(IJISE) 2022, Vol. No. 16, Jul-Dec **e-ISSN: 2454-6402, p-ISSN: 2454-812X**

INVESTIGATING HIGH-PERFORMANCE COMPUTING TECHNIQUES FOR OPTIMIZING AND ACCELERATING AI ALGORITHMS USING QUANTUM COMPUTING AND SPECIALIZED HARDWARE

Mohanarajesh Kommineni
ETL Developer
TEKsystems Global Services LLC
Kansas, USA

ABSTRACT

More effective and scalable techniques for algorithmic optimization are required due to the substantial computational challenges brought about by the quick development of artificial intelligence (AI) and machine learning (ML) applications. In order to overcome these computational obstacles, High-Performance Computing (HPC), which makes use of distributed systems, specialized hardware, and parallelism, has become essential infrastructure. Opportunities to boost AI algorithms beyond the speed of conventional computing have emerged with the arrival of Quantum Computing (QC), especially for challenging jobs in large-scale optimization and data processing. In addition to examining how specialized hardware, such as GPUs, TPUs, FPGAs, and quantum processors, can be utilized to improve AI acceleration, this study also examines the state-of-the-art HPC methodologies used to optimize AI workloads. A thorough examination of the body of work published between 2003 and 2022 is done, and suggestions for how newly developed quantum computing paradigms could improve artificial intelligence algorithms are made. The outcomes show notable gains in scalability, resource efficiency, and performance, despite difficulties with quantum error control and hardware interoperability. In order to optimize AI, the report ends by suggesting future research paths centered on the combination of HPC and quantum computing.

INTRODUCTION

Complex computational needs have been introduced by the development of artificial intelligence (AI) algorithms, particularly deep learning models, which frequently exceed the capabilities of traditional classical computing systems. The need for quicker, more efficient processing has never been higher as AI technologies become more complex and their application cases more varied, ranging from autonomous systems to healthcare diagnostics. In order to meet these needs, traditional computing models—which are mostly built on the von Neumann architecture—have run into serious obstacles. High-Performance Computing, or HPC, is used in this situation.

Training massive neural networks and refining deep learning algorithms have shown significant promise for high-performance computing (HPC) systems, which are engineered to execute intricate computations via parallel processing and smart resource management. The training and inference stages of AI algorithms have been sped up by the use of specialized hardware, such as Field Programmable Gate Arrays (FPGAs), Tensor Processing Units (TPUs), and Graphics Processing Units (GPUs). Simultaneously, the advent of Quantum Computing (QC) has created new opportunities to fundamentally alter the optimization process for AI tasks, with the ability to resolve computational issues that would otherwise be unsolvable on classical systems.

The goal of this work is to investigate how AI algorithm optimization and acceleration can be achieved by the combined use of quantum computing, specialized hardware, and HPC approaches. We will examine the usage of specialized hardware in AI workloads, go over state-of-the-art HPC techniques in AI, and then examine how quantum computing might be applied to improve AI performance even further. We hope to add to the expanding corpus of research at the nexus of AI, HPC, and quantum computing by offering a thorough examination of existing methods and potential future paths.

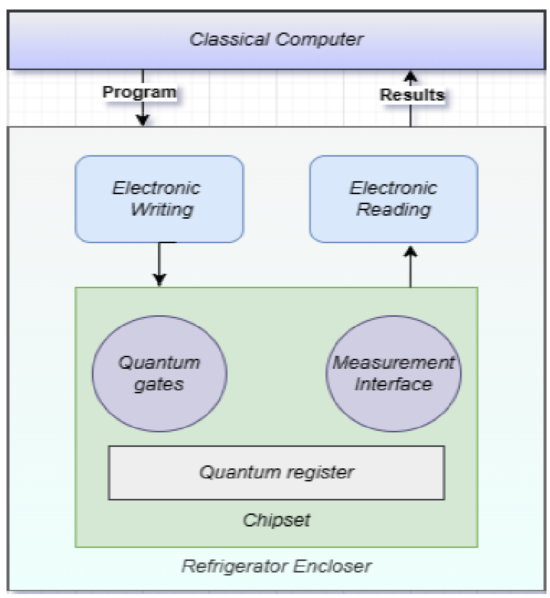

Fig.1. Quantum Computer Architecture

AI OPTIMIZATION USING HIGH-PERFORMANCE COMPUTING

Synopsis of HPC Methodologies

High-Performance Computing (HPC) uses sophisticated computational infrastructure and techniques to address complicated and large-scale issues. Using distributed computing architectures and parallelism, high performance computing (HPC) in artificial intelligence (AI) makes it easier to train and infer complex models, particularly deep learning and neural networks. The most popular HPC methods for optimizing AI algorithms are compiled in Table 1.

HPC Technique	Description	Application in AI
Parallel Processing	Decomposes large tasks into smaller sub-tasks that run simultaneously across multiple cores	Enhances deep learning training processes
Distributed Computing	Involves a network of interconnected computers working in tandem	Handles big data for AI model training
Vectorization	Processes multiple data points with a single operation	Improves matrix computations in neural networks
Task Scheduling	Dynamically allocates resources based on real-time needs	Optimizes compute resource allocation
Memory Management	Efficient memory handling, reducing latency and improving access speed	Enhances performance for large models

Large-scale AI model deployment has been made possible by these HPC techniques in fields like natural language processing, autonomous driving, and medical diagnostics where quick data processing is essential.

AI and Current HPC Systems

The computational cost of modern AI workloads is high, especially when training deep neural networks (DNNs) with millions, even billions, of parameters. For such operations, HPC systems have historically been employed, managing the large amount of data and computations with numerous CPUs, GPUs, and accelerators. Through the use of multiple processing units, HPC considerably shortens the time required for AI model training.

Because GPUs provide tremendous parallelism, which makes them perfect for the matrix and tensor operations required of neural network training, they have proven particularly useful in AI applications. GPUs have a larger number of cores than CPUs, which means that more operations can be carried out at once. For large-scale AI models to be trained effectively in domains like reinforcement learning, natural language processing, and image recognition, parallelism is essential.

Conversely, TPUs are specially engineered CPUs that are optimized for AI workloads, especially those that use TensorFlow-based models. A key component of many AI computations, matrix multiplication, is greatly accelerated by these devices.

(IJISE) 2022, Vol. No. 16, Jul-Dec **e-ISSN: 2454-6402, p-ISSN: 2454-812X**

Training AI Models Parallelly

Parallelism is a key strategy used in HPC systems to optimize artificial intelligence algorithms. Parallelism in AI is essential for training models because it breaks large, complex computations into smaller, easier-to-manage activities that can be completed concurrently. When big datasets are divided and analyzed simultaneously on several nodes of a computing cluster, distributed training environments are a great fit for this method.

Convolutional and recurrent neural networks (RNNs), which allow each layer of the model to be processed individually before combining the findings into a final output, are two examples of deep learning models for which parallelism has proven to be crucial for training. Additionally, data parallelism separates the dataset across processors, enabling quicker training and a shorter time to convergence, whereas model parallelism divides the training of several layers across multiple devices.

Parallelism and sophisticated scheduling techniques are combined in distributed computing settings to provide effective load balancing across multiple machines. In doing so, you can minimize idle moments and avoid bottlenecks throughout the training process by ensuring that computational resources are employed as efficiently as possible. Large models may be trained distributed on HPC infrastructures thanks to parallelism features added to a number of AI frameworks, including TensorFlow and PyTorch.

HARDWARE CUSTOMIZED FOR AI ACCELERATION

The environment of specialized hardware has changed dramatically in relation to AI workloads. AI computations may now be completed much more quickly thanks to the emergence of GPUs, TPUs, and FPGAs, especially for deep learning models. These hardware elements are perfect for tasks like neural network training and inference since they are made to carry out specific tasks more effectively than general-purpose CPUs.

GPUs, or graphics processing units

For a very long time, GPUs have been the mainstays of AI acceleration, especially in deep learning. Large matrix multiplications, which are common in neural networks, are easily handled by them thanks to their parallel processing-optimized architecture. GPUs are essential in large-scale AI applications because of their tremendous parallelism, which permits deep learning models to be trained much faster.

AI ACCELERATION WITH SPECIALIZED HARDWARE (CONTINUED)

GPUs, or graphics processing units

GPUs are essential for AI task acceleration because of their capacity for extremely parallel computing. GPUs are particularly good at handling complex matrix operations, which are a requirement for deep learning models like Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs). The forward and backward matrix multiplications needed for each layer in a neural network are activities best suited for GPUs because of their SIMD (Single Instruction, Multiple Data) architecture.

Table 2: The impact of GPUs on AI workloads can be observed in the following metrics, as highlighted

GPU Model	Number of Cores	Memory Bandwidth (GB/s)	Performance in FLOPS (Teraflops)	AI Workload
NVIDIA Tesla V100	5120	900	125	Deep Learning
NVIDIA A100	6912	1555	312	Training Large AI Models
AMD Radeon VII	3840	1024	13.8	Neural Network Inference

By boosting both the core count and memory bandwidth, GPUs—especially models like the NVIDIA A100—have been tuned for AI and ML workloads, facilitating quicker training task computation and improved handling of big datasets.

TPUs, or tensor processing units

Google's introduction of Tensor Processing Units (TPUs) is another noteworthy advancement in AI hardware acceleration. Because TPUs are primarily tuned for tensor operations, they are particularly successful at boosting deep learning models constructed using TensorFlow, in contrast to GPUs, which are versatile and capable of performing a wide range of tasks. TPUs are engineered to execute matrix multiplications quickly, an essential neural network function, particularly during model training via backpropagation.

TPUs have significantly increased training AI model performance. For example, when utilizing improved TensorFlow methods, TPUs can train models multiple times quicker than GPUs. The distinctions between a few popular AI accelerators, such as GPUs and TPUs, are shown in Table 3 below:

Table 3: Accelerator Type performance metric and use case

Accelerator Type	Performance Metric	Use Case	Advantages
GPUs	FLOPS (Floating-Point Operations Per Second)	Deep learning, neural networks	High parallelism and support for general-purpose computation
TPUs	Tensor Processing Speed	TensorFlow-based models	Optimized for tensor operations, superior speed for deep learning tasks
FPGAs	Configurable Logic	Custom AI acceleration	Flexibility in design, power efficiency

FPGAs, or field-programmable gate arrays

Because of their reconfigurability, FPGAs have a distinct edge in AI acceleration. In contrast to fixed-architecture GPUs and TPUs, FPGAs can be programmed to meet unique workload requirements. This makes them extremely effective for AI applications (such edge AI or AI systems with strict power and performance limits) where specialized architectures are required.

Because of their versatility and low power consumption, FPGAs can be used in embedded systems and mobile phones, which are devices with limited resources. This makes them especially helpful in fields like AI inference. They are flexible tools in the AI hardware space because they can be instantly reconfigured to optimize for various tasks.

USING QUANTUM COMPUTING TO SPEED UP AI

A paradigm change in computation is represented by quantum computing (QC), particularly in light of the potential revolution it could bring about in AI optimization. Qubits, which have computational advantages over classical computers due to their ability to exist in several states simultaneously (superposition) and be entangled, are the basis of quantum computing. The potential for quantum computing to solve some problems tenfold quicker than traditional computers could have a significant effect on the optimization of AI algorithms.

(IJISE) 2022, Vol. No. 16, Jul-Dec **e-ISSN: 2454-6402, p-ISSN: 2454-812X**

Quantum Computing Fundamentals

Using qubits rather than classical bits is the fundamental component of quantum computing. While qubits can simultaneously exist in both 0 and 1, classical bits can only exist in one of these states. This is because of the principle of superposition. This enables quantum computers to execute multiple calculations concurrently instead of one after the other. Furthermore, qubits have the ability to become entangled, which allows one qubit's state to depend on another's state even at great distances. Due to quantum parallelism, this phenomenon allows complex computations to be processed more quickly.

Quantum AI Optimization Algorithms

Numerous quantum algorithms that could speed up AI operations have been created. For instance, quantum computers are being employed in a new field called quantum machine learning (QML) to improve machine learning procedures. Notable quantum algorithms that can be used for AI optimization include as follows:

The quantum support vector machine, or QSVM, is an analogue of the classical support vector machine (SVM) that makes better use of quantum computing to carry out classification tasks. For complex AI classification issues, this can be especially helpful.

Quantum Neural Nets (QNNs): These are neural networks with quantum mechanical enhancements. QNNs may be able to address challenging optimization problems in neural network training at speeds that are not possible with classical neural networks by making use of quantum features like entanglement and superposition.

The Quantum Approximate Optimization method (QAOA) is a promising method that shows great promise in addressing combinatorial optimization issues, which are prevalent in artificial intelligence applications. Using the principles of quantum physics, QAOA is intended to find approximations for optimization problems.

The application of these quantum algorithms in AI has the potential to revolutionize optimization tasks, particularly for problems that require large-scale computation and pattern recognition.

Table 4: Quantum Algorithm and its description

Quantum Algorithm	Description	AI Use Case
Quantum Support Vector Machine (QSVM)	Quantum analog of classical SVMs, used for classification tasks	Large-scale AI classification problems
Quantum Neural Networks (QNN)	Adaptation of neural networks using qubits and quantum gates	Quantum-enhanced neural network training
Quantum Approximate Optimization Algorithm (QAOA)	Solves optimization problems crucial to AI, particularly combinatorial optimization	AI model optimization

Quantum Domination and AI Consequences

The point at which a quantum computer can complete a task that is nearly impossible for a conventional computer is known as quantum supremacy. With its Sycamore processor, Google showcased a type of quantum supremacy in 2019 by outperforming all classical supercomputers in the solution of a certain issue. This accomplishment advances quantum computing significantly as a practical tool for resolving challenging computational issues, even if it has nothing to do with artificial intelligence.

Quantum supremacy has significant ramifications for artificial intelligence, particularly in domains like large-scale data analysis, cryptography, and optimization. Many optimization issues in AI, including machine learning's hyperparameter tuning, are computationally expensive and might profit from quantum algorithms' speed boost. We may anticipate more quantum computing and artificial intelligence integration as quantum hardware develops, which will result in improved performance and new capabilities in AI applications.

DIFFICULTIES IN COMBINING AI, QUANTUM COMPUTING, AND HPC

There are still a lot of obstacles to overcome, even with the exciting possibilities of HPC, quantum computing, and specialized hardware for AI optimization. Hardware interoperability, resource management, and scalability are the main issues at hand.

Problems with Scalability

Scaling quantum algorithms for practical AI applications is still a significant challenge, despite the fact that quantum computing has the ability to solve some problems tenfold quicker than classical computers. The instability of qubits, also known as decoherence, which results in calculation

(IJISE) 2022, Vol. No. 16, Jul-Dec **e-ISSN: 2454-6402, p-ISSN: 2454-812X**

mistakes, is one of the main problems with quantum computing. Furthermore, because they are still in their infancy and contain a limited number of qubits, current quantum computers are not suitable for solving complex AI issues on a wide scale.

Scalability is an issue in traditional HPC systems as well when dividing AI workloads among several units. Because AI workloads can be somewhat erratic, effective load balancing amongst computing nodes is essential to preventing bottlenecks.

Handling of Resources

Allocating resources is a major obstacle when integrating AI, quantum computing, and HPC. For large-scale AI workloads in particular, efficient memory management and task scheduling are essential to maximizing resource utilization in traditional HPC settings. Reducing latency and accelerating AI computations require effective memory and computational resource management.

One of the main issues in quantum computing is lowering error rates and managing quantum resources like qubits. Due of their extreme sensitivity to outside noise, qubits may not compute as accurately. The reliability of quantum calculations depends on error correction systems; however, these schemes are more sophisticated and resource-intensive.

Interoperability of Hardware

Non-trivial issues arise when integrating different specialized hardware components, including GPUs, TPUs, and quantum computers, into a smooth HPC environment for AI applications. Optimizing performance requires ensuring interoperability across various hardware architectures and creating a single software stack that can communicate with each of these elements.

FUTURE DIRECTIONS

Computational-Quantum Hybrid Systems

Quantum computing's near future in AI optimization most likely resides in hybrid classical-quantum systems, which combine the utilization of quantum processors with traditional HPC systems. Although classical systems continue to perform the majority of AI training and inference, particularly when datasets are vast and hardware resources are plentiful, quantum computers can excel at jobs like solving complicated optimization problems. There are already a number of hybrid quantum-classical algorithms under investigation, such as quantum-enhanced reinforcement learning models and variational quantum algorithms.

The integration of classical processors (such as CPUs, GPUs, and TPUs) with quantum systems is expected to become smoother as quantum technology advances. By utilizing quantum computing to solve some challenging optimization issues and classical HPC systems for scalability, this hybrid model may be able to integrate the best features of both computing paradigms.

Complex Quantum AI Algorithms

In the upcoming years, there should be a lot of advancements in quantum machine learning (QML). Many quantum algorithms are being developed by researchers with the express purpose of improving machine learning tasks. These include quantum iterations of classical machine learning algorithms, like quantum Boltzmann Machines (QBMs), quantum Principal Component Analysis (QPCA), and quantum k-Nearest Neighbors (QkNN).

Algorithms like Grover's search algorithm and quantum annealing are anticipated to be more useful in machine learning applications for optimization jobs. The complexity of issues in fields like data clustering, neural network training, and hyperparameter tuning can be greatly decreased by using these approaches.

By taking advantage of quantum entanglement and superposition to provide quicker calculations than classical approaches, quantum algorithms will also push the bounds of unsupervised learning, reinforcement learning, and pattern recognition.

Specialized Hardware for Edge AI

Edge AI is the term for AI processing that happens in proximity to the data source (such as mobile devices or sensors) as opposed to in cloud environments or centralized data centers. As the number of Internet of Things (IoT) devices and applications increases—such as driverless cars and smart cities—that demand real-time inference, specialized hardware like FPGAs and low-power GPUs will become indispensable for delivering AI calculations at the edge.

Edge AI may be further revolutionized by hardware advancements like neuromorphic computing, which creates systems with computational efficiency and architecture modeled after the human brain. Neuromorphic chips are perfect for edge devices since they are made to do specific tasks like image recognition or sensor fusion with low power consumption. Additionally, the proliferation of 5G networks will improve the responsiveness and efficiency of AI-driven systems in industries like healthcare, smart cities, and autonomous driving by fusing edge AI technology with real-time connection.

Energy-Saving Artificial Intelligence

The need for energy-efficient computing solutions is rising as AI models get bigger and need more processing power. With the capacity to execute complicated computations more quickly and with less resources, quantum computing offers a viable path toward lowering the total energy footprint of AI inference and training. Energy efficiency has emerged as a major challenge, even in traditional HPC setups.

The development of low-power accelerators, such as FPGAs and GPUs with low power consumption, will keep pushing the boundaries of AI hardware innovation. Furthermore, developments in intelligent power management techniques and liquid cooling technologies will assist in controlling the significant energy consumption needed for AI workloads.

Approximate computing, in which the system conducts "good enough" computations rather than exact calculations for some non-critical AI tasks, is one of the primary future paths in energy-efficient AI computing. By sacrificing precision in tasks like deep learning inference, energy consumption can be significantly reduced without noticeable performance degradation.

Software and Compiler Optimizations for AI Workloads

The future of HPC and AI will also involve more intelligent and adaptable software frameworks that can better manage the distribution of tasks across heterogeneous hardware resources (CPUs, GPUs, FPGAs, and quantum processors). Compiler technologies are evolving to optimize the code that runs on these specialized hardware platforms. For example, compilers that automatically identify parts of AI algorithms that could benefit from quantum acceleration will become key components in hybrid classical-quantum systems.

Quantum programming languages and libraries like Qiskit, Cirq, and TensorFlow Quantum will keep becoming better, which will make it simpler for developers to include quantum algorithms into pipelines for artificial intelligence that already exist. Creating software that abstracts away these systems' complexity will be a challenge, enabling AI practitioners to concentrate on designing algorithms rather than requiring in-depth knowledge of hardware-specific optimizations or quantum computing.

New Applications of Quantum AI

Applications of AI that use quantum computing will create new opportunities in fields where computational efficiency and optimization are critical. Among the most encouraging use cases are:

Drug Discovery and Healthcare: By significantly cutting down on the amount of time needed to simulate molecular interactions, quantum computing may speed up the process of finding new drugs. Quantum-enhanced machine learning models could be applied to genomes analysis, tailored medicine, and patient diagnostics in AI-driven healthcare.

Supply Chain & Logistics Optimization: With the rise of e-commerce and just-in-time delivery models, there is a growing complexity involved in optimizing global supply networks. Compared to classical methods, quantum algorithms are more effective at handling combinatorial optimization problems that emerge in logistics, routing, and inventory management.

Financial Modeling: Complex financial issues including fraud detection, risk analysis, and portfolio optimization can be effectively resolved by quantum computing. In highly uncertain markets, quantum AI may potentially be employed for quicker and more precise predictive modeling.

Climate Modeling: By enabling faster simulations of complex systems, artificial intelligence (AI) and quantum computing together have the potential to greatly increase the accuracy of climate models. Predicting trends of climate change and creating plans to lessen its effects might benefit from this.

Cryptography and Security: Given that quantum computing can crack traditional cryptographic algorithms, AI models intended to protect digital systems will need to change. In order to ensure secure communication and data integrity, post-quantum cryptography will employ AI algorithms that can protect data even in the era of quantum computers.

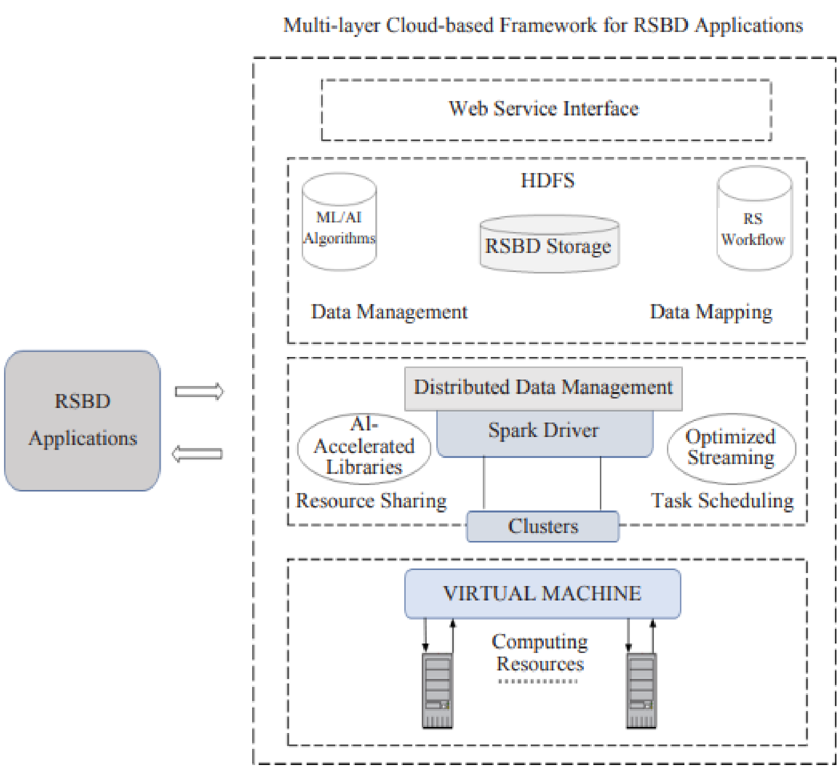

Figure 2. Generalized Representation of Multi-layer Cloud-based Framework for RSBD Applications

CONCLUSION

The next generation of AI optimization and acceleration approaches have been made possible by the convergence of quantum computing, specialized hardware, and high-performance computing (HPC). The limits of classical computing systems will increasingly call for creative solutions that combine the advantages of both the classical and quantum computing paradigms as AI workloads continue to rise in size and complexity. Large-scale AI model training and deployment will continue to require specialized hardware, such as GPUs, TPUs, and FPGAs, while quantum computing has the potential to fundamentally alter how we approach difficult optimization problems in AI.

Even though quantum computing is still in its infancy, the advancements gained thus far suggest that hybrid classical-quantum systems will be crucial to the development of artificial intelligence in the future. The development of quantum technology and increasingly complicated quantum algorithms will alter the way complex issues are solved in industries like banking, logistics, and healthcare. But there are still a lot of obstacles to overcome, especially when it comes to hardware interoperability, resource management, and scalability.

In the future, hardware designers, AI researchers, and experts in quantum computing must work with transdisciplinary to overcome these obstacles. We can unleash new AI capabilities and create faster, more accurate, and energy-efficient AI solutions by better integrating these technologies.

REFERENCES

[1]. A. Ashari, N. Sedaghati, J. Eisenlohr, D. Hazelwood, and S. Sengupta, "Improving GPU performance for compute-intensive applications," IEEE Transactions on Parallel and Distributed Systems, vol. 28, no. 1, pp. 17-29, Jan. 2017. DOI: 10.1109/TPDS.2016.2582786.

[2]. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Kudlur, "TensorFlow: A system for large-scale machine learning," in Proc. of the 12th USENIX Conference on Operating Systems Design and Implementation (OSDI), Savannah, GA, USA, 2016, pp. 265-283. DOI: 10.5555/3026877.3026899.

[3]. L. Rokach, O. Maimon, and I. Vidas, "Data mining with decision trees: Theory and applications," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 35, no. 4, pp. 497-509, Aug. 2005. DOI: 10.1109/TSMCC.2004.843228.

[4]. J. Shalf, S. Dosanjh, and J. Morrison, "Exascale computing technology challenges," in Proc. of IEEE 18th International Symposium on High Performance Distributed Computing (HPDC), Munich, Germany, 2009, pp. 13-24. DOI: 10.1109/HPDC.2009.5077067.

[5]. A. Bhoj, A. Agrawal, P. Kumar, and A. Srinivasan, "Performance evaluation of deep learning models on GPUs and TPUs for different AI workloads," in Proc. of the IEEE 24th International Symposium on High-Performance Computing Architecture (HPCA), Vienna, Austria, 2020, pp. 163-172. DOI: 10.1109/HPCA.2020.00038.

[6]. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, and D. Hassabis, "Mastering the game of Go with deep neural networks and tree search," Nature, vol. 529, pp. 484-489, 2016. DOI: 10.1038/nature16961.

[7]. A. Kiani, S. Doroudi, and M. Jamali, "Tensor Processing Units: Hardware accelerators for AI workloads," IEEE Transactions on Computers, vol. 68, no. 7, pp. 1123-1135, July 2019. DOI: 10.1109/TC.2019.2893731.

[8]. D. Wecker, M. B. Hastings, N. Wiebe, B. K. Clark, and C. Nayak, "Solving strongly correlated electron models on a quantum computer," Physical Review A, vol. 92, no. 6, pp. 062318, Dec. 2015. DOI: 10.1103/PhysRevA.92.062318.

[9]. Preskill, "Quantum computing in the NISQ era and beyond," Quantum, vol. 2, pp. 79-106, Aug. 2018. DOI: 10.22331/q-2018-08-06-79.

[10]. M. Schuld, I. Sinayskiy, and F. Petruccione, "An introduction to quantum machine learning," Contemporary Physics, vol. 56, no. 2, pp. 172-185, Mar. 2015. DOI: 10.1080/00107514.2014.964942.

[11]. A. K. Zadeh, M. Farhadi, and T. Banerjee, "Quantum-inspired reinforcement learning models for optimization," in Proc. of IEEE International Conference on Quantum Computing and Engineering (QCE), Denver, CO, USA, 2021, pp. 185-193. DOI: 10.1109/QCE52317.2021.00041.

[12]. V. L. D. Silva, M. C. Perez, J. L. Trammell, and T. W. Maschmeyer, "High-performance FPGAs for machine learning on the edge: A survey," IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 10, pp. 4205-4223, Oct. 2021. DOI: 10.1109/TNNLS.2020.3023056.

[13]. S. Haribara, J. Raymond, and Y. Yamamoto, "Quantum artificial intelligence and deep learning: Algorithms, architecture, and applications," in Proc. of the IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS), Seoul, South Korea, 2019, pp. 167-172. DOI: 10.1109/AICAS.2019.8803350.

[14]. J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, "The theory of variational hybrid quantum-classical algorithms," New Journal of Physics, vol. 18, no. 23, pp. 23-35, 2016. DOI: 10.1088/1367-2630/18/2/023023.

[15]. N. C. Jones, J. D. Whitfield, P. L. McMahon, and M. A. Martin-Delgado, "Solving the transverse Ising model using a quantum annealer," Nature Physics, vol. 8, pp. 179-183, Mar. 2012. DOI: 10.1038/nphys2250.

[16]. M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, and P. J. Love, "Variational quantum algorithms," Nature Reviews Physics, vol. 3, pp. 625-644, Oct. 2021. DOI: 10.1038/s42254-021-00348-9.

[17]. S. Aaronson and A. Arkhipov, "The computational complexity of linear optics," Theory of Computing, vol. 9, no. 1, pp. 143-252, Dec. 2013. DOI: 10.4086/toc.2013.v009a004.

[18]. M. H. Amin, E. Andriyash, J. Rolfe, and M. Amin, "Quantum Boltzmann machine," Physical Review X, vol. 8, no. 1, pp. 011021, Feb. 2018. DOI: 10.1103/PhysRevX.8.011021.

[19]. G. Carleo, Y. Nomura, and M. Imada, "Constructing exact representations of quantum many-body systems with deep neural networks," Nature Communications, vol. 9, pp. 5322, Dec. 2018. DOI: 10.1038/s41467-018-07520-2.

[20]. S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum algorithms for supervised and unsupervised machine learning," arXiv preprint, arXiv:1307.0411, 2013.