International Journal of Scientific Engineering and Research (IJSER)
ISSN (Online): 2347-3878
Impact Factor (2020): 6.733

An Examination of Different Kinds of Quantum Computing Technologies

Karan Chawla

Abstract: ***One of the main drivers for growing the field of quantum computation has been the prospect of creating a quantum computer capable of carrying out Shor's algorithm for huge numbers. Nonetheless, it is critical to recognise that quantum computers will probably only significantly speed up a small subset of issues if one wants to acquire a more comprehensive perspective on them. Building a system that can accommodate a lot of qubits while preserving stability and coherence is one of the biggest problems in quantum computing. Due to their extreme sensitivity to noise and mistakes, quantum systems are particularly susceptible to computing errors. Building a usable quantum computer requires error mitigation and correction, but the techniques for doing this are still in the early phases of research. In quantum computing, all processes and schemes must be reversible in accordance with the law of unitary development. The circuit model is taken into account in the NISQ framework, but there is also the one-way or measurement-based quantum computation approach, which is not reversible but is demonstrated to be comparable to the circuit model. Noisy suggests that the computing capacity of such quantum computers is constrained because of sufficiently high error and decoherence rates. While they are likely too huge to be replicated by traditional computers using brute force, Intermediate-Scale suggests that they are still too small to be mistake corrected, which also adds to the preceding argument about how noisy they are. P stands for the usage of perfect qubits with perfect quantum gates and no decoherence. The capability to control and protect our qubits in a quantum computer to the degree necessary to run such algorithms is commonly described as fault-tolerant quantum computation. This review paper examines the numerous quantum computing strategies, such as Noisy Intermediate Scale Computing (NISQ) , Perfect Intermediate Scale Computing (PISQ), and Fault Tolerant Intermediate Scale Computing (FTISQ), as well as potential future lines of inquiry.***

Keywords: Noisy Intermediate Scale Computing, Perfect Intermediate Scale Computing, Fault Tolerant Intermediate Scale Computing, Noisy, Shor's algorithm, quantum computation, qubits, error mitigation

1. Introduction

When it comes to processing, storing, and manipulating massive amounts of data and carrying out complex calculations that are beyond the capabilities of conventional computing systems and supercomputers, quantum computing is defined as a computational technology that makes use of the quantum mechanics principles of entanglement, superposition, and interference.

Throughout the past century, scientists have learned that physical rules at the subatomic scale do not apply and are very different from those we witness on a daily basis. This has brought the uncertainty factor to light. As a result, "quantum mechanics," which encoded the science of subatomic components, was developed. It established the basis for biology, chemistry, and physics.

Technologists wanted a tool to do computations while controlling uncertainty now that the uncertainty phenomena was obvious. Quantum computing was thus created. It is based on the fundamental laws that govern the subatomic world, where elementary particles can exist simultaneously in a variety of states and places. The method is applied in a quantum computing model to examine the quantum behavior of matter and energy. Quantum computing is anticipated to transform several industries over the next 10 years, including finance, health, machine learning, and artificial intelligence. The main force behind quantum computer developments is the money poured by investors, governments, and corporations in their pursuit of ultimate quantum supremacy.
The "National Quantum Initiative," which aims to expand the quantum computing area, was introduced by the US government in 2019. Also, the government allocated $1.2 billion to promote the quantum world. Similar to the United States, China is advancing its plans for quantum technology by spending $10 billion to create the "National Laboratory for Quantum Information Sciences."

Supercomputers can be used to address complex issues. Modern cybersecurity issues, optimisation issues, stock profile management, aeronautical issues, molecular research, and other issues are a few instances. Protein modeling is a different illustration. During the COVID-19 outbreak, the scientific community looked for a computer method that could mimic and deactivate a single protein in a shorter period of time. The world may have been spared from this worldwide health calamity if such a technology had been accessible.The use of energy is a current issue as well. Energy use has significantly increased as a result of the exponential growth in global population. This has produced the "energy source optimisation" problem, which is challenging for modern computers to solve. The rise of quantum computing has a silver lining in that it allows for the solution of such challenging challenges.

It has previously been shown that an enzyme known as "nitrogenase" enables our planet to create ammonia fertilizer at ordinary pressure and temperature. Nevertheless, the production of this enzyme involves a challenging catalytic process that is beyond the capabilities of current computers. Using the use of molecular modeling, the path taken by nitrogenase via about 1,000 carbon atoms is traced. As a result, it restricts the industrial production of nitrogenase, which has an impact on the total industrial output of fertilizers based on ammonia.

The development of molecular models of nitrogenase using quantum computers could be able to aid in this case. Computing may also be used to create compounds that are comparable to the enzyme and aid in the production of inexpensive and low-energy ammonia.

Ammonia-based fertilizers would be widely accessible and reasonably priced with quantum computing. Also, the method would lessen the strain on energy consumption that is often seen in the nitrogenase development process.

The ideal applications for classical computing are linear problems where sequential operations are the primary focus. These computer systems are based on the study of transformation characteristics and linear equations in linear mathematics.

Nature, however, is inherently non-linear and contains a little amount of uncertainty. Such nonlinear problems are difficult for classical systems to address. Quantum computers can, however, handle non-linear data. Examples of these nonlinear issues include the optimisation of traffic equilibrium and the likelihood of a lunar landing, among others.

A staggering quantity of data is produced every day in our big data age and digital age. With the advent of the internet of things, every wearable, gadget, IoT device, and sensor is connected to a computing network and adds to the data produced. Domo estimates that computing equipment produces 2.5 quintillion bytes of data every day.

By processing such a vast amount of data, modern computers and supercomputers are prone to mistakes, which negatively impacts performance. Moreover, traditional computers cannot handle computational tasks like analyzing the chemical impacts of medications. Instead, because they can handle large amounts of data more quickly, quantum computers are better suited for such jobs.

Noisy Intermediate Scale Computing

John Preskill first used the term Noisy Intermediate Scale Quantum (NISQ) computing in 2018 to describe how contemporary quantum computers are prone to high mistake rates and are constrained in size by the amount of logical qubits (or even physical qubits) in the system. This basically indicates that they are unreliable for doing broad calculations. Because of this inherent fallibility, some industry professionals foresee a "quantum winter." Some people think it will take decades for the industry to figure out how to get past the NISQ period, but more optimistic people in the sector think it will happen over the next few years.

Algorithms that were created with NISQ processors' limitations in mind are referred to as NISQ algorithms. The fact that they are intended to be implemented on NISQ devices soon, or within the next several years, is crucial. Moreover, this suggests that NISQ programmers want to employ as many qubits as are physically feasible. These algorithms do not specifically mention the lack of QEC, but they do claim to be relatively tolerable to computational noise through error mitigation.

While QEC is the process by which methods are developed to build in a certain form of redundancy in the computation by storing information over multiple qubits, error mitigation is the process by which users aim to build in mitigation strategies in their algorithms to account for the effects of computational noise. The final goal of NISQ algorithms is to use as few complex operations (such as multi-qubit controlled unitaries) and shallow-depth quantum circuits (currently, at most a few hundred gates in depth). The Shor and Grover algorithms do not fall within the umbrella of NISQ algorithms under this concept. This is due to the fact that both of them rely on Oracle Black Box Functions or Unitaries, which on a quantum computer need extremely lengthy and intricate circuits to execute [48]. Therefore, in these algorithms, the quantum advantage cannot be achieved with just a few noisy qubits. Additionally, these algorithms lack explicit noise tolerance because even minor errors and noise have a significant impact on the outcomes of such calculations. As time simulation to arbitrarily long times necessitates the use of quantum circuits with arbitrarily deep depths, full scale Trotterization does not fall under the category of NISQ algorithms.

What defines "short term" and what makes a "complex procedure," according to various scholars, will mean different things. It should be stressed, nonetheless, that the current experimental work serves as the empirical motivation for this operational formulation. The majority of NISQ algorithms are now being experimentally tested and operated on modern hardware. This contrasts with the majority of fault-tolerant algorithms, which, with the exception of simple instances (such as using Shor's method to identify prime factors for the number 15), cannot be executed on contemporary quantum computers. In fact, the majority of these algorithms openly indicate that they need a high level of fault tolerance in order to function.

The first class of NISQ algorithms is called the Variational Quantum Eigensolver. A leading approach for quantum chemistry on near-term quantum computers is the Variational Quantum Eigensolver (VQE). The ground state of a certain molecule is prepared by a quantum computer that has been educated using the Ritz variational principle. A molecular Hamiltonian and a parametrized circuit that prepares the molecule's quantum state are the inputs to the VQE algorithm. The expected value of the Hamiltonian calculated in the trial state is the definition of the cost function in VQE.

Iteratively minimizing the cost function results in the ground state of the target Hamiltonian. A classical optimizer uses a quantum computer to assess the cost function and determine its gradient at each stage of the optimisation process.

What defines "short term" and what makes a "complex procedure," according to various scholars, will mean different things. It should be stressed, nonetheless, that the current experimental work serves as the empirical motivation for this operational formulation. The majority of NISQ algorithms are now being experimentally tested and operated on modern hardware.

This contrasts with the majority of fault-tolerant algorithms, which, with the exception of simple instances (such as using Shor's method to identify prime factors for the number 15), cannot be executed on contemporary quantum computers. In fact, the majority of these algorithms openly indicate that they need a high level of fault tolerance in order to function.

Another class of NISQ algorithms is called Quantum Annealing. Quantum annealing (QA), which was developed as the quantum equivalent of simulated annealing (SA), is a heuristic (no guarantees on quantum speedup) optimisation approach that tries to address challenging optimisation issues.

$$= -\sum \sigma \qquad - \sum \sigma \sigma$$

where σ is the Pauli Z operator on the ith site of the Ising model, encoding the problem's

solution into the ground state of the so-called annealer Hamiltonian, which is typically the transverse field Ising Hamiltonian, achieves this. Finding the ground state of such a transverse field Ising Hamiltonian may be translated into a variety of optimisation issues, including challenges with work scheduling and chain optimisation.

In this mechanism, quantum tunneling replaces thermal fluctuations and enables the system to instantaneously depart local minima and explore nearby state space. The phrase is frequently referred to as the "kinetic term" since it provides a visual explanation of its role in the annealing process. The word is referred to as such because the quantum fluctuations are parameterized by (t). The quantum fluctuations eventually start to fade away when the system cools down to zero "temperature," which enables us to gradually narrow our attention on the actual ground state. The system experiences extremely powerful quantum fluctuations at high (t), allowing it to access almost all states. Moreover, quantum annealing has applications in random sampling, where selecting samples from a large number of the annealer's low energy states aids in characterizing the energy landscape. Quantum annealing is closely connected to adiabatic quantum computing, one of the fundamental models of quantum computation. In fact, it was initially suggested as a useful way to build adiabatic quantum processing. The most cutting-edge quantum annealing processors available today, such as those made by D-Wave systems with over 2000 superconducting flux qubits, can only implement a portion of the protocols necessary for universal quantum computation. However, there is still much space for improvement in the way that quantum annealing is actually implemented. You may read more technical information about quantum annealing in. Another algorithm of the NISQ is the Quantum Approximate Optimization Algorithm. Combinatorial optimization issues can be resolved on a quantum computer using the Quantum Approximate Optimization Algorithm, or QAOA for short. All combinatorial optimisation issues may be seen as issues of locating the eigenvector corresponding to the biggest eigenvalue of a Hamiltonian that is diagonal in the computational basis, as stated in the section on combinatorial optimisation. It is clear that QAOA and quantum annealing are related processes. Both depend on a separate term, whose purpose is to let quantum fluctuations explore adjacent states but does not commute with the issue Hamiltonian. However there are some significant variations. First off, in QAOA we are really seeking to rapidly switch between the issue Hamiltonian and the kinetic/mixing Hamiltonian, but in quantum annealing the goal is to maintain the state in its instantaneous ground state at all times and to gently erase the kinetic/mixing Hamiltonian. Iteratively minimizing the cost function results in the ground state of the target Hamiltonian. A classical optimizer uses a quantum computer to assess the cost function and determine its gradient at each stage of the optimisation process.

What defines "short term" and what makes a "complex procedure," according to various scholars, will mean different things. It should be stressed, nonetheless, that the current experimental work serves as the empirical motivation for this operational formulation. The majority of NISQ algorithms are now being experimentally tested and operated on modern hardware.

It should be emphasized that since we are restricted to shallow circuits in the NISQ period, approximate solutions to these combinatorial issues are required. Despite this, it has been demonstrated (with certain plausible complexity theoretic assumptions) that classical computers are unable to replicate the output probability distribution of the bit strings produced by p=1 QAOA, making QAOA a potential candidate technique for quantum advantage in the NISQ era. By significantly altering the QAOA algorithm to create a variation known as adaptive QAOA, the QAOA algorithm's performance in the NISQ era can also be enhanced. QAOA may be expanded to a form where it can be used to do universal quantum computation in addition to solving combinatorial challenges.

A family of NISQ algorithms that we refer to as quantum-assisted techniques has been suggested in addition to the VQAs already described. These techniques may be used to simulate the time evolution under a given Hamiltonian as well as to determine the ground state energy of a given Hamiltonian. Building an ansatz state in which the states $|\chi i\rangle$ are connected to the Hamiltonian H of the issue is the first stage. The cumulative K moment states, initially introduced in, are one particular option of states that have been frequently employed. There are several approaches to build these problem-aware states $|\chi i\rangle$. These cumulative K moment states are based on the notion of producing a Krylov subspace for the Hamiltonian H that appropriately encodes the problem-specific information H. The next stage is to build the so-called D and E matrices using quantum computer measurements. Based on the selection of states $|\chi i\rangle$ in the ansatz, there are certain simplifications that may be done for this phase. In the case where H is expressed as a linear combination of Pauli strings, for instance, using the cumulative K moment states reduces the measurement of the aforementioned matrix elements to the straightforward task of sampling the efficiently prepareable quantum state $|\psi 0\rangle$ in various Pauli-rotated bases on a NISQ computer. The final stage involves performing some standard post-processing on a standard computer using the D and E matrices. The objective is to generate the vector from the D and E matrices for the Hamiltonian ground state problem such that the ansatz state corresponds to the lowest energy eigenstate. According to the description above, the fundamental concept is to combine a classical computer with a noisy quantum computer in order to take use of the capabilities of both types of computers, much like VQAs. Yet, there are a lot of significant distinctions between VQAs and quantum-assisted approaches. Secondly, since all the necessary quantum processing is completed in a single step, we can see that, unlike VQAs, quantum-assisted techniques do not rely on a feedback loop between a classical and a quantum computer. This is an advantage over VQAs because the classical-quantum feedback loop, which must queue up each work for the quantum computer and send it to the quantum computer at each iteration of the feedback loop, can be a significant bottleneck when running VQAs on cloud-based quantum computers. Second, the barren plateau issue that frequently bedevils VQAs is avoided by the quantum aided approaches since they do not employ a parametric quantum circuit.

Throughout the last two decades, there has been a lot of buzz around quantum computers since they promised to effectively handle several difficult problems that no known classical algorithm could. The fact that difficult issues like prime factorization have broad consequences for several businesses and have the ability to upend whole technological sectors makes this issue even more serious. They could also have an impact on national defense. National research organizations and multinational IT corporations have poured money into their own quantum computing divisions for each of these reasons.

But, developing fault-tolerant quantum computers would likely take some time, making it impossible to solve those really difficult problems without them. Such quantum computers need significant improvements in the number of qubits accessible as well as experimental control methods for qubit gate operations. It is clearly a challenging task to develop a fault-tolerant quantum computer, and it will probably take another decade at the very least. Yet, there is no reason why we won't be able to make such devices someday, and experimental success in recent years offers grounds for confidence.

In terms of technology, NISQ devices are noteworthy because they can show a quantum advantage over classical computation techniques for certain particular issues.The most successful platforms are a select handful. Each platform has intrinsic benefits and drawbacks, and it is yet too early to determine which is most likely to succeed in the long run. It is not helpful to conceive in terms of fault-tolerant algorithms (or at least primarily in such terms) due to the restrictions of NISQ devices, as they are likely to be impossible to implement on such devices.For such devices, distinct, NISQ-friendly algorithms will need to be created. It could also be required to rely less on digital computing and more on analogue computing. Contrary to fault-tolerant algorithms, it is unknown if or under what circumstances quantum advantage is feasible for practical issues that are of interest to people outside the quantum information community (such as optimisation and cryptanalysis).As a result, no "killer apps" exist at this time. However, quantum computers and NISQ algorithms are still a long way from being used in commercial settings. Certain sectors, such as quantum simulations and machine learning, have been singled out as exhibiting the most potential of technological advances.

Perfect Intermediate Scale Computings

P stands for the usage of perfect qubits with perfect quantum gates and no decoherence. Based on the OpenQL programming language, our quantum compiler creates cQASM. We built a back-end compiler pass that converts the cQASM version to either the eQASM version for semiconducting or superconducting qubits when we want to test them in the Intel environment. According to our plan, we may transition to any qubit technology in the same way, including photonics and ion traps. On the other hand, it is obvious that application and algorithm developers working at the top of the stack simply aim to represent their notions and ideas in terms of perfect qubits and validate the results of computation. This is an advantage over VQAs because the classical-quantum feedback loop, which must queue up each work for the quantum computer and send it to the quantum computer at each iteration of the feedback loop, can be a significant bottleneck when running VQAs on cloud-based quantum computers. Second, the barren plateau issue that frequently bedevils VQAs is avoided by the quantum aided approaches since they do not employ a parametric quantum circuit.

Simply put, we must acknowledge that the amount of data that is now available is orders of magnitude too great for the present processing capability. In order to determine how quantum accelerators would affect that specific subject, we have already studied quantum genetics and are starting to study quantum chemistry. As a result, all academic and research groups are cordially invited to start looking into how quantum logic could affect their issues. The parallelization of the quantum algorithm or circuit that will be used to run on the conventional supercomputer is an issue that cannot be disregarded by researchers working in the PISQ path. The qubits in the NISQ method execute the potential solutions implicitly, hence performing the parallelization. A parallel version of the circuit must be explicitly created when running a comparable circuit on a supercomputer in order to provide the same number of solutions as a quantum physics operation would. Supercomputers: While Preskill refers to 50 qubits operating on supercomputers as the highest limit, we will also encounter a comparable restriction. Yet, since the qubits are flawless, there is no doubt regarding the output. A compiler option to construct and simulate a circuit employing perfect qubits is present in the majority of quantum compilers. How many more qubits we can superpose to get more than 50 perfect qubits and run it on a supercomputer is still an open question. Classical Memory Use: When a preliminary phase of the quantum application has been finished, any quantum accelerator will have to represent the pertinent qubit states. This is necessary because a quantum simulator like QBeeSim has to be rebooted and have its memory cleared. The local memory of the quantum accelerator can hold interim solutions required for the subsequent stage in the quantum application, which can then be sent to any simulator employing it. Such local memory must represent the qubit basis as well as the amplitudes of each state and is classical in nature. Simply put, we must acknowledge that the amount of data that is now available is orders of magnitude too great for the present processing capability. In order to determine how quantum accelerators would affect that specific subject, we have already studied quantum genetics and are starting to study quantum chemistry. As a result, all academic and research groups are cordially invited to start looking into how quantum logic could affect their issues.

We have observed that the number of gates does not need the classical computer to use more memory, thus we are not required to impose a cap on the size of the circuits at a specific number of gates. The amount of qubits that we can superimpose or entangle continues to be the major problem. We are now looking at ways to use unique quantum simulators based on tensor networks and structure within the quantum circuit to increase the amount of qubits to reach closer to 100 or 150 completely linked qubits.

Simply put, we must acknowledge that the amount of data that is now available is orders of magnitude too great for the present processing capability. In order to determine how quantum accelerators would affect that specific subject, we have already studied quantum genetics and are starting to study quantum chemistry. As a result, all academic and research groups are cordially invited to start looking into how quantum logic could affect their issues.

Specialized Quantum Gates: Each of the qubit technologies under investigation has its own native gate set, which facilitates universal computing by translating other logic to a decomposition of the native gate set. Based on how simple it is to break down the application's main algorithm for the qubit technology, these translations imply that some applications are significantly easier to translate for one qubit technology than for another. According to this perspective, it would be advantageous to create quantum hardware that is especially made to serve particular applications, such as particular controlled rotations for QFT.

Regarding applications, it is obvious that the physics community is highly eager to investigate challenging issues from their subject. Yet, the quantum application layer is theoretically not subject to any topic restrictions. Although quantum finance and quantum genomics are the areas in which we specialize[11], there is additional interest in problems related to chemistry, biology, and other disciplines. Universities will be able to begin doing research in any scientific subject thanks to the PISQ technique because of how much QC will affect all scientific fields. Starting that fresh research viewpoint as soon as possible is advised.

Simply put, we must acknowledge that the amount of data that is now available is orders of magnitude too great for the present processing capability. In order to determine how quantum accelerators would affect that specific subject, we have already studied quantum genetics and are starting to study quantum chemistry. As a result, all academic and research groups are cordially invited to start looking into how quantum logic could affect their issues.

Fault Tolerant Intermediate Scale Computing

For noisy intermediate-scale quantum (NISQ) systems without fault tolerance, previous research has proposed a quantum instruction set architecture (QISA) and a quantum control microarchitecture. Yet, fault-tolerant (FT) quantum computing calls for repetitive quantum error correction, perhaps at runtime, and FT implementation of logical processes. Although highly patterned, logical processes need a substantial number of (physical) actions, which cannot be efficiently carried out by current quantum control microarchitectures.

The capacity of quantum computers to effectively address issues like integer factorization and quantum chemical modeling, which are inefficiently solved by conventional computers, makes them a viable technology. In order to create a fully programmable quantum computer based on the circuit model, a quantum instruction set architecture (QISA) and quantum control microarchitecture must work in unison with quantum software and hardware.

Large-scale quantum algorithms must be implemented using fault-tolerant quantum computing (FTQC), which is based on quantum error correction (QEC), due to the short qubit coherence times and incorrect quantum operations. The fundamental concept of quantum error correcting code (QEC) is to encode quantum information into a logical qubit using a set of physical qubits (QECC). Error syndrome measurement, a highly structured procedure, is essential to regularly detect and (if necessary) repair any quantum faults in order to achieve fault-tolerance (ESM). Moreover, a succession of physical operations should be used to carry out quantum operations on these logical qubits in order to prevent individual physical operation failures from destroying the data held in the logical qubits. Hence, FTQC significantly raises both the quantity of physical operations and the quantity of physical qubits needed.

Nevertheless, Noisy Intermediate-Scale Quantum (NISQ) devices with around fifty to hundreds of qubits are the major goal of the quantum control microarchitectures and QISAs presented by current research. In these devices, quantum error correction is not used. The QISA needed by FTQC may differ from the QISA required by NISQ technology for the reasons listed below.

Without using QEC, quantum algorithms aimed towards NISQ technology directly interact with each physical qubit. In contrast, logical qubits are used by FTQC quantum algorithms. Individual physical actions, which are necessary to accomplish some logical operations like initialization, should be supported by the microarchitecture in addition to logical operations.

More difficult classical computations are added by QEC at runtime, including quantum error decoding and Pauli frame error tracking, which need for the inclusion of additional QISA instructions and new blocks in the control microarchitecture.

Qubits must undergo repeated physical actions as part of the quantum error correcting procedure. It greatly raises the amount of quantum operations performed on qubits in a given amount of time and exacerbates the control microarchitecture's quantum operation issue rate problem.

Creating a scalable and adaptable control microarchitecture that can meet the needs of quantum error correction and fault-tolerant logical processes is a challenging task. The rotating planar surface code is described in this study as having a fault-tolerant microarchitecture, or FT-QuMA, with logical operations carried out via lattice surgery. With the use of microarchitectural support, we incorporate virtual memory into quantum computing, which helps to create a clean compilation model independent of the real physical addresses of qubits, which can differ from device to device; We provide a method to facilitate quantum error detection and repair at the microarchitecture level, which can enable adaptable fault-tolerant logical processes based on lattice surgery and planar surface codes;

To allow effective execution of quantum instructions, we provide a hardware method that significantly decreases the executable's code size.

The Comparison Pattern, Voting Pattern, and Sparing Pattern are three architectural patterns that are discussed in this section to help software systems be more fault tolerant. The application of the patterns to hybrid quantum software systems to increase fault tolerance is covered in later parts.

The implementation of the comparison pattern involves two channels or components that operate simultaneously while the output from each component is compared by a comparator. Countermeasures must be made to maintain the system responsive if the results differ, such as turning on fail-safe mode.

The voting pattern defines N components, each of which is given an input query. Each component's output is sent along to the voter. By combining all of the results, the voter chooses one out of N, for example, by using a majority voting technique. So, the system is functioning well if the vast majority of the N components—where N is thought to be an odd number—are not reporting any faults.

The sparing pattern also specifies N-1 spare components and 1 operating component. An additional spare part is added to the functional component and the spares.

component for error detection to verify outputs for mistakes. In other words, if a fault is found in the operational component's output, one of the spares assumes control. A Switch component implements the "take over" logic.

Measurements are discrete probability distributions that must be processed in quality control to provide a final result. Nevertheless, the distributions are susceptible to measurement and gate errors, which means that the actual measured distribution may differ greatly from the ideal one. So, it is the combiner's task to synthesize a refined distribution in order to aggregate the distributions of the individual measurements. Many mathematical methods for combining probability distributions are described in literature. More specifically, many expert opinions are combined using aggregation methods. Each opinion has a probability distribution that describes it and a level of uncertainty attached to it (to model the subjectivity or partial lack of knowledge of an expert).

The combiner pattern may be used in a number of different ways depending on the situation. The first thing to consider is how much duplicate design should go into a qrChannel. We discussed two different types of redundancy in section III-A, including hardware and algorithmic level redundancy. It is now necessary to determine which type of redundancy is better suited to the existing situation. For instance, QCs may be employed in situations when there are just a few QC algorithm implementations or variants, making algorithmic redundancy less beneficial. On the other hand, there are several QC methods and modifications for other issues (such as graph problems), making algorithmic redundancy more interesting.

In terms of hardware level redundancy, it may be argued that it is less beneficial if there are fewer quantum computers accessible on which the algorithms can be run. In conclusion, the manner in which qrChannels are rendered redundant is context-dependent and requires careful consideration in light of the resources and QC-specific alternatives that are readily accessible, i.e., accessible quantum computers and relevant quantum algorithms. Redundancy at both the algorithmic and hardware levels is also important. The installation of the combiner must then be taken into account. The choice of weights is the most pertinent query in this situation. As was previously said, one method is to use vendor-specific SDKs to obtain real-time data from backends and calculate the weights on the fly. Alternatively, a domain expert or QC specialist can be contacted to determine the weight to be given to each qrChannel based on prior knowledge about quantum computers, such as the topological graphs they have implemented or, more broadly, the effectiveness of quantum algorithms for solving certain issues.

The redundancy of the qrChannel must be a primary issue for the comparison and sparing patterns, just as it is for the combiner pattern. To this aim, the same factors that apply to the combiner pattern must be taken into account, i.e., whether redundancy realizations should be categorized as hardware, algorithmic, or a combination of both. Thus, only those parts of both patterns that aggregate the several outputs into a single output need to be described.

This component refers to the switch component with regard to the sparing pattern. The rationale of the switch component in QC is the same as in other domains; if an error is found, one of the working spares is chosen. But what's interesting is how the error detection component was implemented. Using Witnesses (or Certificates) from computational complexity theory is one method that might be used to detect errors in quantum measurements. For specific issues, such as Satisfiability, a witness is employed to determine whether a certain solution is appropriate for the issue at hand. When a problem arises, the witness determines if the proposed solution fulfills the logical phrase in question. In other words, a witness confirms the outcome of a measurement or alerts if the outcome is incorrect. The switch component can then transfer control to one of the backups.

The two qrChannels' results are sent to the comparator component for comparison in the comparison pattern. The output is disregarded if the findings differ too widely from one metric to another. The topic of how a comparator or metric for determining the difference between two measurements can seem arises in the context of quality control (QC). The Kullback-Leibler Divergence (KLD), which is used to compare probability distributions, is one potential metric that is well-known from probability theory.

Measurement divergence between two measurements may be determined using KLD since measurements are equivalent to probability distributions. The whole output is rejected and suitable countermeasures can be used if their divergence is too great (i.e., the measured KLD exceeds a certain threshold).
As the measurements' distribution is discrete, one may also take into account easy deviation approaches, such as the average difference between each measurement.

The entire area of quantum error correction is important in terms of fault tolerance in QC. A collection of techniques for addressing qubit fault tolerance at the algorithmic level are referred to as quantum error correction. Contrarily, our suggested method addresses fault tolerance at the architectural level.

Other methods, such as employing reliability measures for quantum circuits, locate qubit allocations (i.e., the allocation of programme qubits to physical qubits) that imply low failure probability in order to deal with NISQ-specific uncertainty. Contrary to our suggested method, these techniques also address fault tolerance at the algorithmic level. It's interesting to note that research has indicated that different mappings of qubit allocation—that is, performing numerous qubit allocations on the same quantum computer—increase the precision of the measurements when the resulting measurements are combined. This strategy demonstrates once again the possibilities of our method and may also be implemented using the combiner pattern.

The system's dependability may be increased using a variety of methods that reuse fault-tolerant design principles in different settings. To the best of our knowledge, no method exists that uses fault-tolerant architectural principles in the context of QC and serves as the primary differentiation from methods that are connected to fault tolerance.

2. Conclusion

According to the law of unitary evolution, in quantum computing, all the processes and schemes must be reversible. In the NISQ framework, the circuit model is taken into consideration but there also exists the one way or the measurement based quantum computation scheme which is not reversible yet it is shown that it is equivalent to the circuit model. Barren plateaus mean that, on average, an exponential number of shots must be taken to resolve gradients to a fixed accuracy. This substantially increases the amount of resources required for Variational Quantum Eigensolver training. Another possible issue is that in order to avoid wasting time transmitting signals back and forth in the feedback loop, the quantum devices must be tightly connected to a classical device. Building a device that can hold a lot of qubits while preserving stability and coherence is one of the fundamental issues in quantum computing.

Due to their extreme sensitivity to noise and mistakes, quantum systems can cause computing to fail and provide unreliable results. Building an usable quantum computer requires both error correction and error mitigation, but the techniques for doing so are still in their infancy.

3. Future Directions

In quantum computing, all processes and schemes must be reversible in accordance with the law of unitary development. The circuit model is taken into account in the NISQ framework, but there is also the one-way or measurement-based quantum computation approach, which is not reversible but is demonstrated to be comparable to the circuit model. To understand how the measurement-based quantum computation scheme, which does not adhere to the law of reversibility that governs unitary development in quantum computing, may yet be equal to the circuit model, further study must be done into how it operates.

Increasing the size of fault-tolerant intermediate-scale quantum devices will lead to fault-tolerant large-scale quantum devices, which are where the production scale quantum advantage thrives. This is another successful field of study for scaling FT intermediate scale devices.

One of the drawbacks with VQA-based problems is that, because of the barren plateaus, it usually requires an exponentially large number of photos to resolve gradients to a fixed accuracy. As a result, a lot more resources are required to train the Variational Quantum Eigensolver. It would be interesting to research the techniques required to significantly decrease processing speed and duration.

References

[1] “ACM Digital Library.” *ACM Digital Library*, dl.acm.org/doi/abs/10.1145/3297858.3304007.

[2] *Quantum Computing: A Taxonomy, Systematic Review ... - Wiley Online Library*. onlinelibrary.wiley.com/doi/full/10.1002/spe.3039.

[3] Niu, Siyuan, and Aida Todri-Sanial. “Enabling Multi-programming Mechanism for Quantum Computing in the NISQ Era.” *Quantum*, doi:10.22331/q-2023-02-16-925.

[4] “Reliability Modeling of NISQ- Era Quantum Computers.” *Reliability Modeling of NISQ- Era Quantum Computers | IEEE Conference Publication | IEEE Xplore*, ieeexplore.ieee.org/abstract/document/9251243.

[5] Ajagekar, Akshay, and Fengqi You. “New Frontiers of Quantum Computing in Chemical Engineering - Korean Journal of Chemical Engineering.” *SpringerLink*, 1 Feb. 2022, doi:10.1007/s11814-021-1027-6.

[6] Salm, Marie, et al. “The NISQ Analyzer: Automating the Selection of Quantum Computers for Quantum Algorithms.” *The NISQ Analyzer: Automating the Selection of Quantum Computers for Quantum Algorithms | SpringerLink*, 7 Dec. 2020, doi:10.1007/978-3-030-64846-6_5.

[7] “ACM Digital Library.” *ACM Digital Library*, dl.acm.org/doi/abs/10.1145/3445814.3446758.

[8] Suzuki, Yasunari, et al. “Quantum Error Mitigation as a Universal Error Reduction Technique: Applications

From the NISQ to the Fault-Tolerant Quantum Computing Eras.” *PRX Quantum 3, 010345 (2022) - Quantum Error Mitigation as a Universal Error Reduction Technique: Applications From the NISQ to the Fault-Tolerant Quantum Computing Eras*, 18 Mar. 2022, doi:10.1103/PRXQuantum.3.010345.

[9] “ACM Digital Library.” *ACM Digital Library*, dl.acm.org/doi/abs/10.1145/3470496.3527434.

[10] *U : Leveraging Operation-Aware Error Rates for Effective ... - Usenix*. www.usenix.org/system/files/atc20-patel.pdf.

[11] D. Malone, Fionn, et al. “Towards the Simulation of Large Scale Protein–ligand Interactions on NISQ-era Quantum Computers - Chemical Science (RSC Publishing) DOI:10.1039/D1SC05691C.” *Towards the Simulation of Large Scale Protein–ligand Interactions on NISQ-era Quantum Computers - Chemical Science (RSC Publishing) DOI:10.1039/D1SC05691C*, 17 Jan. 2022, doi:10.1039/D1SC05691C.

[12] “Investigation of Noise Effects for Different Quantum Computing Architectures in IBM-Q at NISQ Level.” *Investigation of Noise Effects for Different Quantum Computing Architectures in IBM-Q at NISQ Level | IEEE Conference Publication | IEEE Xplore*, ieeexplore.ieee.org/abstract/document/9390130.

[13] “Max-Cut Problem Implementation and Analysis on a Quantum Computer.” *Max-Cut Problem Implementation and Analysis on a Quantum Computer | IEEE Conference Publication | IEEE Xplore*, ieeexplore.ieee.org/document/9826192.

[14] “A New Framework for Quantum Image Processing and Application of Binary Template Matching.” *A New Framework for Quantum Image Processing and Application of Binary Template Matching | IEEE Conference Publication | IEEE Xplore*, ieeexplore.ieee.org/document/9743534.

[15] *https://www.tandfonline.com/doi/abs/10.1080/0194262X.2020.1776193*. www.tandfonline.com/doi/abs/10.1080/0194262X.2020.1776193.

[16] “Experimental Evaluation of NISQ Quantum Computers: Error Measurement, Characterization, and Implications.” *Experimental Evaluation of NISQ Quantum Computers: Error Measurement, Characterization, and Implications | IEEE Conference Publication | IEEE Xplore*, ieeexplore.ieee.org/abstract/document/9355264.

[17] “Survey on Quantum Circuit Compilation for Noisy Intermediate-Scale Quantum Computers: Artificial Intelligence to Heuristics.” *Survey on Quantum Circuit Compilation for Noisy Intermediate-Scale Quantum Computers: Artificial Intelligence to Heuristics | IEEE Journals & Magazine | IEEE Xplore*, ieeexplore.ieee.org/abstract/document/9384317.

[18] “Software Architecture for Quantum Computing Systems — a Systematic Review.” *Software Architecture for Quantum Computing Systems — a Systematic Review - ScienceDirect*, 24 Mar. 2023, doi:10.1016/j.jss.2023.111682.

[19] “Analysis of the Likelihood of Quantum Computing Proliferation.” *Analysis of the Likelihood of Quantum Computing Proliferation - ScienceDirect*, 8 Jan. 2022, doi:10.1016/j.techsoc.2022.101880.

[20] Zajac, Markus, et al. “Implementation Aspects for Linking Data in Hybrid Quantum Applications - Digitale Bibliothek - Gesellschaft FÜ Informatik e.V.” *Implementation Aspects for Linking Data in Hybrid Quantum Applications - Digitale Bibliothek - Gesellschaft FÜR Informatik e.V.*, 1 Jan. 2022, dl.gi.de/handle/20.500.12116/39463.

[21] “Towards Quantum-based Search for Industrial Data-driven Services.” *Towards Quantum-based Search for Industrial Data-driven Services | IEEE Conference Publication | IEEE Xplore*, ieeexplore.ieee.org/abstract/document/9860162.

[22] Mendes, Luís Ricardo Matos. “Scalable Parallelization of Quantum Computing Simulations.” Repositorio-Aberto.up.Pt, https://repositorio-aberto.up.pt/bitstream/10216/142721/2/572019.pdf. Accessed 15 Apr. 2023.

[23] Zajac, Markus, et al. *Implementation Aspects for Linking Data in Hybrid Quantum Applications*. Gesellschaft für Informatik, Bonn 2022.

[24] Duranton, Marc, et al. “Editorial Board.” Inria.Fr, https://hal.inria.fr/hal-04023794/file/HiPEAC_Vision_2023.pdf#page=113. Accessed 15 Apr. 2023.

[25] Jaschke, Daniel, et al. “Ab-Initio Two-Dimensional Digital Twin for Quantum Computer Benchmarking.” *ArXiv [Quant-Ph]*, 2022, http://arxiv.org/abs/2210.03763.