standardized simulated resuscitation: implications for delivery of care, training and technology design. *Resuscitation* 84, 369–372. doi: 10.1016/j.resuscitation.2012.07.035
- Choi, K. S., Riva-Posse, P., Gross, R. E., and Mayberg, H. S. (2015). Mapping the 'Depression Switch' during intraoperative testing of subcallosal cingulate deep brain stimulation. *JAMA Neurol.* 72, 1252–1260. doi: 10.1001/jamaneurol.2015.2564
- Chung, B., and Cebal, J. R. (2015). CFD for evaluation and treatment planning of aneurysms: review of proposed clinical uses and their challenges. *Ann. Biomed. Eng.* 43, 122–138. doi: 10.1007/s10439-014-1093-6
- Crook, S. M., Davison, A. P., and Plesser, H. E. (2013). "Learning from the past: approaches for reproducibility in computational neuroscience," in *20 Years of Computational Neuroscience*, ed. J. M. Bower (New York, NY: Springer), 73–102.
- Dabbish, L., Stuart, C., Tsay, J., and Herbsleb, J. (2012). "Social coding in github: transparency and collaboration in an open software repository," in *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work CSCW '12*, (New York, NY: ACM), 1277–1286. doi: 10.1145/2145204.2145396
- Davison, A. P., Brüderle, D., Eppler, J., Kremkow, J., Müller, E., Pevcevi, D., et al. (2008). PyNN: a common interface for neuronal network simulators. *Front. Neuroinform.* 2:11. doi: 10.3389/neuro.11.011.2008
- Diesmann, M., Gewaltig, M. O., and Aertsen, A. (1999). Stable propagation of synchronous spiking in cortical neural networks. *Nature* 402, 529–533. doi: 10.1038/990101

- 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
- Drummond, C. (2009). “Replicability is not reproducibility: nor is it good science,” in *Proceedings of the Evaluation Methods for Machine Learning Workshop at the 26th ICML*, Montreal, QC.
- Dudani, N., Ray, S., George, S., and Bhalla, U. S. (2009). Multiscale modeling and interoperability in MOOSE. *BMC Neurosci.* 10(Suppl. 1):54. doi: 10.1186/1471-2202-10-S1-P54
- Dura-Bernal, S., Chadderdon, G. L., Neymotin, S. A., Francis, J. T., and Lytton, W. W. (2014). Towards a real-time interface between a biomimetic model of sensorimotor cortex and a robotic arm. *Patt. Recognit. Lett.* 36, 204–212. doi: 10.1016/j.patrec.2013.05.019
- Dura-Bernal, S., Majumdar, A., Neymotin, S. A., Sivagnanam, S., Francis, J. T., and Lytton, W. W. (2015). “A dynamic data-driven approach to closed-loop neuroprosthetics based on multiscale biomimetic brain models,” in *Proceedings of the IEEE International Conference on High Performance Computing 2015 Workshop: InfoSymbiotics/Dynamic Data Driven Applications Systems (DDDAS) for Smarter Systems*, (Bangalore: IEEE).
- Ermak, D. M., Bower, D. W., Wood, J., Sinz, E. H., and Kothari, M. J. (2013). Incorporating simulation technology into a neurology clerkship. *J. Am. Osteopath. Assoc.* 113, 628–635. doi: 10.7556/jaoa.2013.024
- Esmailpour, Z., Marangolo, P., Hampstead, B. M., Bestmann, S., Galletta, E., Knotkova, H., et al. (2017). Incomplete evidence that increasing current intensity of tDCS boosts outcomes. *Brain Stimul.* 11, 310–321. doi: 10.1016/j.brs.2017.12.002
- 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
- Fitzhugh, R. (1961). Impulses and physiological states in theoretical models of nerve membrane. *Biophys. J.* 1, 445–466. doi: 10.1016/S0006-3495(61)86902-6
- Fuerch, J. H., Yamada, N. K., Coelho, P. R., Lee, H. C., and Halamek, L. P. (2015). Impact of a novel decision support tool on adherence to neonatal resuscitation program algorithm. *Resuscitation* 88, 52–56. doi: 10.1016/j.resuscitation.2014.12.016
- Furlanello C., De Domenico, M., Jurman, G., and Bussola, N. (2017). Towards a scientific blockchain framework for reproducible data analysis. arXiv:1707.06552.
- Gewaltig, M.-O., and Diesmann, M. (2007). NEST (NEural Simulation Tool). *Schol. J.* 2:1430. doi: 10.4249/scholarpedia.1430
- Gleeson, P., Crook, S., Cannon, R. C., Hines, M. L., Billings, G. O., Farinella, M., et al. (2010). NeuroML: a language for describing data driven models of neurons and networks with a high degree of biological detail. *PLoS Comput. Biol.* 6:e1000815. doi: 10.1371/journal.pcbi.1000815
- Gleeson, P., Davison, A. P., Silver, R. A., and Ascoli, G. A. (2017). A commitment to open source in neuroscience. *Neuron* 96, 964–965. doi: 10.1016/j.neuron.2017.10.013
- Gleeson, P., Piasini, E., Crook, S., Cannon, R., Steuber, V., Jaeger, D., et al. (2012). The open source brain initiative: enabling collaborative modelling in computational neuroscience. *BMC Neurosci.* 13(Suppl. 1):7.
- Gleeson, P., Silver, A., and Cantarelli, M. (2015). “Open source brain,” in *Encyclopedia of Computational Neuroscience*, eds D. Jaeger and R. Jung (New York, NY: Springer), 2153–2156.
- Goodman, D., and Brette, R. (2008). Brian: a simulator for spiking neural networks in python. *Front. Neuroinform.* 2:5. doi: 10.3389/neuro.11.005.2008
- Grill, W. M., and McIntyre, C. C. (2001). Extracellular excitation of central neurons: implications for the mechanisms of deep brain stimulation. *Thalamus Relat. Syst.* 1, 269–277. doi: 10.1017/S1472928801000255
- Hammond, C., Bergman, H., and Brown, P. (2007). Pathological synchronization in Parkinson's disease: networks, models and treatments. *Trends Neurosci.* 30, 357–364. doi: 10.1016/j.tins.2007.05.004
- Hariharan, P., D'Souza, G. A., Horner, M., Morrison, T. M., Malinauskas, R. A., and Myers, M. R. (2017). Use of the FDA nozzle model to illustrate validation techniques in computational fluid dynamics (CFD) simulations. *PLoS One* 12:e0178749. doi: 10.1371/journal.pone.0178749
- 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
- Holt, A. B., and Netoff, T. I. (2017). Computational modeling to advance deep brain stimulation for the treatment of Parkinson's disease. *Drug Discov. Today Dis. Models* 19, 31–36. doi: 10.1016/j.ddmod.2017.02.006
- Huang, Y., Liu, A. A., Lafon, B., Friedman, D., Dayan, M., Wang, X., et al. (2017). Measurements and models of electric fields in the in vivo human brain during transcranial electric stimulation. *Elife* 6:e18834. doi: 10.7554/eLife.18834
- Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., et al. (2003). The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* 19, 524–531. doi: 10.1093/bioinformatics/btg015
- Hunt, A. C., Erdemir, A., MacGabhann, F., Lytton, W. W., Sander, E. A., Transtrum, M. K., et al. (2018). The spectrum of mechanism-oriented models for explanations of biological phenomena. arXiv:1801.04909.
- Huwendiek, S., De Leng, B. A., Kononowicz, A. A., Kunzmann, R., Muijtjens, A. M., Van Der Vleuten, C. P., et al. (2014). Exploring the validity and reliability of a questionnaire for evaluating virtual patient design with a special emphasis on fostering clinical reasoning. *Med. Teach.* doi: 10.3109/0142159X.2014.970622 [Epub ahead of print].
- Issenberg, S. B., Mcgaghie, W. C., Petrusa, E. R., Lee Gordon, D., and Scalese, R. J. (2005). Features and uses of high-fidelity medical simulations that lead to effective learning: a BEME systematic review. *Med. Teach.* 27, 10–28. doi: 10.1080/01421590500046924
- Jirsa, V. K., Proix, T., Perdikis, D., Woodman, M. M., Wang, H., Gonzalez-Martinez, J., et al. (2017). The virtual epileptic patient: individualized whole-brain models of epilepsy spread. *Neuroimage* 145, 377–388. doi: 10.1016/j.neuroimage.2016.04.049
- Johansen-Berg, H., Gutman, D. A., Behrens, T. E. J., Matthews, P. M., Rushworth, M. F. S., Katz, E., et al. (2008). Anatomical connectivity of the subgenual cingulate region targeted with deep brain stimulation for treatment-resistant depression. *Cereb. Cortex* 18, 1374–1383. doi: 10.1093/cercor/bhm167
- Jones, F., Passos-Neto, C. E., and Baghiroli, O. F. M. (2015). Simulation in medical education: brief history and methodology. *Princ. Pract. Clin. Res.* 1, 56–63.
- Kellner, C. H. (2012). *Brain Stimulation in Psychiatry: ECT, DBS, TMS and Other Modalities*. Cambridge: Cambridge University Press. doi: 10.1017/CBO9780511736216
- Kerr, C. C., O'Shea, D. J., Goo, W., Dura-Bernal, S., Francis, J. T., Diester, I., et al. (2014). Network-level effects of optogenetic stimulation in a computer model of macaque primary motor cortex. *BMC Neurosci.* 15(Suppl. 1):107. doi: 10.1186/1471-2202-15-S1-P107
- Kerr, C. C., Van Albada, S. J., Neymotin, S. A., Chadderdon, G. L., Robinson, P. A., and Lytton, W. W. (2013). Cortical information flow in Parkinson's disease: a composite network/field model. *Front. Comput. Neurosci.* 7:39. doi: 10.3389/fncom.2013.00039
- Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B. E., Bussonnier, M., Frederic, J., et al. (2016). “Jupyter notebooks: a publishing format for reproducible computational workflows,” in *20th International Conference on Electronic Publishing*, Amsterdam: IOS Press, 87–90.
- Konakondla, S., Fong, R., and Schirmer, C. M. (2017). Simulation training in neurosurgery: advances in education and practice. *Adv. Med. Educ. Pract.* 8, 465–473. doi: 10.2147/AMEP.S113565
- Lafon, B., Rahman, A., Bikson, M., and Parra, L. C. (2017). Direct current stimulation alters neuronal input/output function. *Brain Stimul.* 10, 36–45. doi: 10.1016/j.brs.2016.08.014
- Lefaucheur, J.-P., André-Obadia, N., Antal, A., Ayache, S. S., Baeken, C., Benninger, D. H., et al. (2014). Evidence-based guidelines on the therapeutic use of repetitive transcranial magnetic stimulation (rTMS). *Clin. Neurophysiol.* 125, 2150–2206. doi: 10.1016/j.clinph.2014.05.021
- Lisman, J. E., Pi, H. J., Zhang, Y., and Otmakhova, N. A. (2010). A thalamo-hippocampal-ventral tegmental area loop may produce the positive feedback that underlies the psychotic break in Schizophrenia. *Biol. Psychiatry* 68, 17–24. doi: 10.1016/j.biopsych.2010.04.007

- Lloyd, C. M., Lawson, J. R., Hunter, P. J., and Nielsen, P. F. (2008). The CellML model repository. *Bioinformatics* 24, 2122–2123. doi: 10.1093/bioinformatics/btn390
- London, M., Beeren, A. R. L., Häusser, M., and Latham, P. E. (2010). Sensitivity to perturbations in vivo implies high noise and suggests rate coding in cortex. *Nature* 466, 123–127. doi: 10.1038/nature09086
- Lupascu, C. A., Morabito, A., Merenda, E., Marinelli, S., Marchetti, C., Migliore, R., et al. (2016). A general procedure to study subcellular models of transsynaptic signaling at inhibitory synapses. *Front. Neuroinform.* 10:23. doi: 10.3389/fninf.2016.00023
- Lytton, W. W. (2008). Computer modelling of epilepsy. *Nat. Rev. Neurosci.* 9, 626–637. doi: 10.1038/nrn2416
- Lytton, W. W. (2017). Computers, causality and cure in epilepsy. *Brain* 140, 516–526. doi: 10.1093/brain/awx018
- Lytton, W. W., and Hines, M. L. (2005). Independent variable timestep integration of individual neurons for network simulations. *Neural Comput.* 17, 903–921. doi: 10.1162/0899766053429453
- Lytton, W. W., Seidenstein, A., Dura-Bernal, S., Schurmann, F., McDougal, R. A., and Hines, M. L. (2016). Simulation neurotechnologies for advancing brain research: parallelizing large networks in NEURON. *Neural Comput.* 28, 2063–2090. doi: 10.1162/NECO_a_00876
- McDougal, R. A., Bulanova, A. S., and Lytton, W. W. (2016a). Reproducibility in computational neuroscience models and simulations. *IEEE Trans. Biomed. Eng.* 63, 2021–2035. doi: 10.1109/TBME.2016.2539602
- McDougal, R. A., Morse, T. M., Carnevale, T., Marengo, L., Wang, R., Migliore, M., et al. (2016b). Twenty years of ModelDB and beyond: building essential modeling tools for the future of neuroscience. *J. Comput. Neurosci.* 42, 1–10. doi: 10.1007/s10827-016-0623-7
- Miceli, G., Cavallini, A., Santalucia, P., and Gensini, G. (2015). Simulation in neurology. *Neurol. Sci.* 36, 1967–1971. doi: 10.1007/s10072-015-2228-8
- Moore, J. W., and Stuart, A. (2011). *Neurons in Action 2*. Oxford: Oxford University Press.
- Moraru, I. I., Schaff, J. C., Slepchenko, B. M., Blinov, M. L., Morgan, F., Lakshminarayana, A., et al. (2008). Virtual cell modelling and simulation software environment. *IET Syst. Biol.* 2, 352–362. doi: 10.1049/iet-syb:20080102
- Mulugeta, L., and Erdemir, A. (2013). “Committee on credible practice of modeling and simulation in healthcare,” in *Proceedings of the ASME 2013 Conference on Frontiers in Medical Devices: Applications of Computer Modeling and Simulation, V001T10A015–V001T10A015*, (New York, NY: American Society of Mechanical Engineers). doi: 10.1115/FMD2013-16080
- Nadkarni, P., Mirsky, J., Skoufos, E., Healy, M., Hines, M., Miller, P., et al. (2002). “Senselab: modeling heterogenous data on the nervous system,” in *Bioinformatics: Databases and Systems*, ed. S. I. Letovsky (Boston, MA: Springer), 105–118. doi: 10.1007/0-306-46903-0_10
- Neymotin, S. A., Dura-Bernal, S., Lakatos, P., Sanger, T. D., and Lytton, W. W. (2016). Multitarget multiscale simulation for pharmacological treatment of dystonia in motor cortex. *Front. Pharmacol.* 7:157. doi: 10.3389/fphar.2016.00157
- NIH (2015). *PAR-15-085: Predictive Multiscale Models for Biomedical, Biological, Behavioral, Environmental and Clinical Research (U01)*.” Department of Health and Human Services. Available at: <http://grants.nih.gov/grants/guide/pa-files/PAR-15-085.html> [accessed January 8, 2015].
- Nunez, P. L., and Srinivasan, R. (2005). *Electric Fields of the Brain: The Neurophysics of EEG*, 2nd Edn. New York, NY: Oxford University Press.
- Oberkamp, W. L., and Roy, C. J. (2010). *Verification and Validation in Scientific Computing*. Cambridge: Cambridge University Press. doi: 10.1017/CBO9780511760396
- Peterson, B. E., Healy, M. D., Nadkarni, P. M., Miller, P. L., and Shepherd, G. M. (1996). ModelDB: an environment for running and storing computational models and their results applied to neuroscience. *J. Am. Med. Inform. Assoc.* 3, 389–398. doi: 10.1136/jamia.1996.97084512
- Plesser, H., Diesmann, M., Marc-Oliver, G., and Morrison, A. (2015). “NEST: the neural simulation tool,” in *Encyclopedia of Computational Neuroscience*, eds D. Jaeger and R. Jung (New York, NY: Springer), 1849–1852.
- Plesser, H. E. (2017). Reproducibility vs. Replicability: a brief history of a confused terminology. *Front. Neuroinform.* 11:76. doi: 10.3389/fninf.2017.00076
- Potjans, T. C., and Diesmann, M. (2014). The cell-type specific cortical microcircuit: relating structure and activity in a full-scale spiking network model. *Cereb. Cortex* 24, 785–806. doi: 10.1093/cercor/bhs358
- Proix, T., Bartolomei, F., Guye, M., and Jirsa, V. K. (2017). Individual brain structure and modeling predict seizure propagation. *Brain* 140, 651–654. doi: 10.1093/brain/awx004
- Rall, W. (1962). Electrophysiology of a dendritic neuron model. *Biophys. J.* 2, 145–167. doi: 10.1016/S0006-3495(62)86953-7
- Ranjan, R., Khazen, G., Gambazzi, L., Ramaswamy, S., Hill, S. L., Schürmann, F., et al. (2011). Channelpedia: an integrative and interactive database for ion channels. *Front. Neuroinform.* 5:36. doi: 10.3389/fninf.2011.00036
- Reyes, R. (2016). *An Empirical Evaluation of an Instrument to Determine the Relationship Between Second-Year Medical Students’ Perceptions of NERVE VP Design Effectiveness and Students’ Ability to Learn and Transfer Skills from NERVE*. Orlando, FL: University of Central Florida.
- Robinson, P. A., Rennie, C. J., and Rowe, D. L. (2002). Dynamics of large-scale brain activity in normal arousal states and epileptic seizures. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.* 65, 041924. doi: 10.1103/PhysRevE.65.041924
- Ross, S. E., Ouyang, Z., Rajagopalan, S., and Bruns, T. M. (2018). Evaluation of decoding algorithms for estimating bladder pressure from dorsal root ganglia neural recordings. *Ann. Biomed. Eng.* 46, 233–246. doi: 10.1007/s10439-017-1966-6
- Samineni, V. K., Mickle, A. D., Yoon, J., Grajales-Reyes, J. G., Pullen, M. Y., Crawford, K. E., et al. (2017). Optogenetic silencing of nociceptive primary afferents reduces evoked and ongoing bladder pain. *Sci. Rep.* 7:15865. doi: 10.1038/s41598-017-16129-3
- Santos, M. D. D., Cavenaghi, V. B., Mac-Kay, A. P. M. G., Serafim, V., Venturi, A., Truong, D. Q., et al. (2017). Non-invasive brain stimulation and computational models in post-stroke aphasic patients: single session of transcranial magnetic stimulation and transcranial direct current stimulation. A Randomized Clinical Trial. *Sao Paulo Med. J.* 135, 475–480. doi: 10.1590/1516-3180.2016.0194060617
- Sanz-Leon, P., Knock, S., Woodman, M., Domide, L., Mersmann, J., McIntosh, A., et al. (2013). The virtual brain: a simulator of primate brain network dynamics. *Front. Neuroinform.* 7:10. doi: 10.3389/fninf.2013.00010
- Schulte, E., and Davison, D. (2011). Active documents with org-mode. *Comput. Sci. Eng.* 13, 66–73. doi: 10.1109/MCSE.2011.41
- Seidenstein, A. H., Barone, F. C., and Lytton, W. W. (2015). Computer modeling of ischemic stroke. *Scholarpedia J.* 10:32015. doi: 10.4249/scholarpedia.32015
- 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
- Shils, J. L., Mei, L. Z., and Arle, J. E. (2008). Modeling parkinsonian circuitry and the DBS electrode. II. Evaluation of a computer simulation model of the basal ganglia with and without subthalamic nucleus stimulation. *Stereotact. Funct. Neurosurg.* 86, 16–29. doi: 10.1159/000108585
- Stanisic, L., Legrand, A., and Danjean, V. (2015). An effective git and org-mode based workflow for reproducible research. *ACM SIGOPS Operat. Syst. Rev.* 49, 61–70. doi: 10.1145/2723872.2723881
- Stern, F., Wilson, R., and Shao, J. (2006). Quantitative V&V of CFD simulations and certification of CFD codes. *Int. J. Numer. Methods Fluids* 50, 1335–1355. doi: 10.1002/fld.1090
- Szigeti, B., Gleeson, P., Vella, M., Khayrulin, S., Palyanov, A., Hokanson, J., et al. (2014). OpenWorm: an open-science approach to modeling caenorhabditis elegans. *Front. Comput. Neurosci.* 8:137. doi: 10.3389/fncom.2014.00137
- Tikidji-Hamburyan, R. A., Narayana, V., Bozkus, Z., and El-Ghawazi, T. A. (2017). Software for brain network simulations: a comparative study. *Front. Neuroinform.* 11:46. doi: 10.3389/fninf.2017.00046
- Van Albada, S. J., Gray, R. T., Drysdale, P. M., and Robinson, P. A. (2009). Mean-field modeling of the basal ganglia-thalamocortical system. II: dynamics of parkinsonian oscillations. *J. Theor. Biol.* 257, 664–688. doi: 10.1016/j.jtbi.2008.12.013
- van der Krogt, M. M., Bar-On, L., Kindt, T., Desloovere, K., and Harlaar, J. (2016). Neuro-musculoskeletal simulation of instrumented contracture and spasticity assessment in children with cerebral palsy. *J. Neuroeng. Rehabil.* 13:64. doi: 10.1186/s12984-016-0170-5

- Vaseghi, M., Salavatian, S., Rajendran, P. S., Yagishita, D., Woodward, W. R., Hamon, D., et al. (2017). Parasympathetic dysfunction and antiarrhythmic effect of vagal nerve stimulation following myocardial infarction. *JCI Insight* 2:e86715. doi: 10.1172/jci.insight.86715
- Vierling-Claassen, D., Cardin, J. A., Moore, C. I., and Jones, S. R. (2010). Computational modeling of distinct neocortical oscillations driven by cell-type selective optogenetic drive: separable resonant circuits controlled by low-threshold spiking and fast-spiking interneurons. *Front. Hum. Neurosci.* 4:198. doi: 10.3389/fnhum.2010.00198
- Weiss, S., and Mueller, H. M. (2012). Too many betas do not spoil the broth: the role of beta brain oscillations in language processing. *Front. Psychol.* 3:201. doi: 10.3389/fpsyg.2012.00201
- Wilson, C. (2008). Up and down states. *Scholarpedia J.* 3:1410. doi: 10.4249/scholarpedia.1410
- Wrigley, S. (ed.). (2013). "Negative training: when the simulator lies," in *Why Planes Crash Case Files: 2001*. Mannheim: Fear of Landing.
- Yamada, N. K., Fuerch, J. H., and Halamek, L. P. (2017). Simulation-based patient-specific multidisciplinary team training in preparation for the resuscitation and stabilization of conjoined twins. *Am. J. Perinatol.* 34, 621–626. doi: 10.1055/s-0036-1593808
- Zhang, C., Bakshi, A., and Prasanna, V. K. (2007). "ModelML: a markup language for automatic model synthesis," in *Proceedings of the IEEE International Conference on Information Reuse and Integration 2007* (Las Vegas, IL: IEEE), doi: 10.1109/iri.2007.4296640

Conflict of Interest Statement: LM owns and operates InSilico Labs LLC and Medalist Fitness LLC. AE owns and operates Innodof, LLC. MH is employed by ANSYS, Inc.

The other 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 © 2018 Mulugeta, Drach, Erdemir, Hunt, Horner, Ku, Myers, Vadigepalli and Lytton. 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 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.