

# Neuromorphic Computing for Scientific Applications

Robert Patton

*Computer Science and Math Division*  
*Oak Ridge National Laboratory*  
Oak Ridge, TN, USA  
pattonrm@ornl.gov

Prasanna Date

*Computer Science and Math Division*  
*Oak Ridge National Laboratory*  
Oak Ridge, TN, USA  
datepa@ornl.gov

Shruti Kulkarni

*Computer Science and Math Division*  
*Oak Ridge National Laboratory*  
Oak Ridge, TN, USA  
kulkarnisr@ornl.gov

Chathika Gunaratne

*Computer Science and Math Division*  
*Oak Ridge National Laboratory*  
Oak Ridge, TN, USA  
gunaratnecs@ornl.gov

Seung-Hwan Lim

*Computer Science and Math Division*  
*Oak Ridge National Laboratory*  
Oak Ridge, TN, USA  
lims1@ornl.gov

Guojing Cong

*Computer Science and Math Division*  
*Oak Ridge National Laboratory*  
Oak Ridge, TN, USA  
congg@ornl.gov

Steven R. Young

*Computer Science and Math Division*  
*Oak Ridge National Laboratory*  
Oak Ridge, TN, USA  
youngsr@ornl.gov

Mark Coletti

*Computer Science and Math Division*  
*Oak Ridge National Laboratory*  
Oak Ridge, TN, USA  
colettima@ornl.gov

Thomas E. Potok

*Computer Science and Math Division*  
*Oak Ridge National Laboratory*  
Oak Ridge, TN, USA  
potokte@ornl.gov

Catherine D. Schuman

*Department of EECS*  
*University of Tennessee*  
Knoxville, TN, USA  
cschuman@utk.edu

**Abstract**—Neuromorphic computing technology continues to make strides in the development of new algorithms, devices, and materials. In addition, applications have begun to emerge where neuromorphic computing shows promising results. However, numerous barriers to further development and application remain. In this work, we identify several science areas where neuromorphic computing can either make an immediate impact (within 1 to 3 years) or the societal impact would be extremely high if the technological barriers can be addressed. We identify both opportunities and hurdles to the development of neuromorphic computing technology for these areas. Finally, we discuss future directions that need to be addressed to expand both the development and application of neuromorphic computing.

**Index Terms**—neuromorphic computing, spiking neural networks, neural simulation, scientific applications

Notice: This manuscript has been authored in part by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (<http://energy.gov/downloads/doe-public-access-plan>).

## I. INTRODUCTION

The revolution of Artificial Intelligence (AI) in scientific research continues to expand as the AI techniques and available software grow and increase in both capability and usability. There are several drivers behind this rapid adoption of AI technology. First, neural models (typically, convolutional neural networks (CNNs)) have become more advanced in design and performance capability. These models are capable of performing on data and with accuracy levels not previously achievable and well beyond human capability. Second, these models are openly shared, accessible, and often just require retraining on application specific data in order to be operational. For instance, models developed and trained on ImageNet [1] have been simply retrained on application specific images and then used operationally for that application. This reduces the AI expertise needed by a domain scientist looking to apply these models, thereby enhancing adoption. Third, CNNs benefit from a robust software stack that enables domain scientists to simply execute these models without much concern for the underlying hardware needed to execute the model. In fact, the CNN software stack has been used on computing hardware ranging from laptops to supercomputers. Fourth,

the underlying hardware to execute these models has rapidly advanced and have become widely available. This hardware availability further enhances adoption of CNN-based methods. Finally, there is extensive educational support via numerous online content ranging from blogs, tutorials, videos, and industry supported training. Much of this content is free and readily available twenty four hours a day. All of these factors combined have resulted in very fast, wide ranging adoption of CNN-based AI technology in scientific research.

Despite these numerous benefits, there remain several areas of scientific research where CNNs and their corresponding software and hardware stacks are prohibitive, or where a neuromorphic computing approach would provide significantly more benefits such as very low size, weight, and power (SWaP) computing. Neuromorphic computers are implemented on non-von Neumann architecture devices and compute in an event-driven manner with the use of Spiking Neural Networks (SNNs) [2].

The following sections will describe a sample of these potential science areas where neuromorphic computing can have a high positive impact in achieving new scientific discoveries or achievements. Finally, we describe areas where the neuromorphic community can reduce the barriers of entry and increase the adoption of neuromorphic computing in scientific research.

## II. CONSTRAINTS & OPPORTUNITIES

Most scientific applications involve processing huge amounts of data, which demand High Performance Computing (HPC) resources. However, with the ever growing volumes of data generated by these applications, it becomes more imperative that the various aspects of neuromorphic computing for processing the incoming high bandwidth data at a low power be explored. By utilizing non-von Neumann architecture, neuromorphic hardware can better co-locate memory and processing units, thereby enhancing parallelism and potentially reducing the overall memory access power, which is quickly becoming the major bottleneck in increasing the throughput of processors [3]. There have been several neuromorphic hardware demonstrated to date, which include both digital and mixed signal processors [4]. BrainScales 1 and 2, and DYNAPs are mixed signal neuromorphic hardware with neurons and synapses implemented as analog circuits [5]. While analog hardware platforms have lower power consumption, and can support various biological time constant through circuit and device dynamics, but they suffer from device mismatches [4].

Neuromorphic computers are event-driven by nature, and are inherently scalable. Furthermore, on machine learning tasks, they consume orders of magnitude less power as compared to CPUs and GPUs, without compromising on computation time [6]. These characteristics lend them to event-based sensing applications that require low latency processing of signal information in real time [7]. Neuromorphic computing is envisioned to have the most impact in applications that require low size, weight and power (SWaP). Some examples of such applications include autonomous systems (for e.g.,

self-driving cars, drones, automated guided vehicles etc.), embedded systems (for e.g., control circuits, signal processing, power electronics etc.), Internet of Things (for e.g., smart automation), and remote sensing (for e.g., high energy physics). Neuromorphic computers also lend themselves to modeling and simulation applications in computational sciences such as neuroscience and epidemiology. In computing, neuromorphic processors are seen as low power machine learning accelerators in the near future. We shed some light on the utility of neuromorphic computing in many such scientific disciplines in Section III.

## III. HIGH IMPACT SCIENCE AREAS

In this section, we identify several science areas that are fertile for the development and application of neuromorphic computing. This list is not intended to be exhaustive, but intended to highlight areas where neuromorphic computing can either make an immediate impact (1 to 3 years out) or the societal impact would be extremely high. In some areas, there are either no or very few barriers to using neuromorphic computing. In other areas, however, there are significant barriers that remain but the potential societal impact is very high if those barriers can be addressed.

A graphic overview of this work is shown in Figure 1. We have grouped the science areas into "Modeling & Simulation", which involves simulating a physical phenomena to better understand it, or "Data Analytics", which involves the processing of data in order to classify or predict physical events primarily with timing information. Furthermore, we identify several "non-neuromorphic" computer science areas (shown along the bottom of the figure) that can both contribute to the development of neuromorphic computing as well as benefit from the use of neuromorphic computing in that area (i.e., mutually beneficial co-development). Development of these computer science areas would also enable neuromorphic computing to expand into additional science areas not discussed here.

### A. *Simulating Neuromorphic Hardware*

Simulating neuromorphic systems is important to understand their performance and behavior to better inform the architectural design decisions in the future editions of these systems. Current efforts to simulate neuromorphic systems on CPUs and GPUs have several limitations such as poor scalability, restricted size of SNNs and undesirably long running times. In order to simulate neuromorphic systems effectively, it is important to have a simulation engine that can comply with intrinsic neuromorphic characteristics such as co-located processing and memory, inherent scalability and event-driven computation. We postulate that current neuromorphic computers might be the ideal simulation platform for simulating the next generation of neuromorphic computers. Such approaches are currently used in HPC, where current HPC systems are used to model the performance of future HPC systems [8].

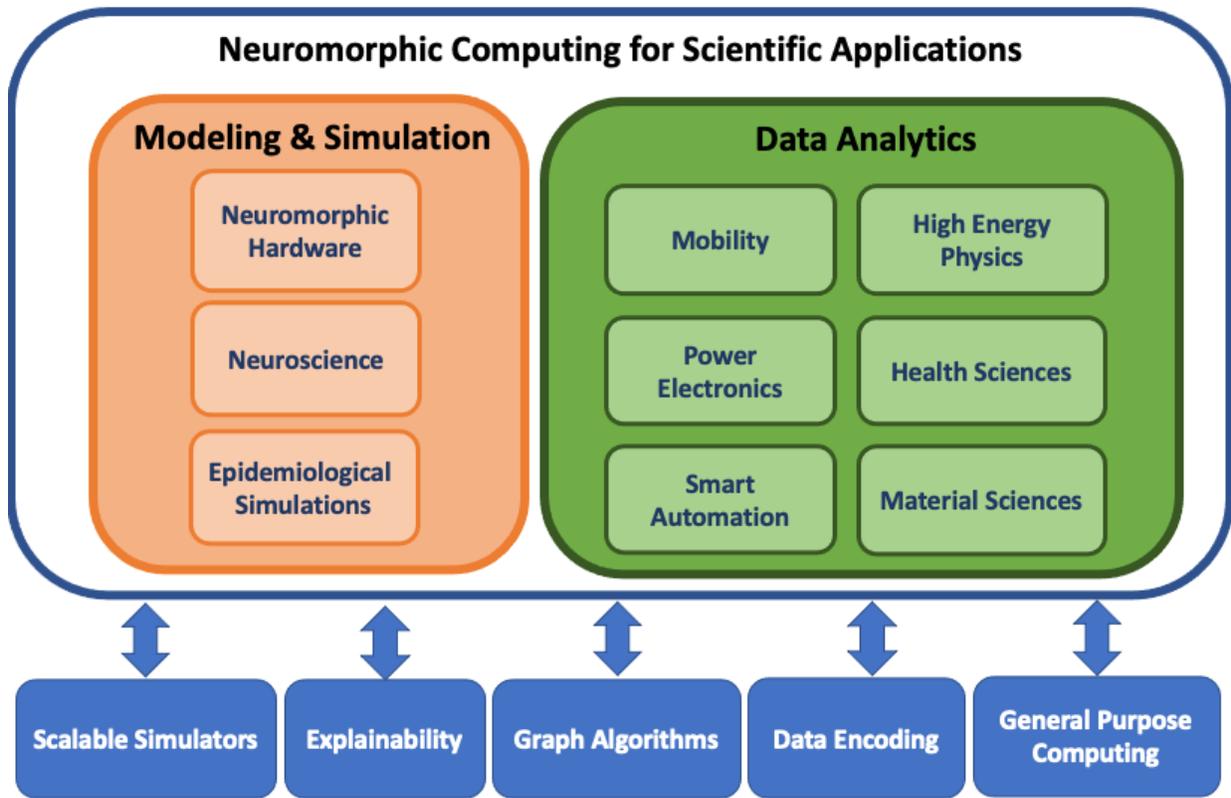


Fig. 1. Science areas that are prime candidates for the application of neuromorphic computing.

### B. Neuroscience

Computational neuroscience modeling was one of the earliest scientific applications of neuromorphic computing, with the goal of understanding the different brain regions [9]. Abstract models of spiking neurons and synapses have provided a seamless way to model the electrical activity in the brain regions [10]. Digital neuromorphic hardware such as SpiNNaker, and GPU-enabled simulation framework - GeNN, have demonstrated simulations of cortical microcircuit with over 70,000 neurons [11]. Several neurological dynamics such as neurogenesis in dentate gyrus, understanding episodic memories, and different neuroscience models of cognition are being studied by modeling and simulating them as SNNs [12]. This also brings several opportunities to develop novel predictive models for other AI applications [13].

### C. Epidemiological Simulations

Epidemiological models that can run in real-time, are adaptive, robust, and efficient, allow us to effectively plan for pandemics such as COVID-19, Ebola and Swine Flu. Such epidemiological models also inform critical decisions taken at the local, state and federal governments. Over the last couple of years, the COVID-19 pandemic has exposed many vulnerabilities in the way we perform epidemiological simulations. These have affected the policy decisions and in turn revealed shortcomings in our healthcare system,

medical science and supply chain operations. Neuromorphic computing-based approaches might hold the key to perform adaptive, robust and efficient (in terms of time, space and energy) epidemiological simulations in the future. Hamilton et al. have successfully demonstrated a spiking approach to Susceptible-Infected-Recovered (SIR) model, which is a standard model used in epidemiology [14]. There is a striking similarity between the individual-scale dynamics of complex network contagion models, such as SIR models, and the spike-threshold-fire mechanisms of SNNs. Further research along this line can enable us to perform epidemiological simulations in a much efficient manner, accounting for larger populations and simulating a wider range of contagion parameters. This has the potential to inform policy makers in taking critical decisions during global pandemics.

### D. Mobility

Mobility has recently experienced new growth in electrification and autonomy, with predicted growth to continue for decades to come. It is also anticipated that 5G cellular communications will play a key role in mobility. Currently, CNN-based approaches are widely used, primarily for camera sensors both on the vehicle and on infrastructure such as a traffic lights. However, vehicles currently have a limited electrical capacity, which is primarily used for vehicle movement. As a result, this electrical capacity does not support a power hungry

computing environment. Furthermore, autonomous navigation is expected to rely on multiple sensors (e.g., cameras, lidar, radar, cellular communications) with each type of sensor requiring processing and analyzing as well as fusion with other sensors. Consequently, a CNN-based approach would currently require power hungry devices to operate. This application space lends itself naturally to the very low SWaP of neuromorphic computing. In addition, there are existing data sets and simulators readily available or attainable for training. Preliminary results have demonstrated neuromorphic computing for fully electrified autonomous driving in a small scaled, controlled environment [15]. However, this is just the beginning of the potential for neuromorphic computing in mobility. Additional areas could extend to mobility infrastructure and communications.

In addition to autonomy and control of vehicles, neuromorphic computing can play a role even with internal combustion (IC) engines. Even though electrification is growing significantly, the use of IC engines is expected to continue for some time still. Preliminary work [16] [17] has demonstrated the use of neuromorphic computing to actively monitor the combustion process within an engine, and then control the air and fuel mixture in order to improve the combustion efficiency and measurably reduce fuel consumption. Additional research is continuing in this area.

#### *E. Power Electronics*

The field of power electronics has been using ANNs and recurrent neural networks for modeling the dynamics of the electrical state of the control system [18]. One such demonstration of using SNN for state prediction, was in solar power plant controller, to monitor and predict the next states of the system for anomalies [19]. While in this study, the performance of SNNs compared to the other AI models such as Decision Trees and non-linear auto regressive exogenous (NARX) models was lower due to the SNNs being employed as classifiers, more study is needed where SNNs can be deployed for tasks such as regression [20]. The inherent temporal characteristics of SNNs make them a suitable candidate for modeling dynamical systems and hence, the field of power electronics presents a great opportunity to suitably make use of these characteristics of SNNs.

#### *F. Smart Automation*

As the Internet-of-Things (IoT) continues to grow and mature, smart home automation as well as smart manufacturing holds significant potential for increased efficiency in time and energy, ultimately resulting in reduced costs. Most CNN-based approaches are typically applied to cameras that may be mounted to observe people or equipment operation. Some convolutional long short-term memory (LSTM) models are used for time series prediction. However, there remains a significant opportunity to bring neuromorphic computing to a wide range of sensors to perform sensor fusion and event detection or prediction. Furthermore, the advantages of neuromorphic computing's low SWaP could further increase

the potential for time and energy efficiency by deploying devices much closer to the sensor locations. In addition, SNN training and development based on evolutionary optimization is uniquely suited toward reinforcement learning and control applications.

#### *G. High Energy Physics*

The advancements in sensing and detector technologies have greatly benefited high energy physics experimentation techniques and thereby, aiding the theories being formulated. Machine learning techniques have also been increasingly deployed in order to intelligently process the high volume of data being generated from these experiments [21]–[23]. A major constraint for these data processing modules is the ability to process the data at very low latency and in a power-efficient manner, in a noisy environment. A recent work has demonstrated Application Specific Integrated Chip (ASIC) that implements a neural network autoencoder for performing data compression on the front-end detector of the High-Luminosity Large Hadron Collider (HL-LHC), with over 2,000 parameters for configuration, and consuming 95 mW of power. Deep Learning CNN models have been applied for identifying the vertex of particle interactions in neutrino scattering experiments [22], [23]. In the same application domain, SNNs trained with EONS were used to classify the segments of the detected neutrino particles with just about 90 neurons and 86 synapses, and operating with less than  $2\mu\text{J}$  per classification when realized on a memristive neuromorphic hardware [24].

#### *H. Health Sciences*

In health sciences, much of the CNN-based approaches drive toward image analysis such as pathology and radiology [25]. This is a natural application for those techniques in that there is no need for real-time processing and the computing environment does not need to be mobile. In addition, there are multiple data sets available for training and development of CNN architectures. However, there remains opportunities within health sciences where mobility, real-time processing, and low SWaP requirements would enable enhanced patient care, thanks to the portability and low power demand of neuromorphic devices. For example, neuromorphic devices could be used to monitor multiple sensors of patient vitals to identify trends or predictive patterns that may not currently be visible. Such a device could be embedded within a hospital bed, as a wrist band, or even as part of a smart watch. However, a significant barrier to this application is the lack of data sets or simulators for training and development of SNNs.

#### *I. Material Sciences*

With theoretic modeling, experimentation, and machine learning, the design of materials for desired target properties such as gas adsorption, conductivity, binding affinity, and others, becomes possible. Machine learning methods, including surrogate model design, active learning, and generative models, are playing an increasingly important role in materials

research [26]. Graph neural networks (GNNs) receive much recent attention in material science as they are a natural representation of the structures of materials [27]. For example, meta-organic framework (MOFs), alloys, nanoporous materials, and drugs, can all be represented as graphs.

In contrast to convolutional neural networks (CNNs), GNNs employ graph convolution, where the feature for a node is computed with all features of its neighboring nodes in the graph. Intuitively, spiking neural networks (SNNs) on neuromorphic hardware lend a natural implementation to the computation in GNNs. Different from the imperative mode of computation on CPUs and GPUs, computation on SNNs is more of a data-flow nature and may enjoy more parallelism. Although research in this area is in its nascent stage, we expect SNNs with neuromorphic computing to bring new capabilities to learning on graph structures for materials research.

The number of nodes in graphs for material structures range between a few hundreds to several thousands. The graphs are oftentimes highly regular and sparse. Such graphs can be easily implemented using neuromorphic hardware. Convolutions on such graphs are naturally recurrent, and may not suffer from over-smoothing [28] that plagues deep GNNs on CPUs and GPUs. Moreover, the relative small graph sizes in materials research can also make neuromorphic implementations fast and energy-efficient.

#### IV. OUTLOOK

Simulators are a vital component of the design, development, and deployment cycle of neuromorphic computing solutions. As pointed out in a survey of several existing neuromorphic computing simulators [29], performance, scalability, and flexibility are of utmost importance when selecting a suitable neuromorphic simulator. While most existing simulators lean toward computational neuroscience problems and not easily deployable in machine learning type problems, there are a few simulators such as NEST/PyNest [30], [31], Brain2 [32], [33], Brain2GeNN [33], [34], Nengo [35], and BindsNET [36] that have been applied to machine learning problems. Although most of these tools were originally designed for computational neuroscience models, BindsNET in particular has been developed for machine learning and reinforcement learning tasks using neuromorphic computing. However, the findings of [29] show that there is still no single simulator that is a best fit for all neuromorphic computing problems under varying demands in speed, scalability in network size, scalability with increasing compute resources, and flexibility. Frameworks suffer from at least one of a set of drawbacks including a lack of flexibility in the the backend deployment architecture (single CPU, GPU, or multi-node), and limitations in neuron and synapse behavior required by particular SNN implementations.

We recognize the need for such a neuromorphic simulator that will provide the flexibility to work across multiple architectures including multi-CPU, GPU, and multi-node computing, as recommended in [29]. Such a simulator should be focused on facilitating spiking neural network solutions

for machine learning and reinforcement learning tasks and support fundamental spiking neural network mechanisms such as synapse delays and weights, and neuronal thresholding, leak, refractory state, and axonal delays. Of particular interest is the close similarity between certain agent-based modeling applications and SNN mechanisms. Specifically, diffusion of information models [37] and SIR-type epidemiology models [38], [39] make use of a graph representation and spike-threshold-fire mechanisms that are highly similar to those of spiking neural networks. Agent-based models have been successfully implemented to use GPU [40], [41] and HPC [42] architectures to simulate large populations at-scale and it is likely that these framework could be extended to provide a basis for neuromorphic simulator development.

A neural network model promises a universal approximated function without laborious feature engineering and complete problem formulation. Despite its huge success in many learning tasks, the advantages of neural network models come at a price: the inability to interpret and understand the model behavior, creating a fundamental barrier to optimize and evaluate the model. Due to this barrier, it is an open research question to systematically design/diagnose a novel neural network model for the target learning objective. Moreover, neural network model families for neuromorphic computing such as Spiking Neural Network (SNN) exhibit a higher degree of freedom and complexity than other neural network models such as Convolutional Neural Network (CNN). For instance, SNN has temporal aspects in the connection between neurons and randomly connecting neurons rather than chain-like topology, common in CNN models. Thus, the understanding of neuromorphic models imposes additional challenges than other neural network models.

Traditional applications of neuromorphic computing have been in modeling and simulations or data analytics as shown in Figure 1. While these applications have demonstrated the efficacy of neuromorphic computing, we believe neuromorphic computing is capable of much more. For instance, neuromorphic computing is shown to be Turing-complete, i.e., it can perform any computation that a Turing machine can perform [43]. This is a key result in the field and shows that neuromorphic computing is capable of general-purpose computation. Furthermore, a framework for determining the computational complexity of neuromorphic algorithms has been proposed in the literature [44]. This allows us to compare the time and space complexity of neuromorphic algorithms to their conventional counterparts in a fair manner. Furthermore, several graph algorithms such as shortest path, minimum spanning trees, neighborhood subgraph extraction and max flow have been proposed in the literature [45]–[47]. Neuromorphic methods to solve differential equations and NP-complete problems have also been presented in the literature [48]–[50]. These advancements in neuromorphic computing showcase only the tip of the iceberg when it comes to the plethora of applications that could potentially benefit from neuromorphic computing.

Having said that, several advancements must be made in

neuromorphic computing, both at the software and hardware level, in order to fully realize its potential. For instance, at the software level, mechanisms for encoding rational numbers, alphabets and symbols need to be defined. Furthermore, neuromorphic abstractions for several logical, relational and math operators need to be built. A unified compiler framework for neuromorphic computing is critical for assisting users in application development and deployment on any neuromorphic hardware. At the hardware level, readily deployable hardware needs to be built and made available to research groups as well as consumers. Most reliable neuromorphic hardware today is digital, but their accessibility is limited. Analog neuromorphic hardware, on the other hand, is still in the research phase and not very reliable. Thus, neuromorphic computing offers tremendous potential in a plethora of scientific disciplines. However, to achieve its full potential, several advancements must be made in neuromorphic computing such as readily available neuromorphic hardware, standardized data encoding mechanisms, neuromorphic operators and gates, neuromorphic compilers, simulators and novel methods of addressing the explainability of SNN models.

## V. SUMMARY

Various scientific domains quickly adopted and applied deep learning (convolutional neural networks) to solve their challenges in analyzing data or as function approximators in simulation systems. However, convolutional neural networks do not lend themselves well to low size, weight, and power science applications. In addition, the inference time of a CNN can be at least an order of magnitude higher than a comparable spiking neural network (SNN). However, SNNs suffer from the lack of developmental tools and a higher learning curve relative to CNNs. Despite these challenges, neuromorphic computing shows significant promise to expand in the scientific applications where CNNs either can not be used or perform poorly relative to their SNN counterparts. In this paper, we shed some light on many scientific disciplines where neuromorphic computing is poised to make a significant impact. We also discussed the advancements that need to be made in neuromorphic computing at the software and hardware level in order to fully realize its potential. To establish and expand the use of neuromorphic computing in scientific applications, an open neuromorphic-oriented developmental ecosystem that reduces the barrier of entry and enhances adoption of the technology must be developed.

## REFERENCES

- [1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.
- [2] C. D. Schuman, T. E. Potok, R. M. Patton, J. D. Birdwell, M. E. Dean, G. S. Rose, and J. S. Plank, "A survey of neuromorphic computing and neural networks in hardware," *arXiv preprint arXiv:1705.06963*, 2017.
- [3] C. D. Schuman, S. R. Kulkarni, M. Parsa, J. P. Mitchell, B. Kay *et al.*, "Opportunities for neuromorphic computing algorithms and applications," *Nature Computational Science*, vol. 2, no. 1, pp. 10–19, 2022.
- [4] B. Rajendran, A. Sebastian, M. Schmuker, N. Srinivasa, and E. Eleftheriou, "Low-power neuromorphic hardware for signal processing applications: A review of architectural and system-level design approaches," *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 97–110, 2019.
- [5] C. S. Thakur, J. L. Molin, G. Cauwenberghs, G. Indiveri, K. Kumar, N. Qiao, J. Schemmel, R. Wang, E. Chicca, J. Olson Hasler *et al.*, "Large-scale neuromorphic spiking array processors: A quest to mimic the brain," *Frontiers in neuroscience*, p. 891, 2018.
- [6] F. Akopyan, J. Sawada, A. Cassidy, R. Alvarez-Icaza, J. Arthur, P. Merolla, N. Imam, Y. Nakamura, P. Datta, G.-J. Nam, B. Taba, M. Beakes, B. Brezzo, J. B. Kuang, R. Manohar, W. P. Risk, B. Jackson, and D. S. Modha, "Truenorth: Design and tool flow of a 65 mw 1 million neuron programmable neurosynaptic chip," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 10, pp. 1537–1557, 2015.
- [7] N. Risi, A. Aimar, E. Donati, S. Solinas, and G. Indiveri, "A spike-based neuromorphic architecture of stereo vision," *Frontiers in neurorobotics*, p. 93, 2020.
- [8] L. Tan and N. DeBardleben, "Failure analysis and quantification for contemporary and future supercomputers," 2019. [Online]. Available: <https://arxiv.org/abs/1911.02118>
- [9] J. L. Krichmar, P. Coussy, and N. Dutt, "Large-scale spiking neural networks using neuromorphic hardware compatible models," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 11, no. 4, pp. 1–18, 2015.
- [10] M. D. McDonnell, K. Boahen, A. Ijspeert, and T. J. Sejnowski, "Engineering intelligent electronic systems based on computational neuroscience [scanning the issue]," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 646–651, 2014.
- [11] J. C. Knight and T. Nowotny, "Gpus outperform current hpc and neuromorphic solutions in terms of speed and energy when simulating a highly-connected cortical model," *Frontiers in neuroscience*, vol. 12, p. 941, 2018.
- [12] J. B. Aimone, "Computational modeling of adult neurogenesis," *Cold Spring Harbor perspectives in biology*, vol. 8, no. 4, p. a018960, 2016.
- [13] E. Y. Walker, F. H. Sinz, E. Cobos, T. Muhammad, E. Froudarakis, P. G. Fahey, A. S. Ecker, J. Reimer, X. Pitkow, and A. S. Tolias, "Inception loops discover what excites neurons most using deep predictive models," *Nature neuroscience*, vol. 22, no. 12, pp. 2060–2065, 2019.
- [14] K. Hamilton, P. Date, B. Kay, and C. Schuman D., "Modeling epidemic spread with spike-based models," in *International Conference on Neuromorphic Systems 2020*, ser. ICONS 2020. New York, NY, USA: Association for Computing Machinery, 2020. [Online]. Available: <https://doi.org/10.1145/3407197.3407219>
- [15] C. Schuman, R. Patton, S. Kulkarni, M. Parsa, C. Stahl, N. Q. Haas, J. P. Mitchell, S. Snyder, A. Nagle, A. Shanafield, and T. Potok, "Evolutionary vs imitation learning for neuromorphic control at the edge\*," *Neuromorphic Computing and Engineering*, vol. 2, no. 1, 1 2022.
- [16] C. D. Schuman, S. R. Young, J. P. Mitchell, J. T. Johnston, D. Rose, B. P. Maldonado, and B. C. Kaul, "Low size, weight, and power neuromorphic computing to improve combustion engine efficiency," in *2020 11th International Green and Sustainable Computing Workshops (IGSC)*. IEEE, 2020, pp. 1–8.
- [17] B. P. Maldonado, B. C. Kaul, C. D. Schuman, S. R. Young, and J. P. Mitchell, "Dilute combustion control using spiking neural networks," SAE Technical Paper, Tech. Rep., 2021.
- [18] B. K. Bose, "Neural network applications in power electronics and motor drives—an introduction and perspective," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 1, pp. 14–33, 2007.
- [19] S. Debnath, S. Kulkarni, and C. Schuman, "Intelligent prediction of states in multi-port autonomous reconfigurable solar power plant (mars)," in *2021 IEEE Energy Conversion Congress and Exposition (ECCE)*. IEEE, 2021, pp. 1339–1346.
- [20] M. Gehrig, S. B. Shrestha, D. Mouritzen, and D. Scaramuzza, "Event-based angular velocity regression with spiking networks," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4195–4202.
- [21] B. Denby, P. Garda, B. Granado, C. Kiesling, J.-C. Prevotet, and A. Wassatsch, "Fast triggering in high-energy physics experiments using hardware neural networks," *IEEE Transactions on Neural Networks*, vol. 14, no. 5, pp. 1010–1027, 2003.
- [22] A. M. Terwilliger, G. N. Perdue, D. Isele, R. M. Patton, and S. R. Young, "Vertex reconstruction of neutrino interactions using deep learning," in

- 2017 *International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 2275–2281.
- [23] L. Song, F. Chen, S. R. Young, C. D. Schuman, G. Perdue, and T. E. Potok, “Deep learning for vertex reconstruction of neutrino-nucleus interaction events with combined energy and time data,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3882–3886.
- [24] C. D. Schuman, T. E. Potok, S. Young, R. Patton, G. Perdue, G. Chakma, A. Wyer, and G. S. Rose, “Neuromorphic computing for temporal scientific data classification,” in *Proceedings of the Neuromorphic Computing Symposium*, 2017, pp. 1–6.
- [25] R. M. Patton, J. T. Johnston, S. R. Young, C. D. Schuman, T. E. Potok, D. C. Rose, S.-H. Lim, J. Chae, L. Hou, S. Abousamra, D. Samaras, and J. Saltz, “Exascale deep learning to accelerate cancer research,” in *2019 IEEE International Conference on Big Data (Big Data)*, 2019, pp. 1488–1496.
- [26] R. Batra, L. Song, and R. Ramprasad, “Emerging materials intelligence ecosystems propelled by machine learning,” *Nature Reviews Materials*, vol. 6, pp. 655 – 678, 2020.
- [27] G. Cong, A. Gupta, and et al., “Prediction of co2 adsorption in nanopores with graph neural networks,” in *Proceedings of Workshop on Deep Learning on Graphs, AAAI-DLG22*. AAAI, 3 2022.
- [28] K. Oono and T. Suzuki, “Graph neural networks exponentially lose expressive power for node classification,” *arXiv: Learning*, 2020.
- [29] S. R. Kulkarni, M. Parsa, J. P. Mitchell, and C. D. Schuman, “Benchmarking the performance of neuromorphic and spiking neural network simulators,” *Neurocomputing*, vol. 447, pp. 145–160, 2021.
- [30] M. Diesmann and M.-O. Gewaltig, “Nest: An environment for neural systems simulations,” *Forschung und wissenschaftliches Rechnen, Beiträge zum Heinz-Billing-Preis*, vol. 58, pp. 43–70, 2001.
- [31] J. M. Eppler, M. Helias, E. Muller, M. Diesmann, and M.-O. Gewaltig, “Pynest: a convenient interface to the nest simulator,” *Frontiers in neuroinformatics*, vol. 2, p. 12, 2009.
- [32] D. F. Goodman and R. Brette, “Brian: a simulator for spiking neural networks in python,” *Frontiers in neuroinformatics*, vol. 2, p. 5, 2008.
- [33] M. Stimberg, R. Brette, and D. F. Goodman, “Brian 2, an intuitive and efficient neural simulator,” *Elife*, vol. 8, p. e47314, 2019.
- [34] E. Yavuz, J. Turner, and T. Nowotny, “Genn: a code generation framework for accelerated brain simulations,” *Scientific reports*, vol. 6, no. 1, pp. 1–14, 2016.
- [35] T. Bekolay, J. Bergstra, E. Hunsberger, T. DeWolf, T. C. Stewart, D. Rasmussen, X. Choo, A. Voelker, and C. Eliasmith, “Nengo: a python tool for building large-scale functional brain models,” *Frontiers in neuroinformatics*, vol. 7, p. 48, 2014.
- [36] H. Hazan, D. J. Saunders, H. Khan, D. Patel, D. T. Sanghavi, H. T. Siegelmann, and R. Kozma, “Bindsnet: A machine learning-oriented spiking neural networks library in python,” *Frontiers in neuroinformatics*, p. 89, 2018.
- [37] C. Gunaratne, N. Baral, W. Rand, I. Garibay, C. Jayalath, and C. Senevirathna, “The effects of information overload on online conversation dynamics,” *Computational and Mathematical Organization Theory*, vol. 26, no. 2, pp. 255–276, 2020.
- [38] P. S. Dodds and D. J. Watts, “A generalized model of social and biological contagion,” *Journal of theoretical biology*, vol. 232, no. 4, pp. 587–604, 2005.
- [39] C. Gunaratne, R. Reyes, E. Hemberg, and U.-M. O’Reilly, “Evaluating efficacy of indoor non-pharmaceutical interventions against covid-19 outbreaks with a coupled spatial-sir agent-based simulation framework,” *Scientific reports*, vol. 12, no. 1, pp. 1–11, 2022.
- [40] B. G. Aaby, K. S. Perumalla, and S. K. Seal, “Efficient simulation of agent-based models on multi-gpu and multi-core clusters,” in *Proceedings of the 3rd international icst conference on simulation tools and techniques*, 2010, pp. 1–10.
- [41] C. Gunaratne, W. Rand, and I. Garibay, “Inferring mechanisms of response prioritization on social media under information overload,” *Scientific reports*, vol. 11, no. 1, pp. 1–12, 2021.
- [42] N. Collier and M. North, “Repast hpc: A platform for large-scale agent-based modeling,” *Large-scale computing techniques for complex system simulations*, pp. 81–110, 2011.
- [43] P. Date, C. Schuman, B. Kay, and T. Potok, “Neuromorphic computing is turing-complete,” 2021. [Online]. Available: <https://arxiv.org/abs/2104.13983>
- [44] P. Date, B. Kay, C. Schuman, R. Patton, and T. Potok, “Computational complexity of neuromorphic algorithms,” in *International Conference on Neuromorphic Systems 2021*, ser. ICONS 2021. New York, NY, USA: Association for Computing Machinery, 2021. [Online]. Available: <https://doi.org/10.1145/3477145.3477154>
- [45] K. Hamilton, T. Mintz, P. Date, and C. D. Schuman, “Spike-based graph centrality measures,” in *International Conference on Neuromorphic Systems 2020*, ser. ICONS 2020. New York, NY, USA: Association for Computing Machinery, 2020. [Online]. Available: <https://doi.org/10.1145/3407197.3407199>
- [46] B. Kay, P. Date, and C. Schuman, “Neuromorphic graph algorithms: Extracting longest shortest paths and minimum spanning trees,” in *Proceedings of the Neuro-Inspired Computational Elements Workshop*, ser. NICE ’20. New York, NY, USA: Association for Computing Machinery, 2020. [Online]. Available: <https://doi.org/10.1145/3381755.3381762>
- [47] B. Kay, C. Schuman, J. O’Connor, P. Date, and T. Potok, “Neuromorphic graph algorithms: Cycle detection, odd cycle detection, and max flow,” in *International Conference on Neuromorphic Systems 2021*, ser. ICONS 2021. New York, NY, USA: Association for Computing Machinery, 2021. [Online]. Available: <https://doi.org/10.1145/3477145.3477172>
- [48] J. D. Smith, W. Severa, A. J. Hill, L. Reeder, B. Franke, R. B. Lehoucq, O. D. Parekh, and J. B. Aimone, “Solving a steady-state pde using spiking networks and neuromorphic hardware,” in *International Conference on Neuromorphic Systems 2020*, ser. ICONS 2020. New York, NY, USA: Association for Computing Machinery, 2020. [Online]. Available: <https://doi.org/10.1145/3407197.3407202>
- [49] K. Corder, J. V. Monaco, and M. M. Vindiola, “Solving vertex cover via ising model on a neuromorphic processor,” in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2018, pp. 1–5.
- [50] C. Yakopcic, N. Rahman, T. Atahary, T. M. Taha, and S. Douglass, “Solving constraint satisfaction problems using the loihi spiking neuromorphic processor,” in *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2020, pp. 1079–1084.