SAND2020-6020C

Truly heterogeneous HPC: Co-design to achieve what science needs from HPC

Suma George Cardwell, Craig Vineyard, Willam Severa, Frances Chance, Frederick Rothganger, Felix Wang, Srideep Musuvathy, Corinne Teeter, and James B. Aimone

Sandia National Laboratories, Albuquerque, NM, USA, 87123
sgcardw@sandia.gov,
WWW home page: www.sandia.gov

Abstract. Future high-performance computing (HPC) platforms increasingly depend on heterogeneous node architectures to meet power and performance requirements. While modern HPC design largely incorporates GPUs with CPU resources, there is potential to further integrate novel forms of computing. The ability to leverage efficient, non-conventional computing technologies would be a fundamentally disruptive development in advancing HPC. Neuromorphic computing is such an emerging technology, which would interest the HPC community, due to its potential for implementing large-scale calculations with an extremely low power footprint. We will explore the example of mapping the connectome of the brain to illustrate advantages of using a heterogeneous system that incorporates neuromorphic hardware.

Keywords: Neuromorphic Computing, Heterogeneous HPC

1 Overview

In recent years, there has been an increasing trend for high-performance computing (HPC) systems to incorporate multiple classes of processors on individual HPC boards. Embracing this heterogeneity has been invaluable in moving towards exascale computing, with significant reliance on general purpose graphics processing units (GPGPUs) to more efficiently implement large-scale problems that heavily rely on dense linear algebra. Recently, there has been more attention given to linear algebra accelerators, such as systolic arrays (i.e., Google's Tensor Processing Unit) to achieve further efficiencies for suitable computations. Unsurprisingly, this shift in HPC configuration has also expanded the scope of applications for which HPC is relevant to include many current computationally-expensive artificial intelligence (AI) tasks such as deep artificial neural networks (ANNs). Importantly, however, this broadening of HPC components has been limited to conventional processor approaches. Here, we present a vision for what we refer to as truly heterogeneous HPC, whereby HPC systems include both conventional components (e.g., CPUs, GPUs, systolic arrays) and non-conventional components, such as neuromorphic hardware and processing-in-memory (PIM)

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

devices. These emerging technologies promise substantial benefits in efficiency, especially in terms of power requirements, but they also require a distinct approach to computation. These architectures often can be thought of as extremely parallel with different trade-offs between precision and speed than are typically encountered in von Neumann systems. Furthermore, the use-cases for neuromorphic hardware continue to evolve. For instance, while the long-term impact of neuromorphic computing likely lies in future brain-derived algorithms [1]; much of the recent focus has been on accelerating ANNs [2, 3] and it is increasingly recognized to be capable for numerical computing applications [4, 5]. It is not immediately obvious whether neuromorphic approaches are critical for scientific applications that have driven HPC development to date. Since the original computers, computing technologies have evolved to solve the computationally intensive components of large physics models and similarly large-scale machine learning approaches, such as ANNs, have outperformed alternatives bolstered by GPUs. However, the scientific computing ecosystem is beginning to change. As data collection begins to outpace theory in fields such as neuroscience, medicine, and climatology, we increasingly find ourselves in a world where the simulation of physics models is less important than deriving insight from extremely large volumes of complex data. To illustrate this shift and how it would drive the eventual requirements of a truly heterogeneous HPC platform, we work through a specific scientific example: mapping and interpreting the connectome of the brain. The connectome example is both salient (the US Government and EU continue to spend significant funds on it) and representative of an emerging class of data-intensive scientific endeavors where classical modeling and analytics are only part of the solution. Within this example, we highlight how incorporating the scientific exploration of data changes how computing needs to be used and highlight how a system leveraging the strengths of CPUs, GPUs, accelerators and emerging technology such as neuromorphic computing will be invaluable and disruptive for HPC systems.

1.1 Connectomics and Electron Microscopy data

Mapping the connectome of a brain and deriving new understanding of the underlying neural circuit function requires addressing a number of key challenges. The technical challenge of scaling electron microscopy (EM) techniques to handle a volume the size of an entire brain[6, 7] comes with the challenge of analyzing the massive amounts of associated data. The first reconstruction of the *C. elegans* nervous system [8] was performed almost entirely by human-hand, requiring more than 10 years to map approximately 300 neurons and 7000 connections between them [9]. For comparison, a *Drosophila melanogaster* (fruit fly) brain comprises on the order of 100,000 neurons [10] while a mouse brain is estimated at 70 million neurons [11]. The raw data for one cubic millimeter of mouse visual cortex is on the order of 2 petabytes [12]. While advances in high-throughput EM [12, 13] and automated segmentation and reconstruction algorithms [14] signify the 'coming of age' of EM, interpreting newly-available, high-resolution whole-brain connectomes will require overcoming significant computational challenges.

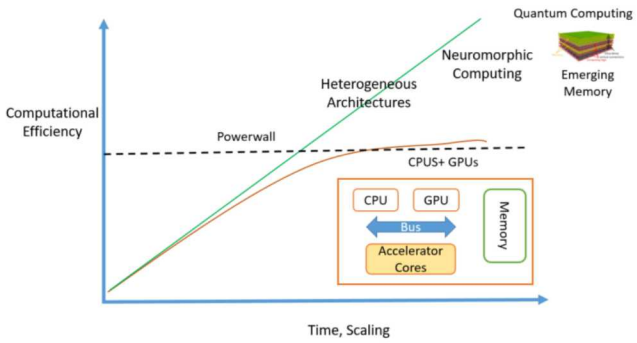

Fig. 1. The computational efficiency of modern general-purpose processors has hit a powerwall leading to the search for novel architectures and emerging devices.

As larger volumes from both invertebrate [10, 15–17] and mammalian [12, 18, 19] brains become available with increasingly dense reconstructions and more complete identification of different cell types and synaptic connections, so will the need for semi-automated and increasingly sophisticated analysis.

In this paper, we focus on how a heterogeneous platform may be leveraged to address the computational challenges associated with processing and analysis of the EM imagery, including segmentation and analysis of the resulting connectome graph. First, emerging technology may be used to accelerate current state-of-the-art methods for EM imagery analysis. The use of flood-filling networks for image segmentation and reconstruction [14] of large-volume EM constitutes state-of-the-art today (e.g. see [16, 17, 20]). While these networks perform with significantly better accuracy compared to alternative approaches, they are also computationally expensive. As we will discuss, we believe that some of our existing approaches to developing for neuromorphic systems may be leveraged to implement these networks at significantly lower computational cost.

Another challenge for fully realizing the potential of high-throughput EM is analyzing the connectome to draw meaningful conclusions regarding the organization and function of neural circuits. Larger-scale connectomes with online tools for visualization and analysis have only recently become widely available (for examples, see https://microns-explorer.org [18, 19] and https://neuprint.janelia.org [21]). Analysis of the associated neural graphs thus far have been largely limited to statistics describing the input/output connectivity of specific cell types [10, 19, 22] within individual volumes. Analysis of graphs combined with functional data [23, 24], or across multiple specimens [25] are less common but will likely require more sophisticated but semi-automated approaches as advanced EM technologies facilitate the availability of larger and more detailed connectome graphs. We believe that our approaches to accelerating AI algorithms on neuromorphic hardware can be extended to accelerate the process

of identifying meaningful graph motifs contained within these images, thereby facilitating meaningful interpretations of the data.

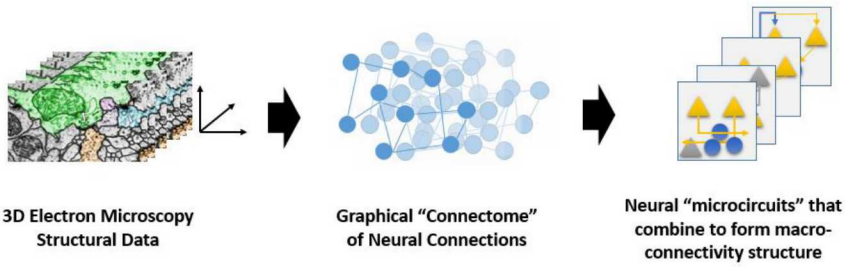

Fig. 2. Mapping the Brain Connectome from 3D EM Structural Data

1.2 Relevance to DOE and HPC

It is worth noting that the recent advances in EM methodologies have continued to draw the interest of BRAIN Initiative stakeholders, including NIH and NSF, as well as potential new investments from DOE. Today, most AI algorithms are designed independently of hardware considerations, with algorithm performance the dominant criterion for a successful AI approach. As a result, the extreme computational costs of emerging AI technologies, especially in deep learning, have led to an explosion of proposed ANN accelerators. These accelerators are largely conventional CMOS approaches tailored to accelerate the linear algebra operations that current algorithms prioritize. We recognize that future AI solutions, such as those integrated into high-throughput scientific pipelines, will leverage both deep learning-based ML approaches and other AI algorithms that may not be ideally suited for the current generation of deep learning accelerators. Additionally, our proposed co-design strategy is scoped for two additional observations regarding the scaling of AI performance: 1) the cumulative performance of an AI system is critical, not simply the acceleration of any particular kernel and 2) hardware acceleration cannot come at the expense of algorithm performance. EM image analysis is an attractive 'test' application space for our truly heterogeneous system because the field includes two important challenge problems. First, image processing of 3D electron microscopy data using deep neural networks already has a well-established approach as its solution (flood-filling networks). Second, decomposition of deep neural connectivity graphs at increasingly large scales is still relatively nascent and more effective approaches have not yet been well-established. Both challenges are in need of approaches to acceleration that can maintain performance without significantly increasing computational cost. We consider this application space to be a particularly attractive

domain because its challenges are illustrative of the data analytics pipeline in a number of scientific research areas and highlight the challenges associated with both ultra-large scale data and still rapidly-evolving AI and ML techniques.

2 Algorithmic Approach

Many scientific domains, ranging from astrophysics to materials science, are leveraging large scale data collection and a series of AI analyses to extract scientifically meaningful data. Image processing, or very similar data processing step, is often the first step of such scientific analysis pipelines, and many of the successful AI techniques being developed today are impacting this stage. The convolutional neural network component of this AI pipeline is a well-established algorithm that has broad applicability, and the process of identifying the computationally expensive parts of these neural networks and tailoring them for hardware acceleration is an immediately approachable research challenge. The algorithmic approach will be primarily to identify critical computational kernels that are suitable for neuromorphic hardware implementation that can be extracted from an overall AI pipeline. Below we describe approaches that can leverage neuromorphic architectures and enable the acceleration of EM image analysis with lower computational cost.

2.1 Deep Graph Decomposition

Deep learning methods, as they may be applied to analyze graph structure is still a developing field, with, for example, the work on graph neural networks (GNN) in recent years, see [26] for a review. In contrast to more commonly studied social or information graphs, however, the data extracted from EM image analysis admits a higher degree of complexity in its structure (e.g. cortical microcircuits, high fan-in/out, etc.). To remain informative and useful to the researcher, subgraph analysis techniques in this area will be important to specify salient neural circuit motifs, as well as measure their occurrence.

While the decomposition of graphs into subgraphs is typically in the purview of conventional graph analytics, the scale of connectomes and the requirement for tailoring answers towards an end-user's needs lends itself to being considered as a data-driven machine learning problem, furthermore leveraging the advancements in deep learning. Because the goal will be to decompose the graph structure from EM data into functionally relevant subgraphs, we refer to the approach as Deep Graph Decomposition, or DGD.

Supporting this approach are recent developments in graph embedding, such as graph2vec, structural-rnn, or LINE, which enable effective vector representations that may be useful in identifying critical, repeating features in graphs [27–29]. This is analogous to the role of convolution filters used for image processing problems or acquiring dictionary elements for sparse coding. The learned filters in either of these domains are effectively data-driven feature extractors.

More specific to image processing, these filters may combine and stack into a layered hierarchy. For our subgraph task, we are specifically interested in patterns that carry critical information about the composition (i.e. rate of occurrence) in the larger graph, and we hypothesize that these can be determined either directly (via inference) or indirectly (via network introspection).

Of note, embedding methods such as DeepWalk, which use random walks from graph vertices to generate representational signatures, may better leverage heterogeneous architectures [30]. We know from previous work [5] that neuromorphic systems can by highly efficient at computing diffusive random walks on graphs. By categorizing and counting the types of walks that are observed, we can extract an approximation of the common connectivity patterns within a given graph. In contrast with more conventional, state-of-the-art algorithms for subgraph counting (e.g. ESCAPE [31]), the motivating trade-off is to be able to extend beyond the limited subgraph sizes (e.g. up to five vertices) of exact methods. This leads to the scalability and subgraph complexity needed to analyze EM data, where moreover, there will be expected variability within equivalence-classes of neural circuit motifs.

2.2 Neuromorphic scaling of 3D Convolutional Neural Networks

Deep learning methods, particularly convolutional neural networks such as used in EM segmentation, are becoming increasingly common within scientific experimental workflows. Researchers in several fields have been able to use deep learning to help shift effort away from time-intensive tasks (e.g. hand-labeling images) or to help mitigate technical bottlenecks (e.g. when storing large-scale raw data is prohibitive). In large-scale applications, such as the use of flood-filling networks to segment neural EM data requires a considerable amount of compute power usually requiring a heterogeneous CPU/GPU system for high performance. This compute requirement is made complicated by the inclusion of 3D convolutional layers – a standard 2D convolution strides a 2D window (filter) across the x and y dimensions of an image, whereas a 3D convolution strides a 3D cube across the x, y and z dimensions of a 3D image or a stack of 2D images. These 3D convolutions are well-suited for stacked frames (such as those found in EM data or video) or other 3D imaging (such as MRI images) and despite possible acceleration via Fourier methods, these algorithms require more compute and more memory than the common 2D counterpart.

We can improve the inference performance of data-heavy neural networks by several orders of magnitude by jointly addressing algorithm and hardware challenges together. The most straightforward approach to making neural networks more efficient is to tailor algorithms to require less precision, in both weights and activation functions, along with hardware capable of benefiting from this low-precision.

High-performing neural networks traditionally use continuous-valued activation functions (e.g. rectified linear units) and floating-point precision weights. However, the high precision afforded by these representations is costly both in computation and communication. However, to address the challenge of big data

science applications, such as the aforementioned EM data, the scale of today's neuromorphic systems is vastly insufficient. For instance, the first layer of the flood-filling network would likely require over 1 billion neurons, well beyond the largest neuromorphic platforms available today. Such a scalable system would require further design trade-offs such as fixed precision weights or limited connectivity. We envision that future large-scale systems as described in Section 3.4 will rise up to these challenges.

3 Hardware Architecture

As digital systems saturate in terms of power efficiency, it is clear that the future of computing is heterogeneous. With the slowing of Moore's law and the resurgence of neural networks, many emerging technologies such as neuromorphic computing have gotten a new lease of life. Inspired by the brain, neuromorphic architectures try to leverage properties such as massive parallelism, sparse connectivity and event-driven computing. Neuromorphic engineering was pioneered by Prof. Carver Mead in the late 1980s to use silicon devices to mimic biology. These were analog circuits that utilized the sub-threshold dynamics of the CMOS transistor to emulate biological systems. Today neuromorphic systems have also come to encompass digital as well as mixed-signal approaches. Recently several large-scale neuromorphic projects have paved the way to demonstrating problems at scale on these systems. Spiking neuromorphic hardware fabricated in cutting edge technology nodes is rapidly progressing to a billion neurons from vendors such as Intel (Pohoiki Spring/Loihi). Recent developments in non-conventional devices like nanoscale memristors that can be integrated with CMOS also shows promising solutions to modeling dense synaptic memory.

Neuromorphic systems are uniquely suited to map and scale graphs because of parallelism, local connections, and efficiency gains. Conventional architectures of GPUs/CPUs are not suitable for graph based approaches, with data movement and updates being a bottleneck. Accelerator approaches can alleviate some of these issues, however neuromorphic approaches can yield 100x-1000x orders of magnitude efficiency gains.

3.1 Analog Neuromorphic Computing

Recently researchers at Sandia have shown that analog-in memory computations have a fundamental scaling advantage over digital memories. Analog crossbars have been projected to reduce energy and latency by three orders of magnitude compared to an optimized digital Application Specific Integrated Circuit (ASIC) [32]. Different classes of devices including TaOx Resitive RAM(ReRAM) and conventional floating-gate SONOS devices show promise. The analog ReRAM shows the most promise when compared to digital SRAM based ASIC with better performance when it came to area, energy and latency [32]. However, the algorithms used to train and learn on these devices are not optimized for the behavior of these devices. In-memory analog kernels are subject to analog noise

and variability. The inherent variability in these devices can be leveraged by incorporating the hardware characteristics as features while training. Furthermore, these systems tend to have lower bit-precision. Co-designing multi-precision algorithms for these devices and integration with conventional CMOS approaches will be crucial to unleashing their potential.

3.2 Digital Neuromorphic Computing

Developments in large-scale digital neuromorphic chips have shown the promise of these systems at scale. University of Manchester's SpiNNaker chip (130nm CMOS) represents a more configurable approach to neuromorphic cores with programmable ARM cores and an interconnect optimized for spiking [33]. This lends the platforms flexible to different neuron and synapse models. IBM's TrueNorth chip was the first neuromorphic chip with a million neurons [34]. Intel's Loihi is fabricated in 14nm FinFET technology with a 128 neuromorphic cores and with an integrated learning engine on-chip [35]. The SpiNNaker and Loihi architectures lend themselves well to scaling and are front runners in the race to achieving billion neurons with a million ARM core SpiNNaker system and Intel's Poihiki Springs at 100 million neurons [36]. Plans on building the next generation of SpiNNaker2 chips in 22nm FDX CMOS are currently underway [37]. Both systems support learning on-chip, are configurable and have a dedicated software stack to program the hardware. These systems also support research communities which is key to the adoption of such emerging technologies.

A recent paper compared simulation of a full scale cortical column simulated at speed 0.5x of real-time using NVIDIA Tesla V100 accelerator, and showed it had better performance than a CPU cluster and SpiNNaker neuromorphic system. However, the researchers conceded that through software improvements alone SpiNNaker could achieve 0.11μJoule/synaptic event [38]. Besides, the current billion neuron SpiNNaker chips were fabricated in 130nm CMOS technology node and scaling down to 22nm(as planned) would considerably improve performance. Such large scale demonstrations of neuromorphic supercomputers will further demonstrate the possibilities for not only brain-inspired simulation but their applicability to other scientific domains.

3.3 Integrating Neuromorphic Computing with conventional HPC: Optimizing System Architecture

The fundamental principle guiding architecture design is to match the structure of the physical machine to the structure of the algorithm. This leads us to focus on two secondary principles: heterogeneity and information distance. **Heterogeneity** – No single machine structure will best fit every algorithm, even within the specific domain of neural-inspired algorithms. The mix of available core types still represents a commitment to a particular range of algorithms. This can be addressed by carefully planning for the average case and then idling some of the system. Alternately, different installation could choose different combinations of 'plug-and-play' hardware modules to target a more specific set of algorithms.

Information Distance – Data movement is the key limit in modern systems. Individual transistors are already extremely efficient, requiring on the order of 1e-17 Joules of energy to switch, not far above the thermal noise limit of ≈ 40kbT=2e-19 J. However, communication is orders of magnitude costlier, requiring around 1 pJ to move data across a chip. The cost of computation is dominated by Joules/(bit*meter). That is, energy cost scales with the distance information must move. Consequently, the focus would be to use neuromorphic accelerator kernels that process in memory and minimize data movement.

A full system design will consist of the following levels:

- **Core**: A single processing block. This may be either analog or digital.
- **Package:** A collection of cores assembled on a single die, or perhaps a vertically-integrated stack of dies. The package may be heterogeneous, containing several different types of cores, and perhaps mixing digital with analog cores. A key question is how heterogeneous cores communicate with each other. We make the simplifying assumption that cores always connect to a digital network and follow a standard protocol. This protocol will be designed to scale up to system and cluster levels.
- **System**: Neuromorphic packages may be integrated with conventional components (GPUs, CPUs, memory banks) on a compute node. Each package could have dual-ported memory, such that it can be accessed on the main system bus, or it may be accessed solely through the neuromorphic network protocol, in which case a bridge device will appear to the rest of the system.
- **Cluster**: Specifies how to scale-up systems which include neuromorphic components to work efficiently at the petascale (machines that occupy an entire warehouse or data-center). Interesting questions include whether there is any impact on the design of cluster system due to the presence of neuromorphic components. For example, will it move event packets over the main network backbone, or will there be a separate neuromorphic network fabric?

To achieve this objective, high-level architecture simulations will be needed to search the design space for good matches to specific algorithms. Tools that optimize the system architecture to minimize costs such as energy, area, and time will be key. This is analogous to the SWaP (size, weight, and power) constraints often cited in neuromorphic applications, but here we are less concerned about spatial restrictions and more concerned with throughput. Developing tools that help evaluate mixed-precision, highly heterogeneous architectures incorporating neuromorphic components will be key to enable adoption of these novel neuromorphic processors. We wil discuss co-design algorithms like the Joint Neural Architecture and Hardware Search in Section 4.2 as well as analytical modeling tools in Section 4.1.

3.4 Novel approaches

Novel approaches to build three-dimensional architectures and wafer-scale integrated circuits can further disrupt. Early demonstration of 3D memory has been

promising and integrating it densely with CMOS processing units will yield further advantages. Stanford's Nano-Engineered Computing Systems Technology (N3XT) program offer insight into 3D architectures via their simulation framework for highly integrated ultradense (monolithic) 3-D integration of thin layers of logic and memory devices that are fabricated at low temperature[39, 40].

Wafer-scale processors are an approach to dramatically reduce communication overhead for large-scale systems but are very challenging owing to thermal issues as well as process yield issue. BrainScaleS is an example of a wafer-scale neuromorphic system with analog circuits to emulate point neurons and digital communication (digital interconnect network) fabricated [41]. Current wafer-scale accelerator chip like Cerebras demonstrate that wafer-scale approaches are feasible at lower technology nodes with a lot of innovation in fabrication and packing of these system.

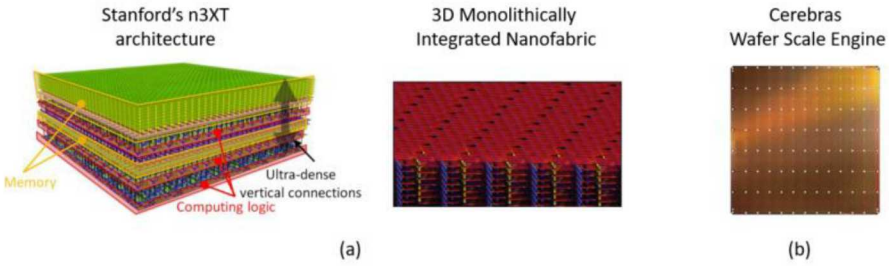

Fig. 3. Novel Approaches in development (a) 3D memory and compute architecture. Breakthroughs in CMOL (CMOS + Molecular nanodevices) allow us to build for Terabyte-density memory cells. Image reproduced from [40]. (b) Wafer-Scale Systems such as Cerebras' Wafer Scale Engine promise high bandwidth and low latency.

4 Co-Design of Heterogeneous Architectures

While algorithm-hardware co-design is critical for achieving high performance and energy efficiency, there is a practical challenge in linking design at these different scales. In terms of hardware development, a bottom-up approach is typically followed, whereby architectural designs are assumed and potentially-accelerated algorithms are sought after the fact. Similarly, because most real-world AI research focuses on task performance, the implications of algorithm design choice on potential hardware acceleration are often considered once an approach is set. To achieve the overall objective, both algorithmic and hardware optimizations need to be incorporated into a design.

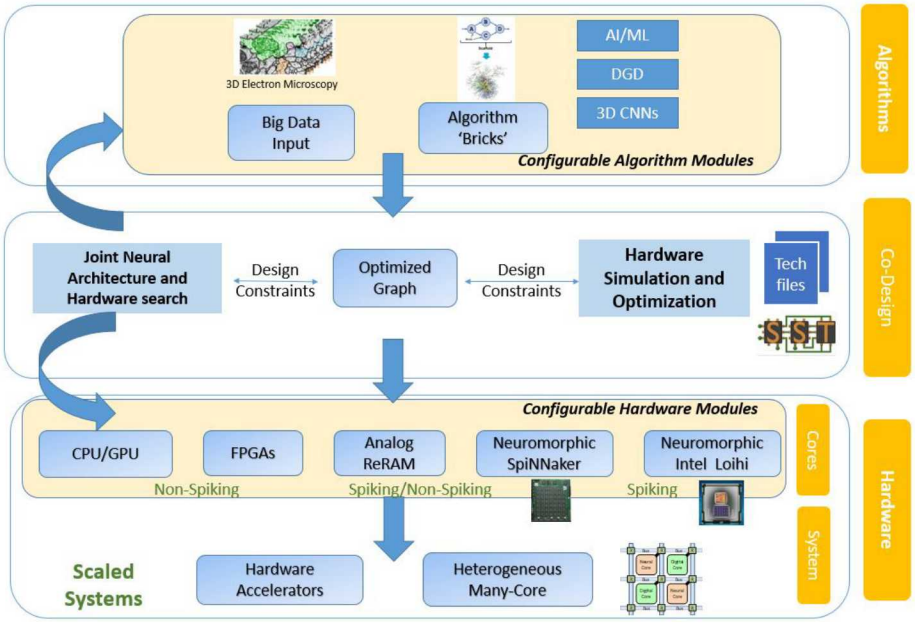

Fig. 4. Co-design of Algorithms and Architectures is critical for heterogeneous HPC systems.

4.1 Analytical Modeling

To meet the computational demands of ML workloads, exploration of accelerator designs has introduced a new 'Golden Age in Computer Architecture' [42]. Enabling this research, a spectrum of computer architecture design tools have been emerging to facilitate research into these new computational architectures. This ranges from analytical assessments to high fidelity simulations. The analytical approaches assess the steps which must occur for a given neural network to be computed given the architectural choices of a target platform. This includes calculating how the computation must be decomposed to pass through the computational units, how many memory accesses are required for retrieving input values and weights as well as storing results, and how communication structures facilitate these data movements. These counts are then multiplied by appropriate costs attributed to a targeted node technology (e.g. how much energy a multiplication or memory access requires). Effectively, this forecasts how a neural network maps onto a target architecture. Example analytical approaches include Modeling Accelerator Efficiency via Spatio-Temporal Resource Occupancy (MAESTRO) and Eyeriss Eyexam [43, 44]. Other analytical tools focus upon assessing properties of a hardware architecture such as the utilization of resources and identifying what is an optimal dataflow strategy for the architecture. An example is the Timeloop tool [45]. More accurate, but slower tools offer cycle accurate simulation capabilities. This increased fidelity often incorporates component models to attain the cycle accurate analysis and some-

times couples with executable hardware description level simulations. Examples include Systolic CNN AcceLErator Simulator (SCALE Sim) and Nvidia Deep Learning Accelerator (NVDLA) [46, 47]. The above techniques have largely focused upon ML accelerator approaches such as systolic arrays and CNN accelerators. Additional interest is in how emerging neuromorphic architectures may also be modeled. For example, NeMo utilizes the Rensselaer's optimistic simulation system (ROSS) discrete event simulation tool to provide a functional simulation of the IBM TrueNorth spiking neuromorphic architecture [48]. Other capabilities seek to account for the performance of emerging device technologies such as CrossSim and PUMA [32, 49]. Effectively, this spectrum of analytical modeling capabilities help enable co-design and the assessment of the impact of incorporating emerging ML accelerator and neuromorphic architectures into truly heterogeneous HPC systems.

4.2 Joint Neural Hardware and Architecture Search

Currently, the deep learning community increasingly leverages systematic parameter exploration of the algorithm space, but it generally does not explicitly consider the interaction of algorithms with its hardware implementation. Hyperparameter optimization techniques are often used to systematically explore sets of parameters – such as learning rates, kernel widths, and layer sizes – to help tune neural network structures to optimize algorithm performance in new domains. Hardware constraints can also be viewed as hyperparameters that can be optimized for.

4.3 Learning Algorithms for Neuromorphic Hardware

In contrast to standard artificial neural network (ANN) training methods, neuromorphic hardware increasingly utilizes brain-inspired, local-learning rules to update weights between nodes. Standard ANNs implemented on CPUs are often trained using extended versions of gradient decent (Rumelhart et al., 1986) learning algorithms. Although these ANNs have proven quite effective at specific tasks, even surpassing human performance on some, such as image processing (Russakovsky, 2015), natural language processing (LeCun et al, 2015), and playing games (Minh et al. 2015; Silver et al. 2016), there are drawbacks to these networks. Weight adjustments require both a forward and a backpropagation pass through the entire network. This makes them computationally expensive to train. They require enormous amounts of labeled data for training and they can be quite rigid and fail in unexpected and catastrophic ways (Eykholt et al, 2018). Many, techniques have been developed to address these problems, however, resolutions only treat the symptoms, not underlying issues.

The ability of biological brains to quickly synthesize, process, and act on large or small amounts of unlabeled data, while consuming very small amounts of power, have long inspired scientists and engineers from all fields. Brains use a different approach for learning. In local learning, also referred to as Hebbian learning (Hebb, 1949) or spike time dependent plasticity (STDP; Caporale and

Dan, 2008) in biology, the weights are adjusted between the pre and post synaptic neurons based on their activity. If a pre synaptic neuron fires before a post synaptic neuron, the weight is strengthened. If post fires before pre, the strength is weakened. Although there are feedback signals in the brain, it is unlikely there is an error signal backpropagated though the network. Instead, it appears the brain uses correlated activity to learn patterns in mostly unlabeled data.

Local learning may have substantial computational benefits. First, it enables learning with spikes without contrived methods to estimate gradients. Second, it is intrinsically parallel; learning does not need a signal to be forward and then backpropagated though a network. Third, it is relatively unsupervised; weights are strengthened via correlated activity, not, via a backpropagated error signal. How to effectively utilize local learning in deep networks is an active research topic in neuroscience and computer science. The realization of local learning will likely unleash the next generation of adaptive, low-power, deep neural networks; neuromorphic hardware is ready to capitalize on these new algorithms.

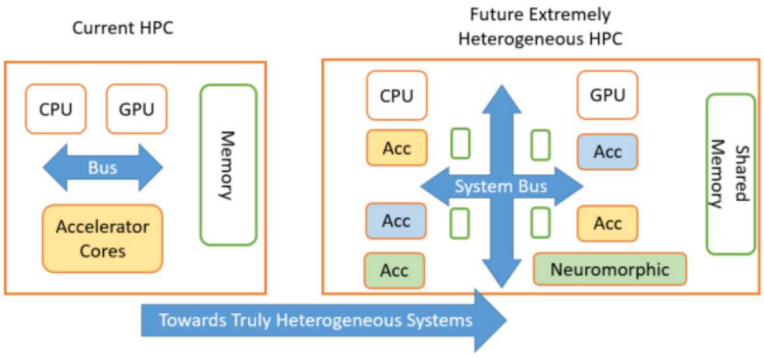

Fig. 5. Future of Heterogeneous computing

5 Future of HPC: Truly Heterogeneous Architectures

The low cost of development, fabrication and testing dictated the development of synchronous digital approaches so far. But, with the very high development and fabrication cost of sub-10nm CMOS circuits, the trend in the industry is to move towards more specialized hardware as opposed to general-purpose processors. This is truly a 'Golden Age for Computer Architecture', with innovation required from devices to architectures. But with AI/ML algorithms as compelling use cases for these architectures, co-design of hardware and algorithms will be crucial. The future of HPC is heterogeneous and committing to a truly heterogeneous approach has the potential to fundamentally change the role of

computing in science. We discussed the example of brain connectomics using serial electron microscopy (EM) to construct the 'connectome' (i.e., the graph of neurons and connections between them) of progressively larger volumes of brain tissue. Producing terabytes of data per day, image analysis of EM data already demands an HPC approach. However, the ultimate goal of EM of the brain is to extract computational understanding of its structure in order to advance neuroscience. Neuromorphic technologies, specifically, provide both low-power and configurable acceleration of such challenging AI algorithms. If designed into a heterogeneous system with other accelerators and conventional computing platforms, this technology has the potential to augment the capabilities of traditional HPC platforms. We described the strategies to integrate neuromorphic accelerators to design highly heterogeneous HPC platforms to enable massively parallel computation.

References

1. J. B. Aimone, "Neural algorithms and computing beyond moore's law," *Communications of the ACM*, vol. 62, no. 4, pp. 110–110, 2019.
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. W. Severa, C. M. Vineyard, R. Dellana, S. J. Verzi, and J. B. Aimone, "Training deep neural networks for binary communication with the whetstone method," *Nature Machine Intelligence*, vol. 1, no. 2, pp. 86–94, 2019.
4. J. B. Aimone, K. E. Hamilton, S. Mniszewski, L. Reeder, C. D. Schuman, and W. M. Severa, "Non-neural network applications for spiking neuromorphic hardware," in *Proceedings of the Third International Workshop on Post Moores Era Supercomputing*, 2018, pp. 24–26.
5. W. Severa, O. Parekh, K. D. Carlson, C. D. James, and J. B. Aimone, "Spiking network algorithms for scientific computing," in *2016 IEEE International Conference on Rebooting Computing (ICRC)*. IEEE, 2016, pp. 1–8.
6. C. J. Peddie and L. M. Collinson, "Exploring the third dimension: volume electron microscopy comes of age," *Micron*, vol. 61, pp. 9–19, 2014.
7. J. Kornfeld and W. Denk, "Progress and remaining challenges in high-throughput volume electron microscopy," *Current opinion in neurobiology*, vol. 50, pp. 261–267, 2018.
8. J. G. White, E. Southgate, J. N. Thomson, and S. Brenner, "The structure of the nervous system of the nematode caenorhabditis elegans," *Philos Trans R Soc Lond B Biol Sci*, vol. 314, no. 1165, pp. 1–340, 1986.
9. V. Jain, H. S. Seung, and S. C. Turaga, "Machines that learn to segment images: a crucial technology for connectomics," *Current opinion in neurobiology*, vol. 20, no. 5, pp. 653–666, 2010.
10. Z. Zheng, J. S. Lauritzen, E. Perlman, C. G. Robinson, M. Nichols, D. Milkie, O. Torrens, J. Price, C. B. Fisher, N. Sharifi *et al.*, "A complete electron microscopy volume of the brain of adult drosophila melanogaster," *Cell*, vol. 174, no. 3, pp. 730–743, 2018.
11. S. Herculano-Houzel, B. Mota, and R. Lent, "Cellular scaling rules for rodent brains," *Proceedings of the National Academy of Sciences*, vol. 103, no. 32, pp. 12 138–12 143, 2006.

12. W. Yin, D. Brittain, J. Borseth, M. E. Scott, D. Williams, J. Perkins, C. Own, M. Murfitt, R. M. Torres, D. Kapner *et al.*, "A petascale automated imaging pipeline for mapping neuronal circuits with high-throughput transmission electron microscopy," *bioRxiv*, p. 791889, 2019.
13. C. S. Xu, S. Pang, K. J. Hayworth, and H. F. Hess, "Enabling fib-sem systems for large volume connectomics and cell biology," *bioRxiv*, p. 852863, 2019.
14. M. Januszewski, J. Kornfeld, P. H. Li, A. Pope, T. Blakely, L. Lindsey, J. Maitin-Shepard, M. Tyka, W. Denk, and V. Jain, "High-precision automated reconstruction of neurons with flood-filling networks," *Nature methods*, vol. 15, no. 8, pp. 605–610, 2018.
15. K. Eichler, F. Li, A. Litwin-Kumar, Y. Park, I. Andrade, C. M. Schneider-Mizell, T. Saumweber, A. Huser, C. Eschbach, B. Gerber *et al.*, "The complete connectome of a learning and memory centre in an insect brain," *Nature*, vol. 548, no. 7666, pp. 175–182, 2017.
16. C. S. Xu, M. Januszewski, Z. Lu, S.-y. Takemura, K. Hayworth, G. Huang, K. Shinomiya, J. Maitin-Shepard, D. Ackerman, S. Berg *et al.*, "A connectome of the adult drosophila central brain," *BioRxiv*, 2020.
17. L. K. Scheffer, C. S. Xu, M. Januszewski, Z. Lu, S.-y. Takemura, K. J. Hayworth, G. Huang, K. Shinomiya, J. Maitlin-Shepard, S. Berg *et al.*, "A connectome and analysis of the adult drosophila central brain," *BioRxiv*, 2020.
18. S. Dorkenwald, N. L. Turner, T. Macrina, K. Lee, R. Lu, J. Wu, A. L. Bodor, A. A. Bleckert, D. Brittain, N. Kemnitz *et al.*, "Binary and analog variation of synapses between cortical pyramidal neurons," *bioRxiv*, 2019.
19. C. M. Schneider-Mizell, A. L. Bodor, F. Collman, D. Brittain, A. A. Bleckert, S. Dorkenwald, N. L. Turner, T. Macrina, K. Lee, R. Lu *et al.*, "Chandelier cell anatomy and function reveal a variably distributed but common signal," *bioRxiv*, 2020.
20. P. H. Li, L. F. Lindsey, M. Januszewski, M. Tyka, J. Maitin-Shepard, T. Blakely, and V. Jain, "Automated reconstruction of a serial-section em drosophila brain with flood-filling networks and local realignment," *Microscopy and Microanalysis*, vol. 25, no. S2, pp. 1364–1365, 2019.
21. J. Clements, T. Dolafi, L. Umayam, N. L. Neubarth, S. Berg, L. K. Scheffer, and S. M. Plaza, "neuprint: Analysis tools for em connectomics," *BioRxiv*, 2020.
22. L. K. Scheffer, "Graph properties of the adult drosophila central brain," *bioRxiv*, 2020.
23. D. D. Bock, W.-C. A. Lee, A. M. Kerlin, M. L. Andermann, G. Hood, A. W. Wetzel, S. Yurgenson, E. R. Soucy, H. S. Kim, and R. C. Reid, "Network anatomy and in vivo physiology of visual cortical neurons," *Nature*, vol. 471, no. 7337, pp. 177–182, 2011.
24. P. Zhou, J. Reimer, D. Zhou, A. Pasarkar, I. A. Kinsella, E. Froudarakis, D. Yatsenko, P. Fahey, A. Bodor, J. Buchanan *et al.*, "Ease: Em-assisted source extraction from calcium imaging data," *bioRxiv*, 2020.
25. D. Witvliet, B. Mulcahy, J. K. Mitchell, Y. Meirovitch, D. K. Berger, Y. Wu, Y. Liu, W. X. Koh, R. Parvathala, D. Holmyard *et al.*, "Connectomes across development reveal principles of brain maturation in c. elegans," *bioRxiv*, 2020.
26. J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *arXiv preprint arXiv:1812.08434*, 2018.
27. A. Narayanan, M. Chandramohan, R. Venkatesan, L. Chen, Y. Liu, and S. Jaiswal, "graph2vec: Learning distributed representations of graphs," *arXiv preprint arXiv:1707.05005*, 2017.

28. A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-rnn: Deep learning on spatio-temporal graphs," in *Proceedings of the ieee conference on computer vision and pattern recognition*, 2016, pp. 5308–5317.
29. J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th international conference on world wide web*, 2015, pp. 1067–1077.
30. B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701–710.
31. A. Pinar, C. Seshadhri, and V. Vishal, "Escape: Efficiently counting all 5-vertex subgraphs," in *Proceedings of the 26th International Conference on World Wide Web*, 2017, pp. 1431–1440.
32. S. Agarwal, A. Hsia, R. Jacobs-Gedrim, D. R. Hughart, S. J. Plimpton, C. D. James, and M. J. Marinella, "Designing an analog crossbar based neuromorphic accelerator," in *2017 Fifth Berkeley Symposium on Energy Efficient Electronic Systems & Steep Transistors Workshop (E3S)*. IEEE, 2017, pp. 1–3.
33. S. B. Furber, F. Galluppi, S. Temple, and L. A. Plana, "The spinnaker project," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 652–665, 2014.
34. P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura *et al.*, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, 2014.
35. M. Davies, N. Srinivasa, T.-H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain *et al.*, "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, 2018.
36. *Intel Scales Neuromorphic Research System to 100 Million Neurons*, March 18th 2020 (accessed June 13th, 2020). [Online]. Available: https://newsroom.intel.com/news/intel-scales-neuromorphic-research-system-100-million-neurons/#gs.7xo39i
37. S. Höppner and C. Mayr, "Spinnaker2-towards extremely efficient digital neuromorphics and multi-scale brain emulation," in *Proc. NICE*, 2018.
38. 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.
39. M. M. S. Aly, M. Gao, G. Hills, C.-S. Lee, G. Pitner, M. M. Shulaker, T. F. Wu, M. Asheghi, J. Bokor, F. Franchetti *et al.*, "Energy-efficient abundant-data computing: The n3xt 1,000 x," *Computer*, vol. 48, no. 12, pp. 24–33, 2015.
40. M. M. S. Aly, T. F. Wu, A. Bartolo, Y. H. Malviya, W. Hwang, G. Hills, I. Markov, M. Wootters, M. M. Shulaker, H.-S. P. Wong *et al.*, "The n3xt approach to energy-efficient abundant-data computing," *Proceedings of the IEEE*, vol. 107, no. 1, pp. 19–48, 2018.
41. J. Schemmel, J. Fieres, and K. Meier, "Wafer-scale integration of analog neural networks," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. IEEE, 2008, pp. 431–438.
42. J. Dean, D. Patterson, and C. Young, "A new golden age in computer architecture: Empowering the machine-learning revolution," *IEEE Micro*, vol. 38, no. 2, pp. 21–29, 2018.
43. H. Kwon, M. Pellauer, and T. Krishna, "Maestro: an open-source infrastructure for modeling dataflows within deep learning accelerators," *arXiv preprint arXiv:1805.02566*, 2018.

44. Y.-H. Chen, T.-J. Yang, J. Emer, and V. Sze, "Eyeriss v2: A flexible accelerator for emerging deep neural networks on mobile devices," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 9, no. 2, pp. 292–308, 2019.
45. A. Parashar, P. Raina, Y. S. Shao, Y.-H. Chen, V. A. Ying, A. Mukkara, R. Venkatesan, B. Khailany, S. W. Keckler, and J. Emer, "Timeloop: A systematic approach to dnn accelerator evaluation," in *2019 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*. IEEE, 2019, pp. 304–315.
46. A. Samajdar, Y. Zhu, P. Whatmough, M. Mattina, and T. Krishna, "Scale-sim: Systolic cnn accelerator simulator," *arXiv preprint arXiv:1811.02883*, 2018.
47. 2020. [Online]. Available: http://nvdla.org/index.html
48. M. Plagge, C. D. Carothers, E. Gonsiorowski, and N. Mcglohon, "Nemo: A massively parallel discrete-event simulation model for neuromorphic architectures," *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 28, no. 4, pp. 1–25, 2018.
49. A. Ankit, I. E. Hajj, S. R. Chalamalasetti, G. Ndu, M. Foltin, R. S. Williams, P. Faraboschi, W.-m. W. Hwu, J. P. Strachan, K. Roy *et al.*, "Puma: A programmable ultra-efficient memristor-based accelerator for machine learning inference," in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, 2019, pp. 715–731.