



OPEN ACCESS

EDITED AND REVIEWED BY

Marcel Filoche,
École Supérieure de Physique et de
Chimie Industrielles de la Ville de Paris,
France

*CORRESPONDENCE

Are Magnus Bruaset,
✉ arem@simula.no

RECEIVED 12 October 2023

ACCEPTED 23 October 2023

PUBLISHED 01 November 2023

CITATION

Bruaset AM, Cai X, Desprez F and
Parsons M (2023), Editorial:
Heterogeneous computing in physics-
based models.
Front. Phys. 11:1320450.
doi: 10.3389/fphy.2023.1320450

COPYRIGHT

© 2023 Bruaset, Cai, Desprez and
Parsons. This is an open-access article
distributed under the terms of the
[Creative Commons Attribution License
\(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or
reproduction in other forums is
permitted, provided the original author(s)
and the copyright owner(s) are credited
and that the original publication in this
journal is cited, in accordance with
accepted academic practice. No use,
distribution or reproduction is permitted
which does not comply with these terms.

Editorial: Heterogeneous computing in physics-based models

Are Magnus Bruaset^{1,2*}, Xing Cai^{1,2}, Frédéric Desprez³ and Mark Parsons⁴

¹Simula Research Laboratory, Oslo, Norway, ²Department of Informatics, University of Oslo, Oslo, Norway, ³Inria at University Grenoble Alpes, Montbonnot-Saint-Martin, France, ⁴EPCC, University of Edinburgh, Edinburgh, United Kingdom

KEYWORDS

heterogeneous computing, physics-based simulation, general-purpose processor, domain-specific accelerator, quantum processor, neural network

Editorial on the Research Topic Heterogeneous computing in physics-based models

Since Nvidia's general-purpose graphics processing units (GPUs) were adopted by technical computing at the beginning of this century, the term *heterogeneous computing* for a long while has been synonymous with GPU computing. Such a simplistic characterization of heterogeneous computing is certainly no longer correct. This Research Topic thus aims to provide a glimpse of some new developments in three aspects of this subject: hardware, software, as well as heterogeneity in computational methodology. To limit the scope, we restrict our attention to heterogeneous computing used in large-scale simulations of systems governed by the laws of physics. Arguably, physics-based simulations are no longer fueling hardware innovation; that role is replaced by the growing data deluge and the advances in data analytics, machine learning (ML), and artificial intelligence (AI). We believe, however, that physics-based simulations are still at the core of scientific and technological advances, thus deserving innovative research effort that can deliver the full potential of heterogeneous computing and also benefit other research domains.

Hardware heterogeneity has historically received most of the attention. GPUs, based on a different architectural design principle than conventional CPUs, have helped to postpone the end of Moore's law. Lately, the demand for more computing power and higher energy efficiency has led to a renaissance of more exotic hardware architectures, as observed in [1,2]. Many of the new architectures can be categorized as domain-specific accelerators, with the wafer-scale engine (WSE) of [3], the intelligence processing unit (IPU) of [4], and the Gaudi processor of [5] as the best known ML accelerators. Another fundamentally different hardware strategy has led to the rise of quantum hardware, most notably the quantum processor developments from [6,7], respectively. Although these ML accelerators and quantum processors are not designed to solve differential equations, we present here two examples of re-purposing such domain-specific hardware for physics-based simulations. In particular, Burchard et al. use Graphcore IPUs to solve the monodomain model of cardiac electrophysiology that is mathematically described by a non-linear reaction-diffusion system, whereas Markidis adopts a simulated quantum processor in physics-informed neural networks to speed up solving the Poisson problems with quadratic and sinusoidal sources. Hardware diversity is also reflected in the other two papers of this Research Topic.

Specifically, Brodtkorb and Sætra use multiple GPUs to solve the Euler equations, whereas Nordhagen et al. use CPU clusters for simulating interacting particles by variational Monte Carlo methods.

Software heterogeneity accompanies hardware heterogeneity, because new hardware architectures typically require completely new programming strategies. For GPU computing, although CUDA is a mature programming model, the low-level programming that is required is still too demanding for many domain scientists. Lifting the level of GPU programming is thus desirable. Brodtkorb and Sætra present a high-level programming framework for implementing a Euler equation solver in the Python programming language. Except for the core numerical computation that is implemented in CUDA C++ (and compiled just-in-time), the entire code is written compactly in Python. Specifically, new Python classes are developed for managing the subgrids and individual MPI processes (each controlling a GPU). Moreover, the existing Python modules PyCUDA and mpi4py further alleviate the programming challenges associated with parallelization. In the same spirit, Markidis uses high-level programming interfaces, in particular, TensorFlow Quantum [8] and Strawberry Fields [9], to abstract the specifics of quantum hardware, thus avoiding explicit management of data movement or offloading of the quantum circuit execution. At the other end of the programming spectrum, Burchard et al. are forced to use the low-level Poplar C++ library that is specific for Graphcore IPUs. In general, new research is needed to overcome the programming hurdles that are often associated with heterogeneous computing.

Heterogeneity has also found its way into computational methodology, where we are referring to the use of ML to improve physics-based simulations. As illustrated by the papers of Markidis and Nordhagen et al., it is possible to adopt neural networks to speed up such simulations. Specifically, Markidis uses a (simulated) quantum processor to run a surrogate neural network as part of a physics-informed ML framework; Nordhagen et al. use the Gaussian-binary restricted Boltzmann machine, which is a shallow neural network, to replace the trial wave function of a quantum mechanics system. There are good reasons to expect more combinations of ML and physics-based simulation in future.

Although the number of papers included in this Research Topic is small, they nevertheless showcase some of the ongoing research

work on various aspects of the now vibrant subject of heterogeneous computing. We expect that continued research effort in this content-rich subject will enable further developments in physics-based simulations, through a combined use of hybrid computing strategies (physics-based and ML-assisted), hybrid software (high-level and low-level), and hybrid hardware (general-purpose, reconfigurable, and domain-specific).

Author contributions

AB: Conceptualization, Writing–review and editing. XC: Conceptualization, Writing–original draft, Writing–review and editing. FD: Conceptualization, Writing–review and editing. MP: Conceptualization, Writing–review and editing.

Acknowledgments

We would like to express our gratitude to the reviewers for their careful reading and constructive comments. We also thank the authors of the papers that are included in this Research Topic for their scientific contributions to this important and forward-looking subject.

Conflict of interest

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

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Hennessy JL, Patterson DA. A new golden age for computer architecture. *Commun ACM* (2019) 62:48–60. doi:10.1145/3282307
- Leiserson CE, Thompson NC, Emer JS, Kuszmaul BC, Lampon BW, Sanchez D, et al. There's plenty of room at the top: what will drive computer performance after Moore's law? *Science* (2020) 368:eaam9744. doi:10.1126/science.aam9744
- Cerebras. The future of AI is here (2022). Available from: <https://www.cerebras.net/product-chip/> (Accessed October, 2023).
- Graphcore. Designed for AI: intelligence processing unit (2021). Available from: <https://www.graphcore.ai/products/ipu> (Accessed October, 2023).
- Habana. GAUDI: first generation deep learning training & inference processor (2023). Available from: <https://habana.ai/products/gaudi/> (Accessed October, 2023).
- IBM. The IBM quantum development roadmap (2023). Available from: <https://www.ibm.com/quantum/roadmap> (Accessed October, 2023).
- Google Quantum AI. Quantum computing hardware (2019). Available from: <https://quantumai.google/hardware> (Accessed October, 2023).
- Broughton M, Verdon G, McCourt T, Martinez AJ, Yoo JH, Isakov SV, et al. TensorFlow Quantum: a software framework for quantum machine learning (2021). *arXiv*. doi:10.48550/arXiv.2003.02989
- Killoran N, Izaac J, Quesada N, Bergholm V, Amy M, Weedbrook C. Strawberry Fields: a software platform for photonic quantum computing. *Quantum* (2019) 3:129. doi:10.22331/q-2019-03-11-129