A Roadmap To Post-Moore Era for Distributed Systems

Vincenzo De Maio
Vienna University of Technology
Vienna, Austria
vincenzo@ec.tuwien.ac.at

Atakan Aral
University of Vienna
Vienna, Austria
atakan.aral@univie.ac.at

Ivona Brandic
Vienna University of Technology
Vienna, Austria
ivona@ec.tuwien.ac.at

ABSTRACT

We are reaching the limits of the von Neumann computing architectures (also called Moore's law era) as there is no free ride of the performance growth from simply shrinking the transistor features. As one of the consequences, we experience the rise of highly specialized architectures ranging from neuromorphic to quantum computing, exploiting completely different physical phenomena and demanding the development of entirely new architectures – that, however, can perform the computations within a fraction of the energy needed by the von Neumann architecture. Thus, we experience the paradigm shift from generalized architectures of the Von Neumann era to highly specialized architectures in the Post-Moore's law era where we expect the coexistence of multiple types of architectures specialized for different types of computation. In this paper, we discuss the implications of the post-Moore's law era for distributed systems.

ACM Reference Format:
Vincenzo De Maio, Atakan Aral, and Ivona Brandic. 2022. A Roadmap To Post-Moore Era for Distributed Systems. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

The distributed systems research community currently faces challenges given by the explosion of data generated from different sources, ranging from IoT sensors, mobile phones, and wearable devices. This vast amount of data is characterized by heterogeneity in format and size (e.g., time-series data, video frames, audio files) and has to be analyzed at different layers of distributed systems within the order of milliseconds to address the requirements of modern applications. To address strict latency requirements, Edge computing has been proposed. Edge computing reduces the latency of data processing by employing so-called Edge nodes, i.e., lower scale data centers, which augment the capabilities of classic Cloud computing by moving processing closer to the source of data. In some cases, Edge can be further augmented with the use of specific hardware accelerators, such as GPUs, FPGAs, ASICs, and TPUs.

However, due to the unsustainability of Moore's law [13] and the failure of Dennard's scaling [5], while on one side amount of data generated is growing together with the demand for computational resources to process them; on the other side, conventional computing architectures are reaching their physical limits. This huge discrepancy between the growth of data generation and the increase of computational processing power demands a robust research effort in novel approaches and alternatives to conventional data processing.

In this work, we discuss the potential of non-Von Neumann's architectures to address the challenges of the so-called Post-Moore's law era. First, we define the two of the architectures that attracted the most interest in the research community and where we witness not only theoretical developments but also first implementations and practical use cases. Afterward, we discuss the first ideas but also challenges in the integration of identified architectures in existing distributed systems. Finally, we present initial use case scenarios for quantum and neuromorphic computing, which have shown promising results in the area of scientific computing and artificial intelligence. In this context, we present the idea of Quantum Edge and Neuromorphic Edge, describing possible ways to integrate them into classic distributed architectures.

This work is organized as follows: first, we describe the limitations of Von Neumann architecture in Section 2. Then, we describe modern architectures addressing Von Neumann architecture limitation in Section 3 and their implications for modern distributed systems in Section 4. In Section 5 we discuss concrete use cases. Finally, we conclude our paper in Section 6.

2 THEORETICAL LIMITATIONS OF VON NEUMANN ARCHITECTURE

Almost all modern computers are based on the von Neumann architecture as illustrated on the left side of the Figure 1. In this architecture, there exists a central processing unit (CPU) and a separate memory that stores data temporarily during processing. The data is transferred back and forth between the CPU and memory unit through a data path. Until now, this architecture allowed continuous growth of the processing speed as predicted by Moore's law. However, the current processing unit and memory speeds have reached a point that the data path becomes a performance bottleneck, which will inevitably, within this decade, stop the already lagging growth [16]. Indeed, there exists a practical limitation to processor frequency of around 4 GHz since 2006.

Since a single computer cannot be any faster, researchers are developing systems composed of thousands of CPUs to exploit parallelism. However, this results in extremely high energy consumption, and therefore, such systems are not feasible for widespread use. As an example, the fastest supercomputer currently, Fugaku[1], can perform 537 quadrillions (10^{15}) floating-point operations per

[1]https://www.fujitsu.com/global/about/innovation/fugaku/

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

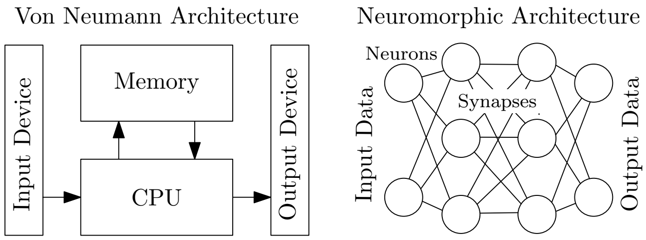

Figure 1: Von Neumann vs Neuromorphic Architecture

second using 158,976 processing units with an energy budget of 30 to 40 MW, which is comparable to 100,000 average EU households.

The main reason for energy consumption is that power density and heat dissipation increases as transistors get smaller and smaller because these do not scale with size anymore (the end of Dennard's scaling [5]). In addition to the loss of energy in the form of heat, cooling systems also consume the majority of energy in large data centers so that the processors can continue functioning properly.

3 THE RISE OF NEW ARCHITECTURES

Since the rise of microprocessors in the early 1970s, we experienced a paradigm shift from highly specialized hardware (i.e., time-shared mainframes used mostly for scientific and engineering calculations) to general-purpose calculators known as personal computers, as envisioned by [1]. However, the constant data increase and the consequent improvement in computational power demand encouraged the scientific community to employ specialized accelerators (e.g., GPUs, FPGAs, ASICs, TPUs) to improve the performance of computation and data-intensive tasks such as scientific workflows [22], deep learning computations [24] and bitcoin mining [20]. The incoming limit of Von Neumann's architecture [25] poses a solid motivation to come back to highly specialized hardware to satisfy the increasing computational power demands. Among a plethora of new computer architecture proposals, we selected the most promising, namely, *neuromorphic computing* and *quantum computing*.

3.1 Neuromorphic Computing

The first definition of neuromorphic computing appears in [12]. The main idea behind neuromorphic computing is to mimic the behavior of the human brain, meaning that computation is represented in the forms of *neurons* and *synapses* (See the right side of the Figure 1). The main differences between neuromorphic architectures are summarized in Figure 1. The most important is that, while Von Neumann's architecture imposes a strict separation between memory and processing, in neuromorphic computing, both processing and memory are encoded by neurons and synapses. Also, while Von Neumann computers encode information as numerical binary values, the input of neuromorphic computers are spikes. The time, magnitude, and shape at which a spike occurs can be used to encode numerical information; therefore, we can convert spikes into binary values and viceversa [17]. Other differences are:

- Highly parallel operations, since each neuron and synapses could be operated simultaneously;
- No memory bottlenecks, since contrarily to Von Neumann's architecture, there is no need to transfer data to/from the CPU, being memory and processing encoded together;
- High scalability, since adding a new neuromorphic chip causes the increase in the number of neurons and synapses, with positive effects on computation [4].

The main application area for neuromorphic computing is Artificial Intelligence [19] and Scientific Computing [17].

3.2 Quantum Computing

The basic unit of quantum computation are the *qubits*. In contrast to classic bits, which can be either 0 or 1, a qubit can be in a *superposition* of both. A set of n qubits taken together forms a *quantum register*. Quantum computation is performed by manipulating qubits in a quantum register. The state of a n-qubits register $|\psi\rangle$ is a linear combination of n column vectors (orthonormal basis)

$$\begin{aligned} |0\rangle &\mapsto [1, 0, \ldots, 0]^T \\ |1\rangle &\mapsto [0, 1, \ldots, 0]^T \\ &\vdots \\ |n-1\rangle &\mapsto [0, 0, \ldots, 1]^T. \end{aligned}$$

$|\psi\rangle = \sum_{i=0}^{n-1} c_i \, |i\rangle$ with *complex* coefficients (*complex amplitudes*).

In contrast to classic registers, a quantum register is in a *superposition* of each state, i.e., can be in each one of $x_0, \ldots x_n$ at the same time. At the moment of observation, the probability that we will find it in the state x_i, $P(x_i)$ is given by c_i. As a consequence of the superposition principle, quantum computers can process 2^n values at the same time (the so-called "quantum parallelism"), which is the primary source of the quantum speedup [15] in comparison with classic Von Neumann architectures.

Quantum computing has the potential to offer a significant computational advantage over Von-Neumann's architectures, which allows for solving different intractable problems in various application domains, ranging from finance, molecular dynamics, computational chemistry [8] and its native modeling of many scientific phenomena [3]. Among different classes of quantum algorithms, Variational Quantum Algorithms [2] are the most promising for achieving the so-called quantum advantage.

However, due to the limited number of resources available and the high noise in their results, quantum processing units (QPUs) are combined with classic architectures, defining hybrid quantum systems and algorithms [18]. In this work, we focus on their integration into modern distributed systems.

4 IMPLICATIONS TO DISTRIBUTED SYSTEMS

4.1 Hyper Heterogeneity and Edge AI

State of the art in heterogeneous distributed systems is well represented by Edge AI [6], which includes different components (sensors, edge nodes, cloud nodes) with specialized tasks. Examples can be found in traffic safety [11], environmental monitoring [10] and smart agriculture [7]. The advent of Post-Moore's law computing is going to bring a disruptive change to distributed systems, where Edge AI is going to play a central role in the integration, acting as

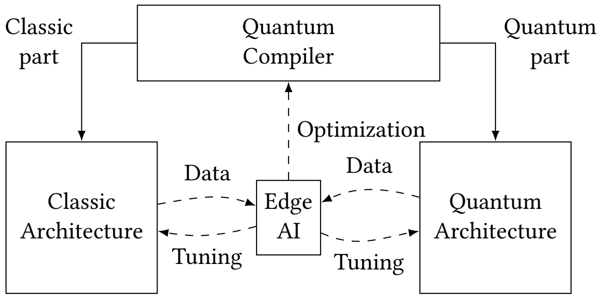

Figure 2: The Quantum Edge.

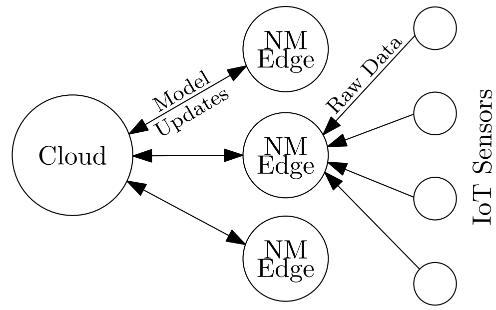

Figure 3: The Neuromorphic Edge.

a control layer between classic and different non-Von Neumann's architectures. Typical actions could be

- Collection and processing of execution data;
- ML-based data analytics;
- Tuning of execution on both sides,

depending on the target architectures. We describe possible scenarios for quantum and neuromorphic computing in the next sections.

4.1.1 The Quantum Edge. We expect that in the future Quantum computing will be used to execute highly specialized scientific computing tasks. Quantum machines need very specialized data centers to be maintained, thus **Quantum Edge** represents the layer connecting user devices with specialized Quantum data centers.

Quantum-Edge is summarized in Figure 2. As already discussed, only very specific parts of the code of an application are suitable for the execution on the Quantum machine. Currently, two types of algorithms, namely Variational Quantum Linear Solver (VQLS), and Variational Quantum Eigenvalue (VQE) algorithms are used to solve large integer programming problems and Eigenvalue calculations. The main idea of VQAs is to minimize a cost function C representing a specific property of the physical system (e.g., the ground state of a Hamiltonian, H_G). The state of the physical system is modeled by a Parametrized Quantum Circuit (PQC), which is a quantum circuit whose state is determined by a set of input parameters Θ. VQAs goal is then finding Θ^* minimizing the value of $C(\Theta)$.

Assuming a large scale application like in the area of scientific computing, in the first step the transpiler generates (i) the quantum part, namely, the parametrized quantum circuit representing the state of the physical system, and (ii) the classic part, which includes the codes to prepare the quantum state and optimize input Θ parameters [2]. In order to control and improve execution both on classic and quantum architecture, we use an intermediate Edge layer which is responsible to collect data about the execution of both architectures and in particular to map the code written for traditional von Neumann architecture onto Quantum architectures, which is a highly complex, iterative, and interactive process. Data are then processed at the edge layer to reduce execution latency and optimize the quality of the execution on both architectures. Data collected are used as input for different algorithms for (*i*) optimization of quantum execution hyperparameters, (*ii*) tuning of classic execution, and (*iii*) design of compiler optimization techniques to improve quantum compilation tasks, i.e., mapping of a logical circuit to the quantum machine topology.

4.1.2 The Neuromorphic Edge. Opposite to Quantum computing, Neuromorphic computing can be used to process huge amounts of raw data with much higher efficiency than traditional architectures. Thus, we define the role of the neuromorphic Edge as a data processing unit between the user devices and large-scale data centers. Figure 3 summarizes the **Neuromorphic Edge**. In order to improve the fan-in (the maximum number of input signals) and data throughput, we propose integrating neuromorphic hardware to the edge servers, denoted *NM Edge* [9]. High parallelism enabled by the neuromorphic hardware removes the computational power imbalance between the cloud and edge resources and allows a higher number of data inputs to be processed concurrently at the edge servers. Since neuromorphic hardware is able to handle analog data inputs, a further analog-digital conversion step is eliminated.

Moreover, event-driven neuromorphic hardware only consumes energy when input spikes are present. This translates to huge energy savings in use cases such as anomaly detection, since events of interest are relatively rare in comparison to expected behavior. Finally, the neuromorphic edge architecture reduces the pressure on the wide-area network (in terms of bandwidth and throughput) since most of the computation based on raw IoT data is handled at the NM Edge servers. Data transmission to the Cloud is only needed to update the AI models and has a relatively smaller volume.

5 USE CASES

In the previous section, we discussed the potential of Neuromorphic and Quantum Edge. In this section, we discuss the first concrete use cases where both Neuromorphic and Quantum Edge can bring significant benefits to geographically distributed applications.

5.1 Quantum Edge: Molecular Dynamics Simulations

Molecular dynamics (MD) simulation is used to compute the atomic states of an evolving molecular system over time by observing microscopic interactions between atoms. Typical computations involved in this model are to extract positions of α-Carbon (Cα) backbone atoms, which provides an indication of modifications of the molecular system. This is done by calculating Euclidean distance between two amino-acid segments, I and J, between C_α atoms i and j, d_{ij}. Results of computation of Euclidean distance will constitute a symmetric bipartite matrix, whose maximum eigenvalue allows to discover the changes in the molecular systems.

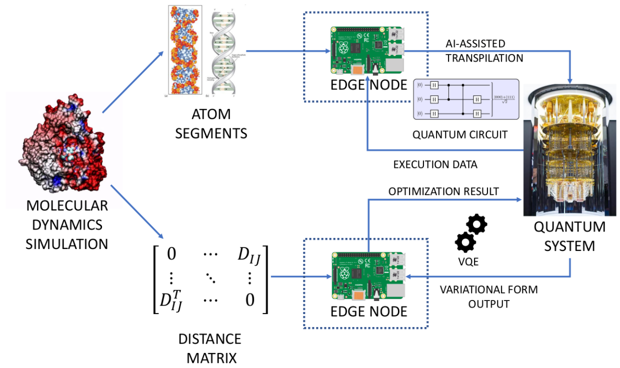

Figure 4: MD Simulation on the Quantum Edge

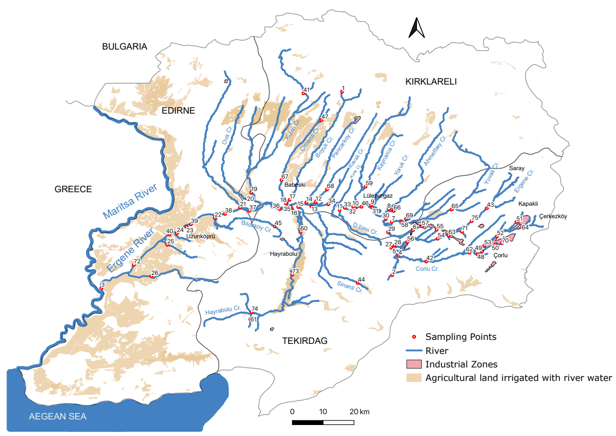

Figure 5: Planned monitoring stations on the Ergene watershed. (The image is a courtesy of TUBITAK project 115Y064.)

Typical computations that could be improved by quantum computing are (i) computation of the Euclidean distance, for which C-SWAP test [23] could be used, and (ii) calculation of maximum eigenvalue, for which the Variational Quantum Eigenvalue (VQE) [2] can be used. The advantage of the C-SWAP test is that it has a $O(\log n)$ complexity in comparison with the $O(n)$ complexity of Euclidean distance, while the use of VQE allows exploiting potentialities of hybrid quantum systems.

Edge computing could play a fundamental role in the integration of quantum computing hardware within classic HPC infrastructure, as described in Figure 4. The use of Edge in this context could improve quantum execution by improving circuit transpilation using ML/data-driven techniques, which will use as input the data about quantum execution which are processed at the Edge. Concerning VQE execution, employing Edge in the continuous loop between classic and quantum hardware could reduce communication latency between the two architectures.

5.2 Neuromorphic Edge: Environmental Monitoring

Monitoring and real-time decision-making is an essential aspect of future environmental protection, and sustainability goals [21]. Pollution monitoring (air and water quality, etc.) and disaster warning systems (seismic or volcanic activity, avalanches, etc.) are the two major areas in which IoT- and AI-based systems are being developed. Such systems are typically deployed in rural areas and characterized by intermittent network connectivity and limited or no access to conventional electricity utilities. The neuromorphic edge provides powerful computational resources in close proximity to environmental sensors and therefore eliminates the need for streaming big data transmission to remote data centers under unreliable connections. Moreover, reduced energy consumption allows use of limited energy sources, including energy harvesting systems.

One of the environmental monitoring applications is the SWAIN project[2], which aims to detect and locate micropollutant sources along European rivers in real-time. One of the use cases of the project is the Ergene watershed in northwestern Turkey as shown in Figure 5. Real-time decision-making is vital in this use case because the industrial zones highlighted in red in the upstream are potential micropollutant sources, whereas agricultural areas highlighted in light brown in the downstream use river water for irrigation. An effective early-warning system can prevent the exposure of crops to micropollutants. The river and pollutant transmission models in the SWAIN project are designed as Graph Neural Networks, which can run natively on the neuromorphic hardware at the edge nodes.

6 CONCLUSION AND OUTLOOK

In this work, we give a glance into the future of distributed systems. First, we identify the limit of Moore's law as an obstacle to further development of distributed systems and discuss possibilities to augment current distributed systems with non-Von Neumann's architecture to pave the way to hyper-heterogeneity in the so-called post-Moore's law era. We identify Neuromorphic and Quantum Architectures as the most promising approaches to accelerate computation in future distributed systems, and describe two possible use cases where neuromorphic and quantum computing could be employed to improve performance currently provided by distributed systems by utilizing the eoncept of Edge computing.

However, the implications of the hyper heterogeneity with Quantum and Neuromorphic Edge will cause significant challenges for the general software development process and the education of computer scientists as discussed next.

6.1 Software Development Processes

The development of quantum programs requires a strong paradigm switch from the design of programs (loops, conditions, sequences) to the design of quantum circuits (qubits, gates, entanglement). Also, it is of capital importance to train computer scientists in the non-determinism of quantum programming: indeed, while in classical programming we perform a single execution, which will give as output the solution of our problem, quantum circuits can provide different results at each execution, since each measurement

[2] http://swain-project.eu/

changes the state of the system. The consequences of this change of mindset are twofold: first of all, while in classic programming our goal is to design algorithms that for each input provide always the same output, in quantum results our goal is to tune the probability distribution of the output in a way that guarantees that the solution to our problem will be the most frequent over a given amount of measurements; second, the necessity of continuous executions implies that also the way we measure the performance of quantum execution changes, having to take into account the intrinsic non-determinism of quantum mechanics.

Concerning neuromorphic computing, creating a spiking neural network (SNN) is usually needed to program a neuromorphic computer [17], which differs significantly from the widely used procedural, object-oriented, and functional programming languages. Although it is expected that high-level languages that are compiled into SNNs would be developed, initial programming efforts will probably entail manual graph creation. Therefore, graph-based programming languages (e.g., [14]) might gain more importance.

6.2 Education

Integration of non-Von Neumann architectures might require the inclusion of new disciplines in computer science curricula. For quantum computing, knowledge about quantum mechanics and the principles regulating quantum physics could be beneficial in the design of quantum circuits. Also, given the non-deterministic nature of quantum computation, improving students' backgrounds in statistics and probability would definitely reduce entry barriers in quantum computing for classical computer scientists.

Similarly, programming neuromorphic hardware requires a graph theoretical and ML background which could gain more importance in computer science curricula. SNNs differ from traditional artificial neural networks as they run asynchronously and are based on analog signals. Currently, asynchronous and analog systems do not get much attention in computer science education.

ACKNOWLEDGEMENTS

This work has been partially funded through the Rucon project (Runtime Control in Multi Clouds), Austrian Science Fund (FWF): Y904-N31 START-Programm 2015 and by the CHIST-ERA grant CHIST-ERA-19-CES-005, by the Austrian Science Fund (FWF): I 5201-N. We would like to thank Ewa Deelman and Tu Mai Anh Do (ISI/USC) for the discussions on molecular dynamics applications. We acknowledge the use of IBM Quantum services for this work. The views expressed are those of the authors, and do not reflect the official policy or position of IBM or the IBM Quantum team.

REFERENCES

[1] Bush, V., et al. As we may think. *The atlantic monthly 176*, 1 (1945), 101–108.
[2] Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S. C., Endo, S., Fujii, K., McClean, J. R., Mitarai, K., Yuan, X., Cincio, L., et al. Variational quantum algorithms. *Nature Reviews Physics 3*, 9 (2021), 625–644.
[3] Chang, W.-L., Chen, J.-C., Chung, W.-Y., Hsiao, C.-Y., Wong, R., and Vasilakos, A. V. Quantum speedup and mathematical solutions of implementing bio-molecular solutions for the independent set problem on ibm quantum computers. *IEEE Transactions on NanoBioscience 20*, 3 (2021), 354–376.
[4] Davies, M., Srinivasa, N., Lin, T.-H., Chinya, G., Cao, Y., Choday, S. H., Dimou, G., Joshi, P., Imam, N., Jain, S., Liao, Y., Lin, C.-K., Lines, A., Liu, R., Mathaikutty, D., McCoy, S., Paul, A., Tse, J., Venkataramanan, G., Weng, Y.-H., Wild, A., Yang, Y., and Wang, H. Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro 38*, 1 (2018), 82–99.
[5] Dennard, R., Gaensslen, F., Yu, H.-N., Rideout, V., Bassous, E., and LeBlanc, A. Design of ion-implanted mosfet's with very small physical dimensions. *IEEE Journal of Solid-State Circuits 9*, 5 (1974), 256–268.
[6] Ding, A. Y., Peltonen, E., Meuser, T., Aral, A., Becker, C., Dustdar, S., Hiessl, T., Kranzlmüller, D., Liyanage, M., Maghsudi, S., Mohan, N., Ott, J., Rellermeyer, J. S., Schulte, S., Schulzrinne, H., Solmaz, G., Tarkoma, S., Varghese, B., and Wolf, L. C. Roadmap for edge AI: a dagstuhl perspective. *Comput. Commun. Rev. 52*, 1 (2022), 28–33.
[7] Friha, O., Ferrag, M. A., Shu, L., Maglaras, L. A., and Wang, X. Internet of things for the future of smart agriculture: A comprehensive survey of emerging technologies. *IEEE CAA J. Autom. Sinica 8*, 4 (2021), 718–752.
[8] Gill, S. S., Kumar, A., Singh, H., Singh, M., Kaur, K., Usman, M., and Buyya, R. Quantum computing: A taxonomy, systematic review and future directions. *Software: Practice and Experience 52*, 1 (2022), 66–114.
[9] INTEL. Beyond today's ai new algorithmic approaches emulate the human brain's interactions with the world. https://www.intel.de/content/www/de/de/research/neuromorphic-computing.html.
[10] Li, W., Shen, G., Zhang, J., Liu, D., and Choi, C. A lorawan monitoring system for large buildings based on embedded edge computing in indoor environment. *Concurrency and Computation: Practice and Experience* (2021), e6306.
[11] Lujic, I., Maio, V. D., Venugopal, S., and Brandic, I. SEA-LEAP: self-adaptive and locality-aware edge analytics placement. *IEEE Trans. Serv. Comput. 15*, 2 (2022), 602–613.
[12] Mead, C. Neuromorphic electronic systems. *Proceedings of the IEEE 78*, 10 (1990), 1629–1636.
[13] Moore, G. E. Cramming more components onto integrated circuits, reprinted from electronics, volume 38, number 8, april 19, 1965, pp.114 ff. *IEEE Solid-State Circuits Society Newsletter 11*, 3 (2006), 33–35.
[14] Plump, D. The graph programming language gp. In *International Conference on Algebraic Informatics* (2009), Springer, pp. 99–122.
[15] Rønnow, T. F., Wang, Z., Job, J., Boixo, S., Isakov, S. V., Wecker, D., Martinis, J. M., Lidar, D. A., and Troyer, M. Defining and detecting quantum speedup. *science 345*, 6195 (2014), 420–424.
[16] Schuller, I. K., Stevens, R., Pino, R., and Pechan, M. Neuromorphic computing–from materials research to systems architecture roundtable. Tech. rep., US Department of Energy, Office of Science, 2015.
[17] Schuman, C. D., Kulkarni, S. R., Parsa, M., Mitchell, J. P., Date, P., and Kay, B. Opportunities for neuromorphic computing algorithms and applications. *Nature Computational Science 2* (2022), 10–19.
[18] Stein, S. A., L'Abbate, R., Mu, W., Liu, Y., Baheri, B., Mao, Y., Qiang, G., Li, A., and Fang, B. A hybrid system for learning classical data in quantum states. In *2021 IEEE International Performance, Computing, and Communications Conference (IPCCC)* (2021), IEEE, pp. 1–7.
[19] Sun, B., Guo, T., Zhou, G., Ranjan, S., Jiao, Y., Wei, L., Zhou, Y. N., and Wu, Y. A. Synaptic devices based neuromorphic computing applications in artificial intelligence. *Materials Today Physics 18* (2021), 100393.
[20] Taylor, M. B. The evolution of bitcoin hardware. *Computer 50*, 9 (2017), 58–66.
[21] UN General Assembly. Transforming our world: the 2030 agenda for sustainable development. https://undocs.org/en/A/RES/70/1, October 2015.
[22] Welivita, A., Perera, I., Meedeniya, D., Wickramarachchi, A., and Mallawaarachchi, V. Managing complex workflows in bioinformatics: an interactive toolkit with gpu acceleration. *IEEE transactions on nanobioscience 17*, 3 (2018), 199–208.
[23] Wiebe, N., Kapoor, A., and Svore, K. M. Quantum algorithms for nearest-neighbor methods for supervised and unsupervised learning. *Quantum Info. Comput. 15*, 3–4 (2015), 316–356.
[24] Yang, K., Chen, Y.-F., Roumpos, G., Colby, C., and Anderson, J. High performance monte carlo simulation of ising model on tpu clusters. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (2019), pp. 1–15.
[25] Yeric, G. Moore's law at 50: Are we planning for retirement? In *2015 IEEE International Electron Devices Meeting (IEDM)* (2015), pp. 1.1.1–1.1.8.