Clifford Initialization for the Quantum Approximate Optimization Algorithm

Dhanvi Bharadwaj
University of Michigan
Ann Arbor, MI, USA

Gokul Subramanian Ravi
University of Michigan
Ann Arbor, MI, USA

Abstract

Variational Quantum Algorithms (VQAs) such as the Quantum Approximate Optimization Algorithm (QAOA) offer a promising path to solving combinatorial optimization problems on near-term quantum devices. However, their success strongly depends on the initial ansatz parameters, and the limited expressiveness of the QAOA ansatz makes it difficult to efficiently and scalably identify good initialization points. To address this challenge, we propose a framework[1] that uses a relaxed QAOA ansatz to enable classical search over a set of Clifford-preparable quantum states with high-quality solutions. We show that these states are used as high-quality initializations for QAOA, facilitating rapid convergence to near-ground-state solutions.

We design our framework to be application-agnostic, supporting a broad class of combinatorial optimization problems, including QUBO and PCBO formulations. Our method improves upon state-of-the-art QAOA initializations by providing higher solution quality across a wide range of combinatorial optimization problems. By efficiently guiding QAOA towards optimal solution states, our framework enhances convergence and unlocks more consistent performance on near-term quantum platforms.

For QAOA on a wide variety of combinatorial optimization problems, our framework achieves nearly 99.9% mean accuracy and produces an exceptionally concentrated initialization distribution, reducing distinct initial states by up to 1000× compared to random initialization. Additionally, we benchmark our framework on QAOA problems across a broad scale, from tens to hundreds of qubits, and evaluate it on a wide range of QUBO and PCBO instances derived from real-world data, demonstrating consistent performance across problem types and sizes.

1 Introduction

Quantum computers offer the promise of transforming computational power, especially in key areas like chemistry [25], optimization [31], and machine learning [4]. In contrast to classical systems, they harness core principles of quantum mechanics—such as superposition, interference, and entanglement—enabling them to tackle problems that are substantially, and sometimes exponentially, more difficult for even cutting-edge classical architectures.

In the current era of quantum computing, we primarily depend on noisy intermediate-scale quantum (NISQ) devices [35]. These platforms are constrained by both limited qubit counts and reliability, with performance affected by short coherence times, state preparation and measurement (SPAM) errors, gate imperfections, and crosstalk. As a consequence, current NISQ systems are not yet capable of running large-scale quantum algorithms that depend on full fault tolerance and comprehensive error correction, which would require millions of qubits [33]. Nevertheless, even with these limitations, NISQ devices may still provide quantum advantage for certain targeted applications, such as optimization [12], physics [27], and chemistry [34].

Interestingly, there are certain algorithms developed for the NISQ era that exhibit an innate robustness to noise, allowing them to generate valuable results even in the absence of full quantum error correction. Among these, Variational Quantum Algorithms (VQAs) are especially notable for their broad applicability. VQAs can be used for tasks such as solving combinatorial optimization problems like Max-Cut [12] and estimating molecular electronic energies [34]. These are hybrid algorithms in which a parameterized quantum circuit is refined through iterative optimization by a classical routine. Over time, the VQA explores the quantum solution landscape, minimizing an objective function to converge on the problem's "ground state"—the optimal solution.

Within the broader class of VQAs, one algorithm specifically designed to approximately solve combinatorial optimization problems is the Quantum Approximate Optimization Algorithm (QAOA) [13]. QAOA alternates between applying problem-specific and mixing Hamiltonians, with the goal of steering the parameterized state toward a low-energy state that encodes a high-quality solution. The algorithm is parameterized by a set of a few angles, which are tuned by a classical optimizer to minimize the expected energy of the final quantum state. As the circuit depth increases, QAOA can, in principle, approximate the optimal solution with increasing accuracy, making it a promising candidate for near-term quantum advantage in discrete optimization tasks.

[1] Our framework will be open-sourced

Correspondence: dhanvib@umich.edu, gsravi@umich.edu

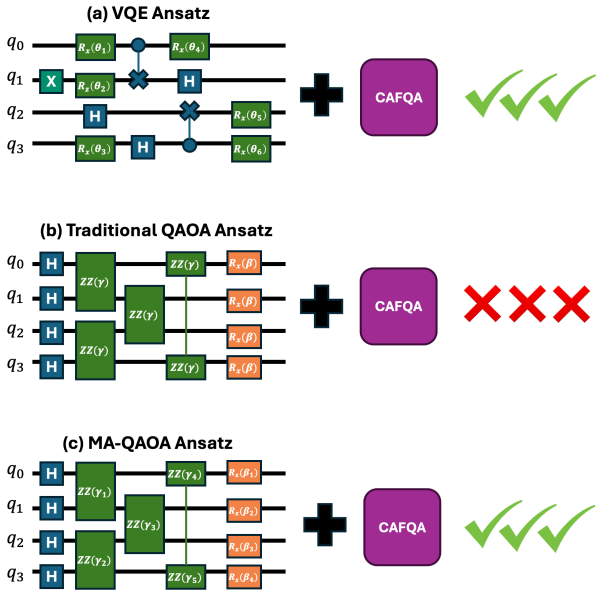

Figure 1. (a) We can design the VQE ansatz to be expressive enough to allow many Clifford states to explore the parameter space. (b) The traditional QAOA ansatz imposes a uniform angle across each layer, which considerably restricts the search space for Clifford angles. (c) The relaxed QAOA ansatz significantly increases the expressivity that allows for high-quality Clifford solutions to exist in the search space.

As with many optimization algorithms, the choice of initial parameters in QAOA plays a critical role in guiding the algorithm toward convergence to the optimal solution. Multiple studies have demonstrated that QAOA's performance heavily depends on its initial parameters [5, 40]. The impact of initialization on QAOA performance is also a well-studied topic within the broader field of variational quantum algorithms. However, searching for these low-energy initial parameters—especially in high-dimensional and noisy landscapes—is computationally intensive. This challenge is amplified when evaluations must be performed on actual quantum hardware, where each function call may require numerous circuit executions (shots), making the process resource- and time-consuming. Previous initialization strategies for QAOA have largely been tailored to specific applications or problem instances [46, 49, 50], often relying on domain knowledge or heuristics. These methods have typically been tested only on constrained or small-scale problems, unweighted graph instances, and have not demonstrated effectiveness at larger scales.

On the other hand, the Variational Quantum Eigensolver (VQE) [34], another well-studied class of VQA, has benefited from scalable, classically simulable initialization techniques that rely on Clifford states. These approaches have demonstrated strong performance in quantum chemistry tasks and underscore the value of better initializations in VQE efficiency and solution quality. These approaches have demonstrated strong performance in quantum chemistry tasks, highlighting the importance of high-quality initializations for improving VQE efficiency and solution quality. However, directly applying this strategy to QAOA is challenging due to the circuit's constrained structure. In particular, the limited expressiveness of the QAOA ansatz prevents it from representing the ideal stabilizer state [10].

As a result, there is currently no general-purpose initialization framework that is both broadly applicable and scalable across diverse problem types. These limitations present a significant barrier to the widespread and efficient deployment of QAOA across the full spectrum of combinatorial optimization problems, including Quadratic Unconstrained Binary Optimization (QUBO), Polynomial Unconstrained Binary Optimization (PUBO), and Polynomial Constrained Binary Optimization (PCBO) tasks.

In this work, we observe that introducing a relaxed parameter structure into the QAOA ansatz significantly boosts the circuit's expressivity for capturing stabilizer states, which serve as powerful initialization points for optimization. Building on this insight, we propose a versatile initialization framework that combines the multi-angle QAOA ansatz with a Clifford-based search strategy to identify near-ground-state solutions efficiently. To improve coverage of the optimization landscape, we deploy a parallel multi-start approach, launching independent optimization runs from a diverse set of top-performing CAFQA points. Figure 1 provides an overview of our approach.. Unlike existing initialization methods that target specific problems, our approach applies broadly to diverse combinatorial optimization tasks, providing a versatile and scalable solution framework. ***The primary contributions and insights of our work are:***

1. We demonstrate that using the multi-angle ansatz for QAOA [23] enables the identification of optimal Clifford states during the initial parameter search, driving more efficient and effective initialization.
2. We show that an optimal Clifford initialization reduces the overall solution space by eliminating a large fraction of non-optimal states.
3. We present a general and scalable initialization framework that can be applied across diverse combinatorial optimization tasks—including QUBO, PUBO, and PCBO—without relying on problem-specific structure.
4. Our evaluations show that we can achieve up to a maximum of 99.9% over current initialization techniques.
5. Importantly, we evaluated our framework on a diverse set of QUBO, PUBO, and PCBO problems, including an instance derived from a real-world dataset, demonstrating its broad applicability and ability to generalize across both synthetic and practical problem types

2 Background & Motivation

2.1 Combinatorial Optimization Problems

Combinatorial optimization lies at the heart of many real-world decision-making and resource allocation tasks, where the objective is to find the best solution from a finite but often exponentially large set of possibilities. These problems arise in diverse fields such as logistics [41], finance [32], and operations research [37]. Formally, a combinatorial optimization problem involves selecting a subset or ordering of discrete variables that minimizes (or maximizes) a given cost function under certain constraints. Despite their simplicity in formulation, these problems are often NP-hard, making them computationally challenging for classical algorithms as problem sizes scale.

Two prominent classes of such problems that we discuss in our work are Quadratic Unconstrained Binary Optimization (QUBO) and Polynomial Constrained Binary Optimization (PCBO). In QUBO, the objective function is quadratic over binary variables, and there are no explicit constraints—making it a widely used and well-studied form suitable for mapping onto quantum algorithms. On the other hand, PCBO problems involve polynomial objective functions and additional algebraic or logical constraints, significantly increasing their complexity. These generalizations make PCBO a more expressive and flexible problem class, though at the cost of higher computational difficulty.

On classical computers, QUBO and PCBO formulations are addressed using a combination of exact and heuristic methods. Exact approaches such as branch-and-bound, branch-and-cut, and semidefinite relaxations, implemented in solvers like CPLEX and Gurobi [20, 24], can provide optimal solutions but quickly become intractable as problem sizes grow due to exponential scaling. To handle larger instances, a wide range of heuristics and metaheuristics—including simulated annealing and genetic algorithms—are employed to find high-quality approximate solutions [8]. PCBO problems are often reduced to equivalent QUBO instances through quadratization techniques [2, 7], enabling the use of the same solver ecosystem but at the cost of introducing additional variables and constraints. Consequently, for both QUBO and PCBO, practical scalability hinges on heuristics and approximation strategies [18, 28].

2.2 Quantum Approximate Optimization Algorithm (QAOA)

The Quantum Approximate Optimization Algorithm (QAOA) is a hybrid quantum-classical algorithm tailored for solving combinatorial optimization problems. At its core, QAOA constructs a parameterized quantum circuit composed of alternating layers of problem-specific and mixing unitaries, each controlled by a set of classical parameters.

To solve an optimization problem using QAOA, we encode the objective function into a cost Hamiltonian H_C and pair it with a mixer Hamiltonian H_M to guide the variational evolution. We define two unitary operations: $U(\gamma, H_C) = e^{-iH_C\gamma}$ and $U(\beta, H_M) = e^{-iH_M\beta}$, where γ and β are variational parameters. These unitary operations are applied alternately with p repetitions to evolve an initial quantum state towards low-energy configurations under H_C. We initialize the circuit in the uniform superposition over all computational basis states:

$$|s\rangle = \frac{1}{\sqrt{2^n}} \sum_z |z\rangle.$$

We refer to this structure as the traditional QAOA ansatz throughout this work.

QAOA relies critically on the choice of variational parameters $(\vec{\gamma}, \vec{\beta})$, which control the depth-p quantum evolution. Optimizing these parameters poses a significant challenge due to the non-convexity and instance-specific structure of the cost landscape [3, 11, 14, 15, 21, 22, 29, 30, 43, 48]. In practice, a hybrid quantum-classical loop drives the optimization: the quantum circuit prepares a parameterized state, the quantum device measures its cost expectation, and the classical optimizer updates $(\vec{\gamma}, \vec{\beta})$.

While QAOA shares structural similarities with Variational Quantum Eigensolver (VQE), it is fundamentally distinct in its design and application. Unlike VQE, which targets ground-state estimation for quantum chemistry or physics-inspired Hamiltonians, QAOA is specifically formulated for discrete optimization over classical cost functions. This difference leads to unique challenges in circuit design and parameter initialization, particularly due to the constrained and problem-specific structure of the QAOA ansatz. These challenges manifest in several ways: the constrained two-parameter ansatz often limits expressivity at shallow depths, barren plateaus and sharp local minima hinder efficient optimization, and performance is highly sensitive to initialization, with poorly chosen parameters leading to stagnation. Moreover, increasing circuit depth can mitigate expressivity issues but comes at the cost of greater susceptibility to noise on near-term hardware. Together, these factors underscore the need for principled initialization strategies that can guide QAOA toward high-quality solutions efficiently.

2.3 Multi-Angle Ansatz for QAOA

The standard QAOA ansatz is problem-specific and minimally parameterized, typically using only $2p$ parameters. To implement our method effectively, we adopt a circuit ansatz that generalizes across problem instances, enabling the representation of problems with shared structure. For more expressivity, we increase the number of classical parameters per QAOA layer. We achieve this by implementing the multi-angle QAOA (ma-QAOA) [23], which enhances the traditional structure by assigning individual parameters to each term in the cost and mixer operators, instead of using

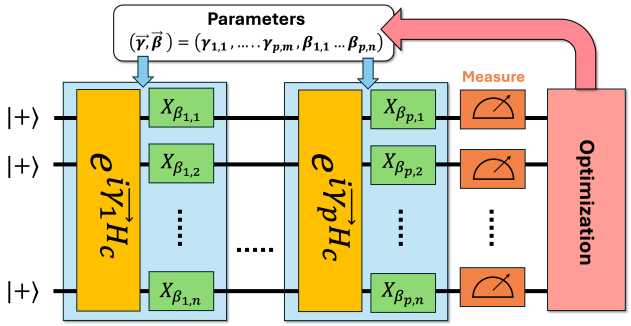

Figure 2. Schematic of a p-layer ma-QAOA structure

a single angle for each. Specifically:

$$U(H_C, \vec{\gamma}_\ell) = e^{-i\sum_{a=1}^{m} H_{C,a}\gamma_{\ell,a}} = \prod_{a=1}^{m} e^{-iH_{C,a}\gamma_{\ell,a}},$$

$$U(H_M, \vec{\beta}_\ell) = e^{-i\sum_{b=1}^{n} H_{M,b}\beta_{\ell,b}} = \prod_{b=1}^{n} e^{-iH_{M,b}\beta_{\ell,b}}.$$

Here, m is the number of clauses and n is the number of qubits. The total number of parameters becomes $(m+n)p$. The standard QAOA is a special case of ma-QAOA, where all parameters within a cost or mixer layer are identical. ma-QAOA expands the parameter space, allowing us to explore more complex solution landscapes and improve optimization efficiency. The ma-QAOA structure is illustrated in Figure 2

2.4 CAFQA

Clifford Ansatz For Quantum Accuracy (CAFQA) [39] is a classical initialization strategy designed to enhance the performance of VQE by providing high-quality initial states. Our implementation leverages a genetic algorithm to navigate this space efficiently, evolving populations of Clifford angles to minimize the energy expectation with respect to the cost Hamiltonian. The method leverages the efficiently simulable structure of Clifford circuits to perform tractable search over a restricted, yet expressive, subset of the Hilbert space. Since Clifford circuits map Pauli operators to Pauli operators under conjugation and can be simulated in polynomial time on classical hardware, CAFQA can systematically evaluate a large number of candidate states to identify one that yields a low energy with respect to the cost Hamiltonian.

Once a promising Clifford state is identified, it is used as the starting point for a standard VQE procedure. This significantly narrows the search landscape, enabling faster convergence and improved performance, especially in complex problem instances. CAFQA has shown notable success in quantum chemistry tasks, where the structure of molecular Hamiltonians aligns well with the expressivity of Clifford-based states [38, 47]. CAFQA demonstrated that Clifford-initialized states can exhibit remarkably high overlap with the true ground state, often achieving 90–99% accuracy in estimating the ground state energy prior to executing any variational updates on quantum hardware.

However, despite its effectiveness, the method is largely tailored to specific problem domains and has not been generalized to combinatorial optimization settings such as QUBO or PCBO problems. Unlike VQE, which allows for flexible, problem-specific circuit constructions, QAOA employs a fixed, layered structure composed of alternating unitary operations generated by the cost Hamiltonian H_C and a non-commuting mixer Hamiltonian H_M. This constrained ansatz limits the set of reachable states, presenting a challenge on the feasibility of incorporating pre-optimized Clifford states as initialization points. While CAFQA originally targeted VQE initialization, our work extends its implementation to QAOA, enabling efficient, high-quality initialization across a broader class of combinatorial optimization problems.

3 Design

We propose a framework that leverages the expressive solution space enabled by the ma-QAOA ansatz to efficiently search for optimal Clifford states as high-quality initialization points. The following subsections describe each component of the framework in detail. At a high level, here are the important steps in the workflow:

1. **Generate Problem Instance:** Utilize a series of mathematical reformulations to represent this problem using the QUBO and PCBO notations.
2. **Reformulate Problem as Hamiltonian:** Rewrite the optimization problem as a Hamiltonian for which the ground state corresponds to the solution that maximizes the cost function.
3. **Build ma-QAOA Ansatz:** Create a multi-angle ansatz which will prepare the ground state of this Hamiltonian.
4. **Search Clifford Space for initial parameters:** We use CAFQA to perform a classical search for optimal Clifford parameters for $(\vec{\gamma}, \vec{\beta})$.
5. **Sample final distribution:** The final quantum state is sampled by extracting the most probable bitstring from the state vector, which corresponds to the solution of the optimization problem.
6. **Post Processing:** The post-processing step interprets the sampling output to return a solution for the original problem.

3.1 Classical-to-Quantum Encoding

To initialize our algorithm, we first generate an instance of the combinatorial optimization problem to be solved. This instance can take the form of a Quadratic Unconstrained Binary Optimization (QUBO), Polynomial Unconstrained Binary Optimization (PUBO), or Polynomial Constrained Binary Optimization (PCBO) formulation, depending on the

problem's structure. These notations express the objective function in terms of binary variables and polynomial cost terms, making them suitable for quantum optimization techniques.

A general QUBO problem is expressed as:

$$\text{minimize} \quad x^T Q x$$

where $x \in \{0,1\}^n$ and $Q \in \mathbb{R}^{n \times n}$ is a real-valued symmetric matrix.

A PUBO extends this to higher-order terms:

$$\text{minimize} \quad \sum_i a_i x_i + \sum_{i<j} b_{ij} x_i x_j + \sum_{i<j<k} c_{ijk} x_i x_j x_k + \cdots$$

Alternatively, A PCBO introduces equality or inequality constraints on binary variables:

$$\text{minimize} \quad f(x) \quad \text{subject to} \quad g_k(x) = 0, \quad h_l(x) \leq 0$$

where $f(x)$ is a polynomial objective and g_k, h_l are polynomial constraint functions.

Representative examples of QUBO problems include Max-Cut and the Maximum Independent Set [6, 44]. For PCBO, classic instances include the Knapsack Problem and 3-SAT, both of which impose structured constraints [9, 26]. PUBO problems often arise in domains such as hypergraph partitioning and higher-order Max-SAT [1], where the objective involves interactions among more than two variables.

These instances can be created using open-source libraries like Qiskit Optimization [36], and the graphs are created using Rustworkx [45], or derived from custom user-defined models.

Once the problem instance is defined, it is typically mapped to a QAOA cost Hamiltonian by translating the objective function and constraints into a Hamiltonian form. This transformation typically involves encoding the optimization problem's cost function into a diagonal matrix, where each binary variable is represented by a Pauli-Z operator, and the interaction terms are encoded as weighted Pauli products. For constrained problems like PCBO, additional terms are introduced to enforce the constraints by using penalty terms that penalize infeasible solutions. The structure of the Hamiltonian is specific to the problem type, and the details of this mapping are problem-dependent. We delve into the exact methods and specific formulations for encoding QUBO, PCBO, and PUBO problems into QAOA Hamiltonians in Section.

Once the cost Hamiltonian is constructed, it serves as the foundation for constructing the QAOA ansatz, which governs the evolution of the quantum state. The QAOA ansatz is constructed based on this Hamiltonian, which is a defining feature of the QAOA algorithm. As outlined above, we enhance the expressiveness of the ansatz by incorporating the ma-QAOA structure. Initially, we construct a standard QAOA ansatz using the Qiskit `QAOAAnsatz` API, which provides the traditional ansatz with $2p$ parameters. We then relax the parameters in $(\vec{\gamma}, \vec{\beta})$, allowing them to take independent values and increasing the flexibility of the ansatz. The total number of parameters becomes $(m+n)p$, where m is the number of clauses in the problem and n is the number of qubits.

3.2 Optimization

The ma-QAOA structure aligns more effectively with the CAFQA initialization technique, as it offers a broader solution space for exploration. We utilize the CAFQA technique to explore Clifford angles that minimize the expectation value of the quantum circuit with respect to the cost Hamiltonian. To perform this search, we leverage a Genetic Algorithm (GA) optimizer, which is a gradient-free approach, to navigate the Clifford solution space. The GA is implemented using the PyGad library [16]. Since these parameters have to be Clifford angles, they can only take values that are integer multiples of $\frac{\pi}{2}$, corresponding to rotations by 0, $\frac{\pi}{2}$, π, or $\frac{3\pi}{2}$ radians. These specific angles ensure that the quantum operations remain within the Clifford group, which is efficiently simulatable and preserves Pauli operators under conjugation.

Once the optimal parameters are identified by CAFQA, we use them to seed a multi-start optimization strategy. This approach is motivated by the observation that some of the best-performing CAFQA points themselves may be located at or close to a deep local minimum. By launching multiple asynchronous parallel optimization runs, each initialized with different top-performing CAFQA points, we improve coverage of the complex and potentially rugged solution landscape. We enable this concurrent evaluation by assigning each optimization run a separate core.

For the subsequent optimization process within the traditional QAOA workflow, we can flexibly choose from a variety of classical optimizers depending on the problem characteristics. Gradient-free optimizers such as COBYLA, SPSA, and Nelder-Mead are often preferred in noisy or non-smooth landscapes, while gradient-based optimizers like L-BFGS-B and Adam can offer faster convergence when gradient information is reliable.

Each classical optimizer instance repeatedly updates the variational parameters $(\vec{\gamma}, \vec{\beta})$ of the quantum circuit, improving the solution and maximizing the cost function based on quantum measurement outcomes.

This continuous optimization process is not limited to discrete Clifford angles; it allows any angle in the interval $[-\pi, \pi)$, expanding the search beyond the Clifford subspace. After all runs complete, we select the best result across the multiple starts, thus increasing the likelihood of reaching a globally optimal or near-optimal solution.

3.3 Post Processing

After the optimization process, a final post-processing step is performed to extract the approximate optimal result from QAOA for the respective cost Hamiltonian. The quantum state is measured in the computational basis, yielding a set of bitstrings that represent possible solutions to the problem. These bitstrings are sampled according to the final quantum state, and the most probable bitstring is selected.

Let the optimized quantum state after the variational optimization process be denoted as:

$$|\psi(\vec{\gamma}, \vec{\beta})\rangle = U_{\text{QAOA}}(\vec{\gamma}, \vec{\beta})|0\rangle,$$

where $U_{\text{QAOA}}(\vec{\gamma}, \vec{\beta})$ is the quantum evolution operator parameterized by the variational parameters $\vec{\gamma}$ and $\vec{\beta}$, and $|0\rangle$ is the initial state. The probability of measuring a specific bitstring $\mathbf{z}$ is given by:

$$P(\mathbf{z}) = |\langle \mathbf{z}|\psi(\vec{\gamma}, \vec{\beta})\rangle|^2.$$

The bitstring $\mathbf{z}$ with the highest probability is then selected as the candidate solution. To increase reliability, the sampling process is often repeated multiple times, and the frequency of occurrence of each bitstring is analyzed. The bitstring that appears most frequently is deemed to be the final solution.

For example, in Max-Cut, the bitstring is directly mapped to a graph partition. If the solution bitstring is $\mathbf{z} = (z_1, z_2, \ldots, z_n)$, where n is the number of vertices in the graph, then the vertices corresponding to $z_i = 0$ are placed in one set, and the vertices corresponding to $z_i = 1$ are placed in the other set. The objective of the Max-Cut problem is to partition the graph into two sets such that the number of edges between the sets is maximized. The solution is determined by the most probable bitstring, which represents the optimal partition. For other combinatorial problems, such as the knapsack problem, the bitstring requires a different interpretation. Let $\mathbf{z} = (z_1, z_2, \ldots, z_n)$ denote a bitstring representing the selection of items in a knapsack, where n is the number of items. Each bit z_i takes a value of either 0 or 1, where $z_i = 1$ indicates that item i is selected, and $z_i = 0$ indicates that item i is not selected. The objective is to maximize the total value of the selected items, subject to the constraint that their total weight does not exceed the capacity W of the knapsack. This can be formulated as:

$$\text{Maximize } V(\mathbf{z}) = \sum_{i=1}^{n} v_i z_i \quad \text{subject to} \quad \sum_{i=1}^{n} w_i z_i \leq W,$$

where v_i is the value and w_i is the weight of item i. After obtaining the most probable bitstring, we perform a post-processing step to ensure that the total weight of the selected items satisfies the capacity constraint. If the constraint is violated, a heuristic or a correction procedure can be applied to adjust the selection.

Thus, the interpretation of the bitstring depends on the specific problem formulation, with post-processing steps tailored to the constraints and objectives of the problem at hand. In the end, the post-processing step ensures that the quantum-optimized solutions are interpreted correctly and transformed into feasible results for the specific problem at hand.

4 Methodology

To systematically evaluate the performance of our framework, we use one QUBO problem and two PCBO problems as benchmarks for QAOA.

4.1 Benchmarks

4.1.1 QUBO Benchmark. We select Max-Cut as the QUBO benchmark. The cost Hamiltonian for a graph G with edge set $E(G)$ is typically expressed as

$$H_{\text{C}} = \sum_{(i,j)\in E(G)} \frac{1}{2} w_{ij} \left(I - Z_i Z_j\right),$$

where w_{ij} are the edge weights and Z_i, Z_j are Pauli-Z operators acting on qubits corresponding to nodes i and j. In our study, we simplify the cost Hamiltonian and directly minimize

$$H_{\text{C}} = \sum_{(i,j)\in E(G)} w_{i,j} Z_i Z_j,$$

which preserves the essential structure of the problem, allowing for some scaling and translation. Max-Cut is particularly suitable for evaluating our framework, as it enables exploration of diverse graph structures under consistent optimization settings. This simplified form maintains the problem's optimization landscape while streamlining implementation and analysis. We evaluate performance across three distinct types of graphs, which are illustrated in Figure 3: complete graphs, 3-regular graphs, and ego graphs. Complete graphs are densely connected, with every node linked to every other node, creating a highly entangled and complex optimization landscape. In contrast, 3-regular graphs are sparsely connected, with each node connected to exactly three others, leading to more localized interactions and a different entanglement structure. Ego graphs are constructed around a central "ego" node and include its immediate neighbors, capturing local community structure and heterogeneous connectivity patterns that lie between the dense and sparse extremes. Comparing these cases allows us to systematically assess how different initialization strategies perform under varying problem complexity and structure.

4.1.2 PCBO Benchmarks. We select the Knapsack problem and a Cancer Genomic dataset [42] as representative PCBO benchmarks. In the Knapsack problem, the objective is to maximize the total value of selected items while ensuring their combined weight does not exceed a predefined

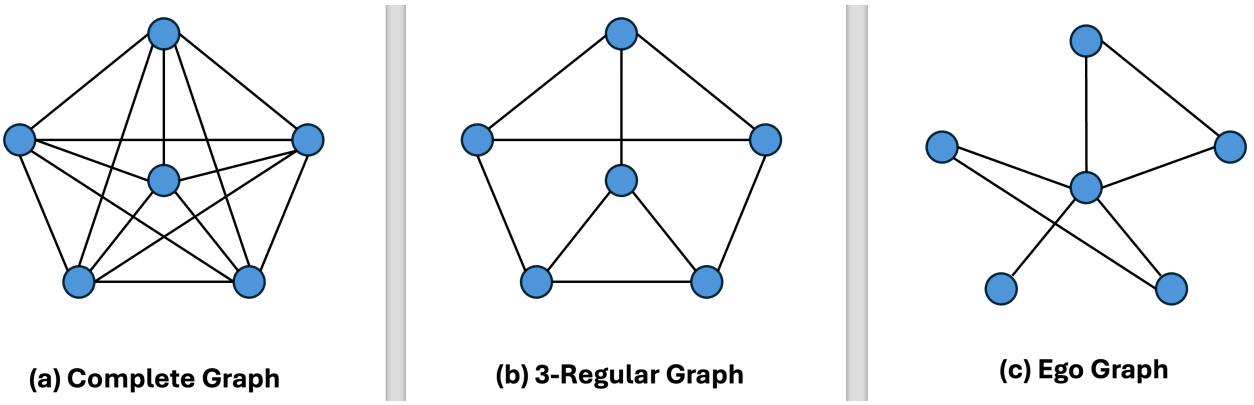

Figure 3. Illustration of graphs used for MaxCut with 6 nodes as an example: (a) Each node is connected to every other node. (b) Each node is connected to exactly three other nodes. (c) Central node and its neighbors.

limit. To transform the constrained problem into a form suitable for QAOA, we leverage the Knapsack class from the Qiskit Optimization module [36]. This tool automatically encodes the problem into a PUBO Hamiltonian by introducing penalty terms that enforce the weight constraint and by adding slack variables to properly represent feasible solutions. The penalty terms ensure that any violation of the weight constraint incurs a significant energy penalty, while the slack variables help maintain an unconstrained optimization landscape. The resulting Hamiltonian, which incorporates both the objective and the constraint penalties, is then directly used as the cost Hamiltonian in our framework. We generate random instances of the Knapsack problem by selecting a fixed number of items and assigning random item weights and values, with the weight limit set relative to the total item weights to ensure a meaningful optimization challenge.

For the cancer genomics PCBO benchmarks [42], we work with real-world datasets where each feature represents a biological measurement relevant to cancer classification tasks. Each bit in the bitstring corresponds to the inclusion or exclusion of a particular feature. The PCBO formulation captures not just individual feature relevance but also higher-order correlations—including pairwise and triplewise mutual information—between features and labels. This structure is encoded as a hypergraph, where the edge weights reflect the strength of these correlations. The resulting optimization task is to select a subset of M features from N total features that maximizes relevance while minimizing redundancy, subject to a Hamming weight constraint with a Lagrange penalty.

4.2 Evaluation Metrics

The primary metric used in this work is *Accuracy*, which quantifies the quality of our initialization relative to the true ground state. We define accuracy as:

$$\text{Accuracy} = \frac{E_{\text{init}}}{E_{\text{opt}}}, \tag{1}$$

where E_{init} is the objective value of the initial solution provided by different techniques, and E_{opt} is the objective value of the optimal ground state. A higher accuracy indicates that the initialization produces a state closer to the true optimum, serving as a more effective starting point for subsequent QAOA optimization.

In addition to accuracy, we introduce the *Search Space Reduction Factor (SSRF)* -just referred to *Reduction Factor (RF)* for short- to quantify the reduction in the number of solution states between initial points. Specifically, this metric captures how concentrated the quantum state becomes after initialization by comparing the number of unique bitstrings sampled:

$$\text{RF} = \frac{N_{\text{random}}}{N_{\text{init}}},$$

where N_{random} is the number of distinct bitstrings observed when sampling from a randomly initialized ma-QAOA ansatz, and N_{init} is the number of distinct bitstrings observed after applying our initialization framework. All measurements are performed using the same total number of shots. A higher RF indicates that the initialized state is more sharply concentrated over a smaller portion of the solution space, enabling more focused and efficient exploration during QAOA optimization.

For large-scale problem instances where the true ground state energy is intractable to compute, we introduce *Relative Improvement* to quantify the effectiveness of initialization. It is defined as

$$\text{Relative Improvement} = \frac{E_{\text{init}}}{E_{\text{rand}}}, \tag{2}$$

where E_{init} is the objective value of the state obtained from our Clifford-initialized framework, and E_{rand} is the objective

value of a randomly initialized state. A higher relative improvement indicates that the Clifford initialization provides a better starting point for optimization, effectively guiding the algorithm toward lower-energy regions even when the exact solution is unavailable.

4.3 Baseline and Evaluation Settings

The baseline for this study is the traditional QAOA, which employs random initialization. We compare this against alternative approaches, including the ma-QAOA ansatz initialized randomly, evaluating their performance under varying experimental conditions. We design our ma-QAOA ansatz with two repeating layers, corresponding to a depth of $p = 2$.

To assess the performance of these methods, we first create a QAOA ansatz with an input Hamiltonian using the Qiskit library and then free the parameters in each layer to be independent, to create the relaxed ma-QAOA ansatz.

For our evaluation described in subsection 5.4, We use Qiskit's optimization module with the COBYLA optimizer to explore the solution space. The framework also supports other optimizers that can be tuned for improved performance. Additionally, we select the top 5 unique CAFQA points for multi-start optimization, choosing them to be well-separated in the solution ranking to avoid redundant initializations.

These evaluations allow us to assess the robustness of each initialization method and ansatz under practical constraints.

5 Evaluation

5.1 Initialization Performance Across Combinatorial Benchmarks

We first assess the performance of our initialization strategy across the benchmark combinatorial optimization problems detailed in subsection 4.1, as visualized in Figures 4, 5, and 6. We present the initialization accuracies that are computed relative to the ground state energy of the Hamiltonian associated with each combinatorial problem.

5.1.1 Max-Cut Problem on Complete, 3-Regular, and Ego Graphs. In this section, we evaluate the quality of the optimal initialization point generated by our framework for the ma-QAOA ansatz applied to three distinct graph types in the Max-Cut problem. More information on Max-Cut and the types of graph benchmarks is discussed in subsubsection 4.1.1.

Figure 4 presents the CAFQA accuracy for each graph type across different numbers of qubits. We calculate the geometric mean over 10 weighted Max-Cut instances per graph type. When running our initialization framework, we allowed the optimizer to run for up to 48 hours—the maximum wall time permitted on our HPC system. In practice, however, most instances converged well before this limit

We see that the Clifford initialization technique can find strong initialization points for diverse graph structures and sizes. Among the tested graphs, the Clifford initialization

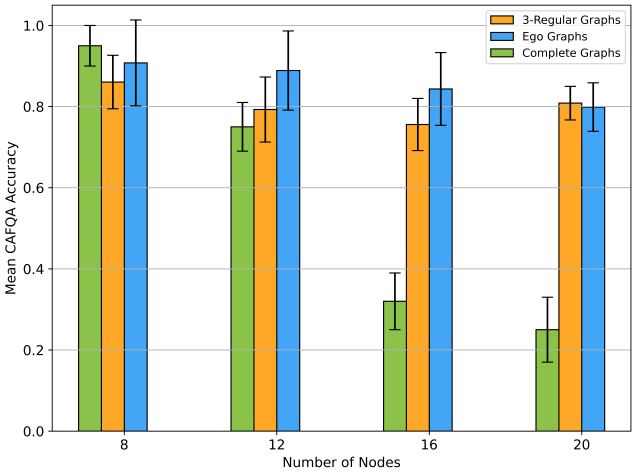

Figure 4. CAFQA Accuracy evaluated on Max-Cut Problems for various graphs

consistently achieves high proximity to the true ground state energy, suggesting it provides a strong starting point for optimization. The highest accuracy we achieve is 99.9% for the 3-Regular and Ego graphs, which have fewer edges between the nodes as compared to Complete Graphs. The mean accuracy decreases more starkly for the complete graphs as the number of nodes increases. This is expected as the edges between the nodes increase quadratically with the number of nodes, which makes the landscape extremely difficult to navigate. An increase in the number of edges directly translates to a larger set of $R_X(\theta)$ parameters in the QAOA ansatz, which enlarges the optimization space and makes the training process more challenging. We also note that there are pre-processing techniques [46] to further improve the quality of initialization that we don't explore in our work.

5.1.2 Knapsack Problem. Next, we assess initialization accuracy on a harder subclass of combinatorial optimization problems—PCBO (see Section subsubsection 4.1.2)—using the Knapsack problem as a representative example. In this section, we evaluate the impact of Clifford-based initialization on solving three representative Knapsack problem instances. We generate random 4, 9, and 12-item problems, chosen specifically to align with the number of qubits used in other QUBO and PCBO benchmarks once the problems are translated into Hamiltonian form. This enables consistent comparisons across problem classes. As with the Max-Cut case, we set a 48-hour wall time limit due to HPC constraints; however, most Knapsack runs finished well before this, indicating efficient convergence from the initial point.

Figure 5 presents the quality of the initial solution from Clifford initialization across four spaced-out problem sizes. We find that Clifford initialization yields strong overlap with the ground state, demonstrating its effectiveness even before any quantum optimization is performed. In the 4-item

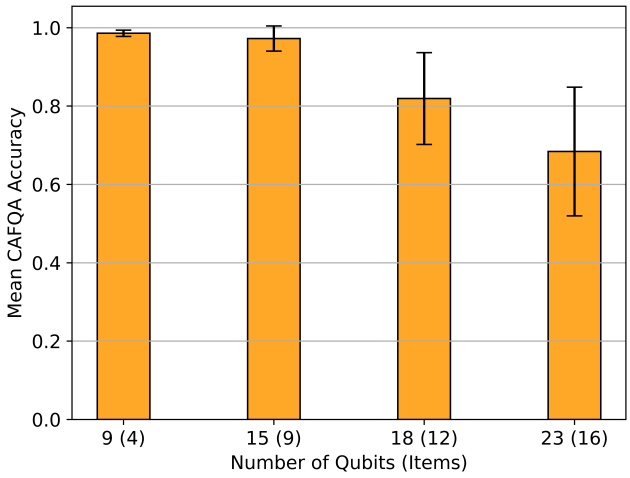

Figure 5. CAFQA Accuracy evaluated on Knapsack

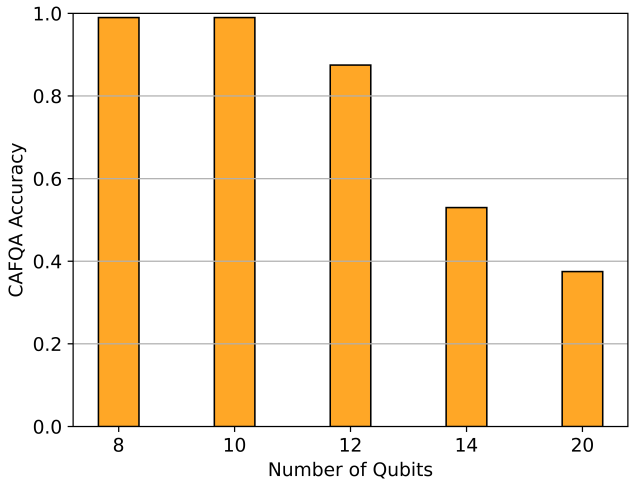

Figure 6. CAFQA Accuracy evaluated on Cancer Genomics Data

case (9 qubits), CAFQA recovers 99.9% of the true energy on average. For the 9-item instance (15 qubits), it retains 98.2% of the ground state energy. Although performance slightly decreases in the largest case (12 items, 18 qubits), recovering just 70.2% on average, this result demonstrates that the initialization strategy remains effective at producing low-energy configurations that are still significantly correlated with the ground state, even in high-dimensional, constrained Hamiltonians. The observed trend reflects the increasing difficulty of the optimization problem as the number of parameters grows, though longer runs of the genetic algorithm can mitigate this, as discussed in subsection 6.3.

5.1.3 Feature Selection on Cancer Genomics Data.

In this section, we evaluate the Clifford initialization by applying it to a real-world cancer biology challenge: selecting maximally informative yet non-redundant gene subsets from high-dimensional genomic data. We frame the feature selection task as a PCBO problem, encoding mutual information relevance scores between genes and tumor phenotypes alongside anti-redundancy constraints through third-order interaction terms.

The objective function, whose formulation is described in subsubsection 4.1.2, intentionally models complex gene regulatory dependencies via higher-order (k=3) correlations, creating a non-submodular optimization landscape that tests QAOA's ability to handle constrained combinatorial problems. This PCBO formulation represents an NP-hard optimization challenge derived from real biological data, providing a distinct test case from the conventional graph-based problems typically used to benchmark QAOA performance.

In our evaluations, we convert the PCBO object, which is represented as a hypergraph, into an Ising Hamiltonian formulation with the process described in subsubsection 4.1.2. In our study, the problems are subject to the constraint of choosing exactly 4 features out of the N total features in the datasets.

The benefits of Clifford-based initialization on this problem are illustrated in Figure 6. The best-performing CAFQA-initialized point achieves a remarkably high 94% accuracy, indicating close recovery of the ground state. While there is a decrease in performance as the number of qubits increases, the initialization strategy continues to yield high-quality solutions, maintaining strong accuracy even in complex landscapes. We also note that these numbers can be boosted with tailored computational techniques and increased run time of the genetic algorithm. These results demonstrate the value of CAFQA in steering the optimizer toward high-fidelity solutions, even as the problem complexity increases.

Together, these results demonstrate that combining the expressiveness of ma-QAOA with Clifford initialization (CAFQA) significantly improves the solution quality of the initial point. This approach reliably finds near-optimal solutions even in complex, structured optimization problems like those arising in molecular biology. Once again, we note that our current estimates are based on a maximum of 48 hours of runtime, but they can be improved with increased memory or compute resources, extended execution time, improved search heuristics, more efficient parallelization, or targeted exploration beyond Clifford space, as discussed in subsection 6.3.

5.2 Scaling to Larger Problem Instances

To assess the scalability of our initialization framework, we apply it to larger problem instances. Specifically, we evaluate performance on Max-Cut problems defined over 3-regular graphs with sizes ranging from 50 to 800 qubits, representing a challenging yet structured class of combinatorial optimization problems.

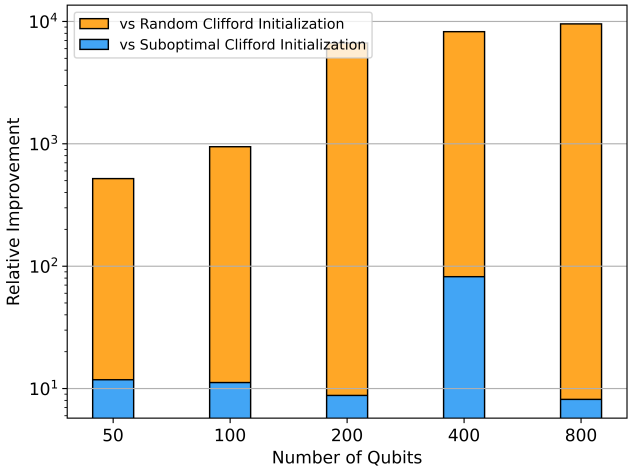

Figure 7. CAFQA Accuracy evaluated on Large 3-Regular Max-Cut instances

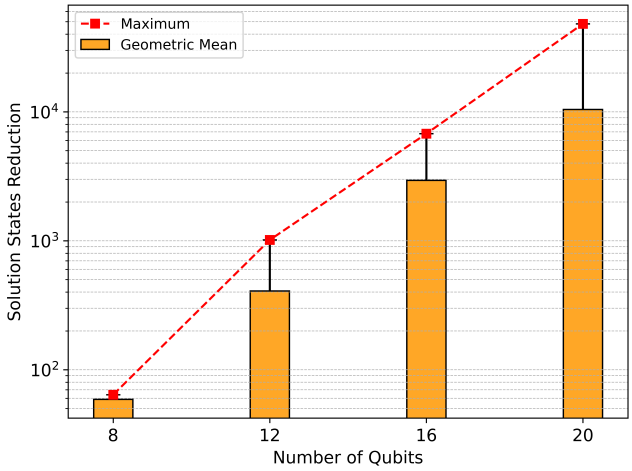

Figure 8. Factor of Reduction in Number of Initial Solution States in 3-Regular Max-Cut

To quantify initialization effectiveness at this scale, we compute the relative improvement, defined as the ratio between the energy of our Clifford initialized state and that of a randomly initialized state. As shown in Figure 7, our method achieves significantly better starting points, with improvements reaching up to 10,000× over random initialization. We also report the reduction achieved by selecting the first Clifford state found during the preliminary search, showing that even minimal exploration is enough to provide an initialization point that clearly outperforms random initialization.

These results highlight the critical role of informed initialization in guiding the optimizer toward low-energy regions of the solution space, even as the problem size grows substantially. Such large initial gains reduce the burden on quantum optimization, making the method especially useful in runtime-constrained or low-depth settings, which is the case for realistic quantum optimization workloads.

We note that since the exact ground-state energy is intractable for these large-scale problems, we rely on this metric as a practical proxy, which still provides a meaningful indicator of initialization quality.

5.3 Reduction in Solution Space with Initialization

To evaluate how effectively our initialization method narrows the solution search space, we compare the number of solution states explored by QAOA when starting from our initialization versus random initialization. We perform this analysis on both the Knapsack problem and 3-regular Max-Cut instances across several medium-sized problem sizes.

To accurately estimate the distribution over solution states, we sample the initialized QAOA states 100 million times. As shown in Figure 8, our initialization method achieves a maximum reduction of 48,132× and an average reduction of 10,448× in the number of solution states compared to random initialization on 3-regular Max-Cut instances. These graphs are challenging due to their uniform degree and highly non-trivial connectivity, and the observed reductions highlight the method's ability to generalize across complex, structured problem classes.

Similarly, in Figure 9, we observe strong reductions on the Knapsack problem, with a maximum reduction of 4,095× and an average reduction of 789× across the qubit sizes evaluated. Despite the added complexity of non-binary constraints and rugged energy landscapes, the initialization consistently concentrates amplitude into structured, low-energy regions of the solution space. Together, these results demonstrate that our framework enables QAOA to begin from compact, high-quality regions across a range of combinatorial optimization problems.

5.4 Post-Initialization QAOA Exploration

A crucial aspect of QAOA performance lies in its ability to effectively explore the quantum parameter space after initialization. The quality of this exploration directly impacts convergence behavior, distinguishing well-informed starting points from naive random initializations.

This is illustrated in Figure 10, where we see the benefits from the full QAOA run performed on the 10-qubit cancer genomic dataset. In the noiseless setting, well-chosen initialization points enable convergence within only a few thousand iterations, whereas randomly initialized points fail to converge even after 10,000 iterations. Moreover, the Vanilla QAOA ansatz with random initialization becomes rapidly trapped in a local minimum, underscoring the importance of high-quality starting parameters.

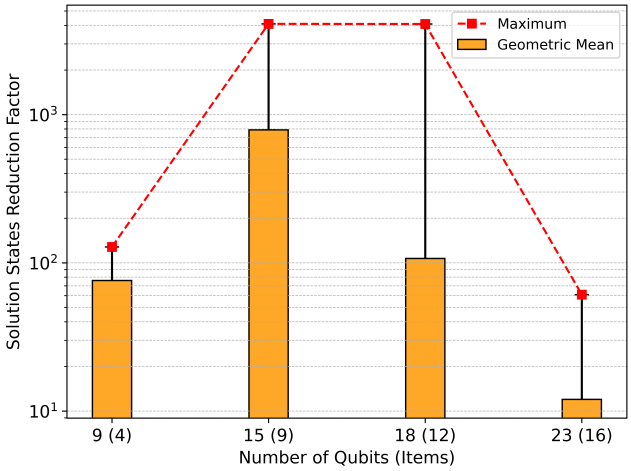

Figure 9. Factor of Reduction in Number of Initial Solution States in Knapsack Problem

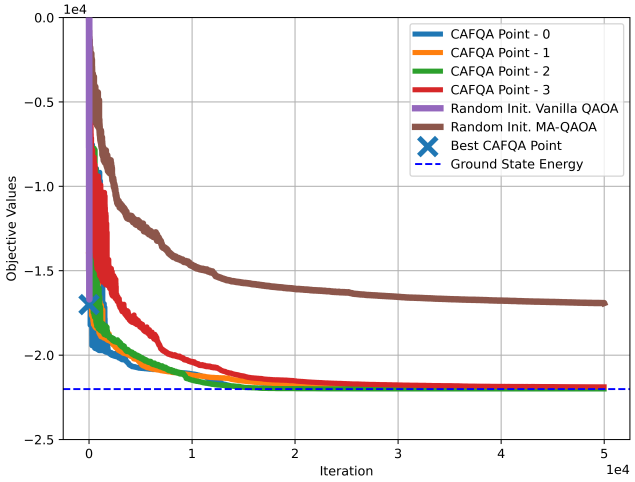

Figure 10. CAFQA Accuracy evaluated on 10-Qubit Cancer Genome Feature Selection Problem

In this experiment, we also employ a multi-start strategy with the Spread-Out method (which selects high-quality Clifford points with well-separated objective values to cover broad regions of the parameter space). Since a detailed treatment of multi-start strategies is provided in subsection 6.2, here we simply highlight that different starting points explore the quantum space along distinct trajectories yet ultimately converge, motivating a deeper investigation of multi-start approaches in this context.

CAFQA can also be uniquely suited to noisy models, enabling effective parameter initialization even under noisy simulations and guiding optimization toward promising regions despite hardware imperfections. While the focus of this work is not on quantum space exploration, which depends on multiple factors such as problem landscape, optimizer design, and device noise, Clifford search remains a critical tool for achieving rapid and reliable convergence. Its advantages are expected to scale substantially with larger problem sizes, which will soon be accessible as NISQ devices continue to advance. Because current NISQ devices are highly prone to noise, the quality of Clifford-based initialization can be degraded during full QAOA execution. In such settings, CAFQA initialization offers a more targeted starting point, reducing susceptibility to noise and barren plateaus, and thereby enabling faster and more reliable convergence.

We also note that reducing reliance on repeated quantum executions offers a major economic benefit. Prior work [19] has shown that variational algorithms on quantum cloud platforms can incur prohibitive costs, often thousands of dollars for problems of only modest scale, underscoring the necessity of methods like CAFQA that deliver both computational and financial efficiency.

6 Discussion

6.1 Initialization Strategies Beyond Max-Cut-Specific Approaches

Prior work on parameter initialization for QAOA and related VQA algorithms has largely focused on Max-Cut, with techniques such as Red-QAOA [46] and Angle Rounding [49] providing instance-specific heuristics. These methods primarily aim to locate regions of parameter space where optimal solutions are empirically concentrated, rather than explicitly targeting low-energy points. While effective within restricted settings, such strategies face important limitations: they provide no theoretical error guarantees, require exponential classical overhead, and demonstrate poor generalization beyond Max-Cut-like or unweighted graphs. For example, Red-QAOA's degree-based reduction criterion is not guaranteed to transfer reliably across broader weighted problem classes, and Angle Rounding's validation is restricted to small-scale instances, raising concerns about the scalability of its quantization scheme.

In particular, Red-QAOA illustrates the risks of heuristic transfer: when reduced graphs fail to preserve weighted interactions, fidelity can degrade substantially. Such behavior underscores the need for initialization approaches that remain robust under realistic, weighted scenarios.

Expanding beyond these specialized heuristics, our framework supports a much broader class of problem instances, including those that better reflect practical applications. Our framework suggests that initialization need not be confined to unweighted Max-Cut-like benchmarks but can instead extend toward domains where combinatorial structure and weighted interactions play a central role.

6.2 Multi-Start for Efficient Exploration

Optimization landscapes in QAOA often contain many local optima, making initialization critical for avoiding premature convergence. A multi-start strategy—where local optimization is initialized from multiple, well-chosen points—can improve robustness by exploring diverse regions of the parameter space and reducing dependence on any single initial guess. This approach also highlights degenerate optima, which may offer practical flexibility.

We experimented with three variants: Spread-Out, which selects angles with well-separated objective values to cover broad regions of the space; Clustered, which refines consecutive or closely spaced angles in a promising region; and Degenerate, which targets multiple configurations achieving the same minimum objective value.
While a full exploration of this direction lies beyond the scope of our present study, preliminary results suggest that multi-start strategies can meaningfully alter landscape exploration, with different optimizers favoring different regions. The quantum exploration of the Spread-Out technique with five starting points is illustrated in Figure 10.

6.3 Optimization and Resource Scaling

The performance of the Clifford search is closely tied to the choice of classical optimizer. In our current implementation, we employ a genetic algorithm, which balances global exploration with robustness against noisy objective evaluations. While this choice has proven effective for identifying high-quality initial points, more sophisticated optimizers could further enhance performance, particularly under larger or more complex problem settings. Importantly, additional computational resources and runtime can amplify the benefits of such optimizers by enabling deeper search and broader parallel exploration. Thus, optimizer selection, together with resource allocation, represents a key axis for improving the quality of initialization and scaling our framework to more challenging domains.

6.4 Future Work

Several directions emerge naturally from our study. First, a systematic investigation of multi-start strategies could clarify their role in balancing global exploration with local refinement, particularly when combined with diverse optimizers. Second, expanding the benchmark space beyond Max-Cut-like instances to weighted and application-oriented problems would test the generality and robustness of our initialization scheme. Third, parameter transfer techniques in QAOA—while promising for accelerating convergence by reusing solutions across related problem instances—currently struggle to account for weighted graphs [17]. Exploring how such transfer strategies could be adapted or integrated with our framework represents an important open question. Finally, hybrid methods that combine classical heuristics with quantum-aware initialization may open pathways toward more resource-efficient and scalable QAOA deployments.

7 Conclusion

The Quantum Approximate Optimization Algorithm (QAOA) is a promising approach for solving combinatorial optimization problems using quantum computers. In this work, we introduce an initialization framework to improve the performance of QAOA by leveraging the increased expressivity of the relaxed QAOA ansatz to efficiently search for optimal Clifford states that serve as high-quality initial points in QAOA. We evaluated this framework against a wide range of combinatorial problems, including Max-Cut, the Knapsack problem, and real-world cancer genome datasets, and found that our method consistently yielded better initialization in terms of reaching higher overlap with the ground state energy.

While our results highlight the effectiveness of Clifford initialization in improving QAOA performance, there are still areas for further investigation. Future work includes exploring more advanced optimizers to better handle noisy environments and to refine initialization techniques for complex problem instances. Additionally, we plan to investigate the application of QAOA to even larger datasets and real-world combinatorial problems. Related work in the field has shown promising results, and our work aims to contribute further insights into optimizing QAOA for practical quantum applications.

Acknowledgment

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Accelerated Research in Quantum Computing under Award Number DE-SC0025633. This research used resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231 using NERSC award ASCR-ERCAP0033197.

References

[1] Carlos Ansótegui, Maria Luisa Bonet, and Jordi Levy. 2013. SAT-based MaxSAT algorithms. *Artificial Intelligence* 196 (2013), 77–105.

[2] Martin Anthony, Endre Boros, Yves Crama, and Aritanan Gruber. 2017. Quadratization of higher degree pseudo-Boolean functions. In *International Conference on Integer Programming and Combinatorial Optimization*. Springer, 458–469. https://doi.org/10.1007/978-3-319-59250-3_37

[3] Boaz Barak, Ankur Moitra, Ryan O'Donnell, Prasad Raghavendra, Oded Regev, David Steurer, Luca Trevisan, Aravindan Vijayaraghavan, David Witmer, and John Wright. 2015. Beating the random assignment on constraint satisfaction problems of bounded degree. *arXiv* (Aug. 2015). https://doi.org/10.48550/arXiv.1505.03424

[4] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. 2017. Quantum machine learning. *Nature* 549, 7671 (2017), 195–202.

[5] Kostas Blekos, Dean Brand, Andrea Ceschini, Chiao-Hui Chou, Rui-Hao Li, Komal Pandya, and Alessandro Summer. 2024. A review on quantum approximate optimization algorithm and its variants. *Physics Reports* 1068 (2024), 1–66.

[6] Hans L Bodlaender and Klaus Jansen. 2000. On the complexity of the maximum cut problem. *Nordic Journal of Computing* 7, 1 (2000), 14–31.

[7] Endre Boros and Peter L. Hammer. 2002. Pseudo-boolean optimization. *Discrete Applied Mathematics* 123, 1-3 (2002), 155–225. https://doi.org/10.1016/S0166-218X(01)00341-9

[8] Endre Boros, Peter L Hammer, and Gabriel Tavares. 2007. Local search heuristics for quadratic unconstrained binary optimization (QUBO). *Journal of Heuristics* 13, 2 (2007), 99–132.

[9] Ming-Te Chao and John Franco. 1986. Probabilistic analysis of two heuristics for the 3-satisfiability problem. *SIAM J. Comput.* 15, 4 (1986), 1106–1118.

[10] M. H. Cheng, K. E. Khosla, C. N. Self, M. Lin, B. X. Li, A. C. Medina, and M. S. Kim. 2022. Clifford Circuit Initialisation for Variational Quantum Algorithms. arXiv:2207.01539 [quant-ph] https://arxiv.org/abs/2207.01539

[11] Chi-Ning Chou, Peter J. Love, Juspreet Singh Sandhu, and Jonathan Shi. 2022. Limitations of Local Quantum Algorithms on Random Max-k-XOR and Beyond. https://doi.org/10.48550/arXiv.2108.06049

[12] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. 2014. A Quantum Approximate Optimization Algorithm. arXiv:1411.4028 [quant-ph]

[13] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. 2014. A Quantum Approximate Optimization Algorithm. https://doi.org/10.48550/arXiv.1411.4028

[14] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. 2015. A Quantum Approximate Optimization Algorithm Applied to a Bounded Occurrence Constraint Problem. *arXiv* (June 2015). https://doi.org/10.48550/arXiv.1412.6062

[15] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Leo Zhou. 2022. The Quantum Approximate Optimization Algorithm and the Sherrington-Kirkpatrick Model at Infinite Size. *Quantum* 6 (July 2022), 759. https://doi.org/10.22331/q-2022-07-07-759

[16] Ahmed Fawzy Gad. 2021. PyGAD: An Intuitive Genetic Algorithm Python Library. arXiv:2106.06158 [cs.NE] https://arxiv.org/abs/2106.06158

[17] Alexey Galda, Xiaoyuan Liu, Danylo Lykov, Yuri Alexeev, and Ilya Safro. 2021. Transferability of optimal QAOA parameters between random graphs. In *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*. 171–180. https://doi.org/10.1109/QCE52317.2021.00034

[18] Fred Glover, Gary Kochenberger, and Yu Du. 2019. A tutorial on formulating and using QUBO models. *arXiv preprint arXiv:1811.11538* (2019). https://arxiv.org/abs/1811.11538

[19] Andi Gu, Angus Lowe, Pavel A Dub, Patrick J Coles, and Andrew Arrasmith. 2021. Adaptive shot allocation for fast convergence in variational quantum algorithms. *arXiv preprint arXiv:2108.10434* (2021).

[20] Gurobi Optimization, LLC. 2024. Gurobi Optimizer Reference Manual. https://www.gurobi.com

[21] Stuart Hadfield, Zhihui Wang, Bryan O'Gorman, Eleanor G. Rieffel, Davide Venturelli, and Rupak Biswas. 2019. From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz. *Algorithms* 12, 2 (Feb. 2019), 34. https://doi.org/10.3390/a12020034

[22] M. B. Hastings. 2019. Classical and Quantum Bounded Depth Approximation Algorithms. *arXiv* (Aug. 2019). https://doi.org/10.48550/arXiv.1905.07047

[23] Rebekah Herrman, Phillip C. Lotshaw, James Ostrowski, Travis S. Humble, and George Siopsis. 2021. Multi-angle Quantum Approximate Optimization Algorithm. arXiv:2109.11455 [quant-ph] https://arxiv.org/abs/2109.11455

[24] IBM. 2024. *IBM ILOG CPLEX Optimization Studio*. https://www.ibm.com/products/ilog-cplex-optimization-studio.

[25] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M Chow, and Jay M Gambetta. 2017. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* 549, 7671 (2017), 242–246.

[26] Hans Kellerer, Ulrich Pferschy, and David Pisinger. 2004. Multidimensional knapsack problems. In *Knapsack problems*. Springer, 235–283.

[27] Youngseok Kim, Andrew Eddins, Sajant Anand, Ken Xuan Wei, Ewout van den Berg, Sami Rosenblatt, Hasan Nayfeh, Yantao Wu, Michael Zaletel, Kristan Temme, and Abhinav Kandala. 2023. Evidence for the Utility of Quantum Computing before Fault Tolerance. *Nature* 618, 7965 (June 2023), 500–505. https://doi.org/10.1038/s41586-023-06096-3

[28] Gary Kochenberger, Jin-Kao Hao, Fred Glover, Mark Lewis, Zhipeng Lü, Haibo Wang, and Yang Wang. 2014. The unconstrained binary quadratic programming problem: a survey. *Journal of Combinatorial Optimization* 28 (2014), 58–81. https://doi.org/10.1007/s10878-014-9734-0

[29] Cedric Yen-Yu Lin and Yechao Zhu. 2016. Performance of QAOA on Typical Instances of Constraint Satisfaction Problems with Bounded Degree. *arXiv* (Jan. 2016). https://doi.org/10.48550/arXiv.1601.01744

[30] Kunal Marwaha. 2021. Local classical MAX-CUT algorithm outperforms $p=2$ QAOA on high-girth regular graphs. *Quantum* 5 (April 2021). https://doi.org/10.22331/q-2021-04-20-437

[31] Nikolaj Moll, Panagiotis Barkoutsos, Lev Bishop, Jerry Chow, Andrew Cross, Daniel Egger, Stefan Filipp, Andreas Fuhrer, Jay Gambetta, Marc Ganzhorn, Abhinav Kandala, Antonio Mezzacapo, Peter Müller, Walter Riess, Gian Salis, John Smolin, Ivano Tavernelli, and Kristan Temme. 2018. Quantum optimization using variational algorithms on near-term quantum devices. *Quantum Science and Technology* 3 (06 2018). https://doi.org/10.1088/2058-9565/aab822

[32] Julius Beneoluchi Odili. 2017. Combinatorial optimization in science and engineering. *Current Science* (2017), 2268–2274.

[33] Joe O'Gorman and Earl T. Campbell. 2017. Quantum computation with realistic magic-state factories. *Physical Review A* 95, 3 (Mar 2017). https://doi.org/10.1103/physreva.95.032338

[34] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J Love, Alán Aspuru-Guzik, and Jeremy L O'brien. 2014. A variational eigenvalue solver on a photonic quantum processor. *Nature communications* 5 (2014), 4213.

[35] John Preskill. 2018. Quantum Computing in the NISQ era and beyond. *Quantum* 2 (2018), 79.

[36] Qiskit contributors. 2023. Qiskit: An Open-source Framework for Quantum Computing. https://doi.org/10.5281/zenodo.2573505

[37] Ronald L Rardin. 1998. *Optimization in operations research*. Vol. 166. Prentice Hall Upper Saddle River, NJ.

[38] Gokul Subramanian Ravi, Pranav Gokhale, Yi Ding, William M. Kirby, Kaitlin N. Smith, Jonathan M. Baker, Peter J. Love, Henry Hoffmann, Kenneth R. Brown, and Frederic T. Chong. 2022. CAFQA: Clifford Ansatz For Quantum Accuracy. https://doi.org/10.48550/ARXIV.2202.12924

[39] Gokul Subramanian Ravi, Pranav Gokhale, Yi Ding, William M. Kirby, Kaitlin N. Smith, Jonathan M. Baker, Peter J. Love, Henry Hoffmann, Kenneth R. Brown, and Frederic T. Chong. 2023. CAFQA: A classical simulation bootstrap for variational quantum algorithms. In *ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*. https://doi.org/10.48550/ARXIV.2202.12924

[40] Stefan H. Sack and Maksym Serbyn. 2021. Quantum annealing initialization of the quantum approximate optimization algorithm. *Quantum* 5 (July 2021), 491. https://doi.org/10.22331/q-2021-07-01-491

[41] Abdelkader Sbihi and Richard W Eglese. 2010. Combinatorial optimization and green logistics. *Annals of Operations Research* 175, 1

(2010), 159–175.

[42] Dhirpal Shah, Mariesa Teo, Ryan A. Robinett, Sophia Madejski, Zachary Morrell, Siddhi Ramesh, Colin Campbell, Bharath Thotakura, Victory Omole, Ben Hall, Aram W. Harrow, Teague Tomesh, Alexander T. Pearson, Frederic T. Chong, and Samantha J. Riesenfeld. 2025. Toward Quantum-Enabled Biomarker Discovery: An Outlook from Q4Bio. arXiv:2509.25904 [quant-ph] https://arxiv.org/abs/2509.25904

[43] Michael Streif and Martin Leib. 2020. Training the quantum approximate optimization algorithm without access to a quantum processing unit. *IOP* 5, 3 (May 2020), 034008. https://doi.org/10.1088/2058-9565/ab8c2b Publisher: IOP Publishing.

[44] Robert Endre Tarjan and Anthony E Trojanowski. 1977. Finding a maximum independent set. *SIAM J. Comput.* 6, 3 (1977), 537–546.

[45] Matthew Treinish, Ivan Carvalho, Georgios Tsilimigkounakis, and Nahum Sá. 2022. rustworkx: A High-Performance Graph Library for Python. *Journal of Open Source Software* 7, 79 (Nov. 2022), 3968. https://doi.org/10.21105/joss.03968

[46] Meng Wang, Bo Fang, Ang Li, and Prashant J Nair. 2024. Red-qaoa: Efficient variational optimization through circuit reduction. In *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*. 980–998.

[47] Qingfeng Wang, Liudmila Zhukas, Qiang Miao, Aniket S. Dalvi, Peter J. Love, Christopher Monroe, Frederic T. Chong, and Gokul Subramanian Ravi. 2024. Demonstration of a CAFQA-bootstrapped Variational Quantum Eigensolver on a Trapped-Ion Quantum Computer. arXiv:2408.06482 [quant-ph] https://arxiv.org/abs/2408.06482

[48] Zhihui Wang, Stuart Hadfield, Zhang Jiang, and Eleanor G. Rieffel. 2018. Quantum Approximate Optimization Algorithm for MaxCut: A Fermionic View. *Physical Review A* 97, 2 (Feb. 2018), 022304. https://doi.org/10.1103/PhysRevA.97.022304

[49] Anthony Wilkie, James Ostrowski, and Rebekah Herrman. 2024. An angle rounding parameter initialization technique for ma-QAOA. *arXiv preprint arXiv:2404.10743* (2024).

[50] Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. 2020. Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices. *Phys. Rev. X* 10 (Jun 2020), 021067. Issue 2. https://doi.org/10.1103/PhysRevX.10.021067