CLASSICAL STATE PREPARATION FOR VARIATIONAL QUANTUM ALGORITHMS VIA REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Variational Quantum Algorithms (VQAs) potentially offer a pathway to practical quantum advantage, but their optimization is heavily hindered by barren plateaus and numerous local minima. While classically simulable Clifford circuits can warm-start VQAs to accelerate convergence, existing heuristic-based initialization methods struggle to scale within vast combinatorial search spaces. We address this bottleneck with CRiSP (a Clifford Reinforcement Learning agent for State Preparation), which casts Clifford prefix selection as a sequential decision-making problem. CRiSP utilizes Neural-Guided Monte Carlo Tree Search, driven by a Transformer-based policy trained via self-play, to insert learned Clifford gates before fixed parameterized rotations. This enables the construction of high-quality initial states entirely through polynomial-time classical stabilizer simulation without altering the underlying circuit architecture. By integrating a curriculum learning strategy that progressively expands the search horizon, the agent efficiently scales to deep circuits. Evaluated on QAOA benchmarks of up to 22 qubits and $1{,}370$ parameters, CRiSP outperforms state-of-the-art Clifford initialization methods by a mean of $3.19\times$ (max $45.02\times$) in average energy accuracy and $2.41\times$ (max $16.01\times$) in best-achieved energy accuracy. Assessments on VQE tasks further demonstrate the framework's robustness and generalizability.

1 INTRODUCTION

The potential for quantum computing to deliver a computational edge is particularly strong in domains such as chemistry (Kandala et al., 2017), optimization (Moll et al., 2018), and machine learning (Biamonte et al., 2017). The field is currently evolving past the era of Noisy Intermediate-Scale Quantum (NISQ) hardware (Preskill, 2018), approaching a transition phase characterized by devices with thousands of physical qubits, augmented by advanced error mitigation and early forms of quantum error correction (J. Preskill, 2025; Google, 2024; IBM, 2024; Katabarwa et al., 2024). While deep algorithms such as Shor's or Grover's (Shor, 1997; Grover, 1996) remain out of reach due to strict fault-tolerance requirements (O'Gorman & Campbell, 2017), Variational Quantum Algorithms (VQAs) (Cerezo et al., 2021) allow shallow, hardware-native circuit designs that can help mitigate noise exposure. Operating as hybrid quantum-classical frameworks, VQAs employ a classical optimizer to iteratively update a parameterized quantum circuit, navigating the solution landscape to minimize a target energy expectation value ($\langle H_c \rangle$) (Peruzzo et al., 2014; Kim et al., 2023; Farhi et al., 2014). Improving the efficiency of VQA execution on early fault-tolerant hardware is therefore an important near-term goal.

On early fault-tolerant hardware, the cost of each VQA iteration is dominated by non-Clifford gates, which carry substantial overhead (Kremer et al., 2025). Since every circuit evaluation is expensive, limiting the total number of VQA evaluations is critical. However, VQA optimization landscapes are notoriously difficult to navigate, frequently forcing optimizers to evaluate thousands of expensive circuits just to make marginal progress. A strong starting point can bypass these optimization traps and substantially reduce the number of circuit executions, which motivates classically tractable state-preparation and initialization strategies (Ravi et al., 2023; Bharadwaj et al., 2026). Among simulation-compatible gate sets, the Clifford group provides a theoretically principled discrete search space.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

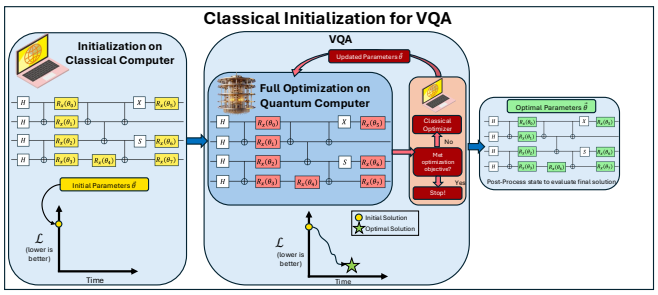

Figure 1: Classical initialization for VQA.

Since Clifford circuits can be simulated in polynomial time classically, they enable exact computation of the initial energy expectation value without requiring quantum hardware. Existing Clifford initialization methods exploit this simulability through Bayesian or genetic-algorithm-based heuristics (Ravi et al., 2023; Bharadwaj et al., 2026). However, the Clifford search space grows exponentially with the number of circuit parameters, and on challenging landscapes these heuristic search techniques often struggle to navigate it. Figure 1 gives an overview of VQA with classical initialization.

To overcome the scaling limitations of standard heuristics, we formulate discrete initialization as a sequential decision-making problem suited for reinforcement learning (RL) (Sutton et al., 1998). Specifically, we take an ansatz with fixed rotation gate positions and prepend a discrete Clifford gate to each rotation to act as a prefix modifier. When all continuous parameters are initialized to $\theta = 0$, the rotation gates reduce to the identity up to a global phase, so the circuit state is determined entirely by the chosen Clifford prefixes. This state can be simulated exactly on a classical machine and serves directly as the starting point for VQA optimization. Choosing the prefixes is thus a combinatorial search task where RL has shown strong results (Fawzi et al., 2022; Mazyavkina et al., 2021).

We present a Clifford Reinforcement Learning agent for State Preparation (CRiSP), a discrete-gate warm-start framework for VQAs. Using Neural-Guided Monte Carlo Tree Search (NG-MCTS) (Silver et al., 2017), the agent searches the discrete Clifford space entirely through classical simulation, selecting sequential Clifford gates that minimize the Hamiltonian expectation value. At each decision step, the search is driven by a Transformer-based (Vaswani et al., 2017) policy and value network, which encodes the current gate sequence to output prior probabilities for the next gate alongside an estimated return. The episodic reward is the negative energy expectation value of the synthesized circuit, and the agent uses two types of replay buffers (Fawzi et al., 2022) to improve training efficiency. For tasks with large parameter counts and sparse energy signals, we further introduce an incremental horizon curriculum that gradually expands the search horizon during training.

While CRiSP applies to any VQA, we focus primarily on the Quantum Approximate Optimization Algorithm (QAOA) (Farhi et al., 2014), benchmarking against SPIQ (Bharadwaj et al., 2026), the state-of-the-art Clifford initialization framework, on combinatorial optimization problems including MaxCut, Knapsack, and feature selection on high-order medical datasets. We also evaluate CRiSP on the Variational Quantum Eigensolver (VQE) (Peruzzo et al., 2014) to show that the approach generalizes beyond QAOA.

Overall, this work makes three primary contributions. First, we formalize Clifford prefix initialization in VQA as a sequential decision-making problem over the discrete gate space, developing an RL framework that is classically simulable and avoids exponential state representations. Second, we introduce CRiSP, a Transformer-guided tree search agent trained via self-play over the Clifford prefix space, incorporating a curriculum (Bengio et al., 2009) with stepwise-increasing episode length to handle long-horizon tasks. Third, we evaluate CRiSP on QAOA benchmarks with up to 22 qubits and 1,370 circuit parameters. Given the same evaluation budget, CRiSP consistently outperforms SPIQ, improving average energy accuracy by a mean of $3.19\times$ (max $45.02\times$) and best achieved energy accuracy by a mean of $2.41\times$ (max $16.01\times$). These gains come entirely from classical computation, giving the downstream VQA a better starting point at no additional quantum cost.

2 Background and Related Work

Variational quantum algorithms and optimization traps. VQAs are iterative hybrid quantum-classical methods, including the VQE (Peruzzo et al., 2014) and the QAOA (Farhi et al., 2014). VQE is widely used in quantum chemistry and condensed matter physics, whereas QAOA targets

combinatorial optimization problems, particularly Quadratic Unconstrained Binary Optimization (QUBO). In each VQA iteration, a classical optimizer (Lavrijsen et al., 2020; Spall, 2002) updates a parameterized quantum circuit (ansatz) to approximate the ground-state energy of a problem Hamiltonian (McClean et al., 2016). However, achieving rapid convergence is notoriously difficult. VQA optimization landscapes feature numerous local minima and barren plateaus (McClean et al., 2018; Sack & Serbyn, 2021; Blekos et al., 2024)—regions where gradients vanish exponentially as circuit size and depth increase. These traps make convergence from arbitrary initial points highly unreliable, potentially curtailing quantum benefits.

Fault-tolerant costs and classical simulability. In fault-tolerant quantum computing (FTQC) (Shor, 1996), circuit cost is dominated by non-Clifford gates (Kremer et al., 2025). Arbitrary-angle rotations such as $R_x(\theta)$ and $R_z(\theta)$ must be approximately decomposed into a universal gate set, most commonly Clifford+T, using compilers such as Gridsynth (Ross & Selinger, 2016) or tensor-network-based synthesizers (Hao et al., 2025). Each resulting T gate requires costly magic state distillation or cultivation (Bravyi & Kitaev, 2005; Gidney et al., 2024), making every VQA iteration expensive. It is therefore attractive to do as much of the optimization as possible classically, before using quantum hardware. Classical simulation, however, is efficient only in restricted regions of the state space. The Clifford group defines one such region, and it ties directly to the cost model above: when θ is a multiple of $\pi/2$, a rotation reduces to a Clifford gate and requires no T gates. By the Gottesman-Knill theorem, Clifford circuits can be simulated in polynomial time and space with stabilizer tableau methods (Gottesman, 1998; Aaronson & Gottesman, 2004), so their energy expectation values can be computed exactly without quantum hardware (Gidney, 2021). Other tractable regions include low-entanglement states simulated by tensor networks (Rudolph et al., 2023) and circuits with polynomially sized dynamical Lie algebras (Goh et al., 2025). These are complementary to the Clifford space, which is the focus of this work.

Classical VQA initialization. VQA performance is highly sensitive to initialization (Grant et al., 2019). Early heuristics either exploit ansatz-specific structure, as in INTERP (Zhou et al., 2020) and TQA (Sack & Serbyn, 2021) for QAOA, or mitigate barren plateaus without using the target Hamiltonian, as in identity-block initialization (Grant et al., 2019). Clifford-based methods instead use stabilizer simulation to evaluate the target energy exactly before any quantum hardware is used. CAFQA (Ravi et al., 2023) and Cheng et al. (2025) search the Clifford space with Bayesian optimization and simulated annealing, respectively. Clapton (Seifert et al., 2024) adds noise-aware transformations and a genetic algorithm, and SPIQ (Bharadwaj et al., 2026), our state-of-the-art baseline, extends this approach to QAOA (Farhi et al., 2014). However, these randomized or population-based heuristics struggle to scale, since the Clifford search space grows exponentially with the number of rotation gates. CRiSP instead trains a Transformer-guided tree search agent that selects a Clifford prefix for each rotation gate, so the VQA begins from a classically computed stabilizer state and refines it on quantum hardware rather than searching from scratch. Appendix A discusses initialization based on the other tractable regions above, as well as circuit-construction methods.

RL for quantum search. RL (Sutton et al., 1998) shows strong potential in quantum applications, from error correction (Sivak et al., 2025) to circuit optimization (Weiden et al., 2025). Search-based methods like MCTS (Browne et al., 2012) and AlphaZero (Silver et al., 2017) effectively navigate massive combinatorial spaces. Within the quantum domain, AlphaTensor-Quantum (Ruiz et al., 2025) applies learned search to minimize T-counts. Most similarly, MCTS-QAOA (Yao et al., 2022) jointly optimizes discrete and continuous variables, but relies on costly quantum circuit evaluations during search. In contrast, CRiSP performs prefix selection entirely within the classically simulable Clifford space. By training a Transformer-based policy-value network, we enable principled exploration while eliminating quantum hardware overhead during initialization.

3 METHODS

In this section, we describe the configuration of the reinforcement learning (RL) environment and agent for the VQA state preparation task, as depicted in Fig. 2, given a problem Hamiltonian and a fixed underlying circuit structure. We first specify the environment's state, action, and reward formulations. We then introduce CRiSP, the NG-MCTS agent that performs stepwise inference to identify the Clifford prefix configuration yielding the lowest circuit energy, alongside its self-

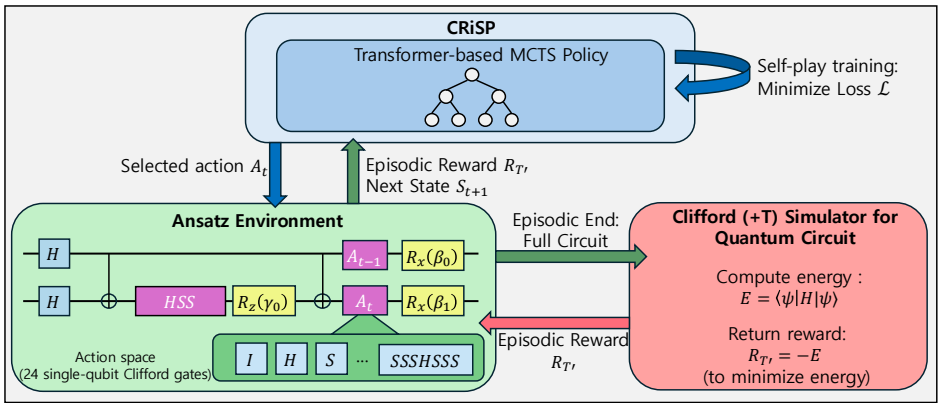

Figure 2: Overview of our proposed neural-guided framework (CRiSP) for state preparation in VQAs. The HSS gate positioned before $R_z(\gamma_0)$ represents the action A_{t-2}. The environment depicts a multi-angle QAOA ansatz (Herrman et al., 2022). The circuit's final state $|\psi\rangle$ is generated by setting $\gamma_0 = \beta_0 = \beta_1 = 0$.

play training procedure and a curriculum variant. Formally, given an n-qubit cost Hamiltonian H_c and a fixed ansatz containing $D = \text{poly}(n)$ rotational gates, such as hardware-efficient and QAOA architectures (Kandala et al., 2017; Farhi et al., 2014; Cerezo et al., 2021), we cast the state preparation process as a sequential decision-making problem defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma)$. The agent's objective is to construct a gate sequence that prepares a quantum state $|\psi\rangle$ minimizing the expected energy $E = \langle H_c \rangle = \langle\psi| H_c |\psi\rangle$.

3.1 RL ENVIRONMENT

The **state** s_t at step t in the RL environment is defined as the ordered sequence of Clifford gates assigned to the first t prefix positions, preceded by a `START` token. Since the sequence length scales linearly with D, this representation avoids the exponential dimensionality of storing the raw quantum state vector. Transitions are deterministic, where s_{t+1} is formed by appending gate action a_t to s_t. An episode begins with the `START`-only state and terminates either when all gate prefix positions have been assigned ($t = D$) or when the agent reaches the maximum step count T' ($t = T'$).

The **action space** comprises all 24 single-qubit Clifford gates (Epstein et al., 2014), each representing a candidate prefix for the corresponding rotational gate, since the target gates are single-qubit operations. We consider all axes regardless of the associated rotation gate's axis, opening the possibility of configurations that yield favorable energy through interactions with other circuit components. While exhaustive search over the discrete Clifford space of 24^D configurations is intractable for practical system sizes, the proposed RL agent addresses this by concentrating exploration on promising regions, effectively reducing the number of nodes evaluated during tree traversal.

The **reward** R is provided upon episode termination and is defined as the negative energy expectation value of the synthesized circuit under H_c, $R = -E = -\langle H_c \rangle = -\langle\psi| H_c |\psi\rangle$, where the resulting state $|\psi\rangle$ is obtained by setting all continuous parameters to $\theta = 0$. Intermediate transitions receive zero reward, and the discount factor is set to $\gamma = 1$, reflecting the episodic task structure. Since all gates in the synthesis procedure are Clifford, the reward can be evaluated efficiently via a classical stabilizer tableau simulator such as Stim (Gidney, 2021). Because the magnitude of $\langle H_c \rangle$ varies substantially across different Hamiltonians (from sub-unit values to over 10^4), directly using the raw reward as a training target introduces significant gradient instability. To address this, we normalize raw reward targets in the buffers using a running mean μ and standard deviation σ as $R_{\text{norm}} = \frac{R - \max(\mu, 0)}{\sigma + \epsilon} - 1$, where ϵ is a small numerical offset. The -1 term penalizes $R = 0$ cases, encouraging the agent to seek negative energies (positive rewards). The normalized reward is subsequently clipped to $[-1, 1]$ to suppress instability from outlier values. This adaptive rescaling continuously rebalances the reward range, sustaining active search for higher-return configurations.

3.2 THE CRISP FRAMEWORK

We now introduce CRiSP, a Clifford Reinforcement Learning agent for State Preparation in VQA, which utilizes Neural-Guided MCTS (NG-MCTS) to discover the best gate action in each step.

NG-MCTS procedure. At each step within an episode, CRiSP selects the next gate action via MCTS guided by a neural network that estimates the expected return of a given state and the prior action probabilities used in simulation (Silver et al., 2017). The search proceeds through three stages, namely selection, expansion, and backpropagation. In the **selection** phase, the agent traverses the tree by iteratively choosing actions according to the following upper confidence bound (UCB) criterion Silver et al. (2017) when in state s,

$$a_{\text{next}} = \arg\max_a \left(Q(s,a) + c_{\text{puct}} \cdot P(s,a) \cdot \frac{\sqrt{N(s)}}{1+N(s,a)} \right), \tag{1}$$

where $Q(s,a)$ is the empirical expected state-action value, $P(s,a)$ is the prior probability predicted by the network, and $N(s)$ and $N(s,a)$ denote the visit counts of the parent and child nodes, respectively. The coefficient c_{puct} controls exploration within each simulation, where higher values encourage broader search. To further promote exploration, Dirichlet noise (Silver et al., 2017) is injected into the root node's prior probabilities at each episode step. Upon reaching a leaf node, the tree undergoes **expansion**, where new child nodes are added and the neural network evaluates the leaf state value along with prior probabilities for its children. If the leaf is terminal, no expansion occurs and the estimated value is passed directly to the next procedure. During **backpropagation**, this value is propagated upward to the root, updating the mean state-action values and visit counts of all non-terminal nodes along the traversed path. After collecting m simulation results per step, the agent samples the next gate proportionally to the visit counts of the child nodes. To balance exploration and exploitation and facilitate convergence, step-wise decay is applied to the temperature τ when sampling the next action (Silver et al., 2017). Between consecutive steps within an episode, the search tree is reused by re-rooting at the selected action node, improving simulation quality.

Neural network architecture. The policy and value networks directing MCTS are composed of four modules: an embedding layer, a shared network, a policy head, and a value head. Given the current gate sequence and the problem Hamiltonian, the embedding layer maps each input token into a dense vector representation augmented with sinusoidal positional encodings (Vaswani et al., 2017). Hamiltonian terms are encoded via one-hot representations of their Pauli structure, with coefficients normalized to balance the relative contribution of each term and prevent numerical instability from outliers. The Hamiltonian encoding is added to the state embedding with a learnable scaling value, forming the input embedding to the shared network. The shared network is a multi-head Transformer encoder (Vaswani et al., 2017), which captures sequential dependencies among gate tokens in the current prefix. From the final hidden representation, two separate heads produce distinct outputs, where the policy head applies a softmax over the action space to generate prior probabilities for the next gate and the value head computes a scalar estimate of the expected return. To improve value discrimination among states at the same prefix depth, the value head receives the concatenation of the final Transformer output with the embedding of the most recently added gate token as input. To bound memory consumption, the model processes only the last c_{ctx} tokens of the sequence, where c_{ctx} is a hyperparameter. Layer normalization (Ba et al., 2016) and the GELU activation function (Hendrycks & Gimpel, 2016) are employed to stabilize forward-pass signal propagation.

Agent training. The neural network is trained through experience replay to progressively improve simulation quality. At each episode step, a sample consisting of the current gate prefix, the MCTS-derived policy target, and the terminal episode reward is stored in a replay buffer. In addition, a compact best-game buffer (Fawzi et al., 2022) retains high-reward trajectories to facilitate convergence toward promising search regions, as well as to prevent zero or negative signals from dominating the learning process. In each training round, the network is trained for K epochs using the AdamW optimizer (Loshchilov & Hutter, 2017) and a linear learning rate warmup (Vaswani et al., 2017), with each epoch sampling mini-batches from the replay buffer mixed with a fixed proportion of best-game samples once a sufficient number of episodes have been collected. The total training loss $\mathcal{L}$ combines policy and value objectives as: $\mathcal{L} = \mathcal{L}_{\text{p}} + c_v \mathcal{L}_{\text{v}}$, where $\mathcal{L}_{\text{p}}$ is the cross-entropy

loss (Mao et al., 2023) between the current policy output and the MCTS-derived target distribution and $\mathcal{L}_{\text{v}}$ is the Huber loss (Huber, 1992) between the predicted and realized returns, balanced by the coefficient c_v. Throughout training, the policy is periodically evaluated by greedily selecting the highest-probability action at each step with m_{eval} MCTS simulations and reduced Dirichlet noise, tracking whether the learned model recovers the high-reward configurations found during training.

3.3 Curriculum learning for long-horizon and sparse-reward tasks

To address tasks with extremely large maximum episode lengths and sparse-reward signals by progressively guiding the agent through increasingly complex search horizons, we devise a curriculum variant of CRiSP, which we call the incremental horizon curriculum. Inspired by Bengio et al. (2009) and the role of horizon length in offline RL scalability (Park et al., 2025), this strategy consists of three stages with monotonically increasing maximum step counts. This curriculum is made possible by the early-termination property of the target RL environment, which enables the episodic evaluation at any circuit construction step by treating all unassigned prefix positions as identity (I) gates. Specifically, the agent operates with 25% of the full episode length ($0.25T'$) until 25% of total training episodes have elapsed, then expands to 50% ($0.5T'$) until the midpoint of training, and finally opens the complete horizon (T') for the remainder. Although this naive staged schedule may be suboptimal for some cases, empirical results confirm that the strategy is particularly effective on benchmarks with large search spaces and limited simulation budgets, where concentrating early training on shorter, tractable sequences accelerates identification of promising partial configurations.

4 Evaluation

In this section, we evaluate CRiSP on multiple standard VQA benchmarks of practical interest. We first detail the experimental methodology (Sec. 4.1) and present the QAOA benchmark results (Sec. 4.2). Next, we perform downstream VQA optimization to investigate the effect of initialization (Sec. 4.3). Finally, we present exploratory results applying CRiSP to the VQE domain to demonstrate the broader generalizability of our framework (Sec. 4.4). Implementation details are provided in Appendix B, and additional experiments can be found in Appendix C.

4.1 Methodology

QAOA benchmarks. To evaluate CRiSP, we consider three classes of combinatorial optimization problems, with Hamiltonian derivations following Bharadwaj et al. (2026). First, we adopt MaxCut on weighted complete graphs as our primary Quadratic Unconstrained Binary Optimization (QUBO) benchmark, providing highly entangled and globally coupled optimization landscapes. Second, for Polynomial Constrained Binary Optimization (PCBO) evaluation, we utilize the Knapsack problem formulated as a Quadratic Unconstrained Binary Optimization (QUBO) Hamiltonian (Qiskit contributors, 2023). Third, we evaluate a biomedical high-order feature selection task (Shah et al., 2025), comprising the "MedicalDense" and "MedicalSparse" benchmarks. We specifically include the latter because its extreme sparsity induces severe barren plateaus, presenting a critical challenge for VQA optimization. Detailed formulations for these Hamiltonians are deferred to Appendix D.4.

Comparative baseline. To benchmark the quality of the classically-prepared state for these QAOA tasks, we compare CRiSP against SPIQ (Bharadwaj et al., 2026), a state-of-the-art Clifford initialization method leveraging a genetic algorithm (GA). All QAOA experiments are conducted in multi-angle QAOA (ma-QAOA) (Herrman et al., 2022) circuits with $p = 1$, where all parameterized rotations are relaxed, thereby allowing each parameter to have its respective gate prefix.

Metrics. To enable consistent performance comparison across all tasks with substantially different energy scales, we report an accuracy metric similar to that employed in Bharadwaj et al. (2026), defined as Accuracy $= E_{\text{best}}/E_{\text{opt}}$, where E_{best} denotes the minimum energy achieved by a given method and E_{opt} is the ground-state energy of the corresponding Hamiltonian. Across experiments, we employ three random seeds and report the mean and standard deviation of the results.

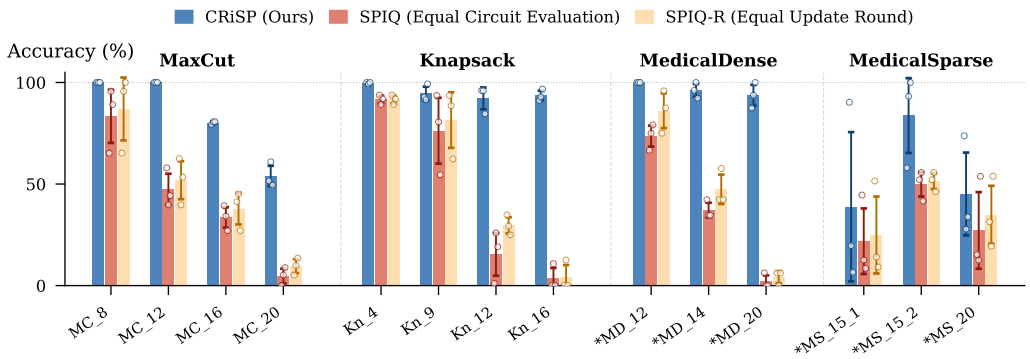

Figure 3: Classical initialization accuracy for QAOA benchmarks. Asterisks (*) indicate tasks using the incremental horizon curriculum. The bar reflects the average accuracy, the error bar represents the standard deviation (std), and the points show results from three independent runs.

4.2 QAOA RESULTS

To assess the CRiSP's performance, we compare the proposed agent against SPIQ (Bharadwaj et al., 2026) on combinatorial optimization problem suites formulated as Hamiltonians. These include MaxCut on weighted complete graphs, the Knapsack problem, and a biomedical high-order interaction task (Shah et al., 2025). Both methods employ QAOA circuit architectures to search for the ground state of each respective Hamiltonian. While CRiSP is broadly applicable to arbitrary Parameterized Quantum Circuits (PQCs), we adopt a ma-QAOA (Herrman et al., 2022) framework to ensure a fair comparison, since SPIQ is specifically tailored to such architectures. Following Bharadwaj et al. (2026), we set the population size of each GA generation to 100. However, we adopt a single-start configuration, as our empirical testing showed that it generally yields superior performance to the original multi-start setting under an equivalent computational budget.

Since both frameworks rely on iterative refinement processes to improve solutions using accumulated performance data, we evaluate SPIQ under two comparative conditions. Our primary baseline matches SPIQ to CRiSP by the total number of circuit evaluations. As a supplementary baseline, we also report the performance of SPIQ-R with the number of update iterations matched to CRiSP (i.e., SPIQ generations are equated to CRiSP's training rounds, where each CRiSP round comprises episode collection from 30 workers followed by K epochs of network updates). Furthermore, for the medical benchmarks (Shah et al., 2025)—featuring either extreme episode lengths up to 1,370 parameters (MedicalDense) or exceptionally sparse reward landscapes (MedicalSparse)—we report CRiSP results utilizing the incremental horizon curriculum detailed in Section 3.3.

Figure 3 summarizes the QAOA benchmark results, demonstrating that CRiSP consistently achieves higher energy accuracy than the competing approaches across all tasks (see Appendix E for numerical details). With an overall mean accuracy of 83.7% across 14 distinct tasks, CRiSP outperforms the evaluation-matched SPIQ baseline by a geometric mean (GeoMean) of $3.19\times$ in average energy accuracy and $2.41\times$ in best-achieved energy accuracy. Maximum improvements peak at $45.02\times$ and $16.01\times$, respectively, on the MedicalDense_20 task. Against SPIQ-R, CRiSP achieves GeoMean improvements of $2.51\times$ (average) and $2.13\times$ (best), with maximum gains of $22.51\times$ (average) and $16.01\times$ (best), again on the MedicalDense_20 benchmark.

A categorical breakdown further underscores the robustness of the proposed framework. Compared to the evaluation-matched baseline, CRiSP achieves GeoMean improvements of $2.89\times$ (average) and $2.25\times$ (best) on MaxCut, $3.81\times$ (average) and $2.48\times$ (best) on Knapsack, and $5.41\times$ (average) and $3.63\times$ (best) on MedicalDense tasks. Notably, even on MedicalSparse tasks—where extreme landscape sparsity makes it inherently difficult for either method to discover non-zero reward configurations—CRiSP still realizes GeoMean advantages of $1.70\times$ (average) and $1.71\times$ (best) higher accuracy than SPIQ across all instances.

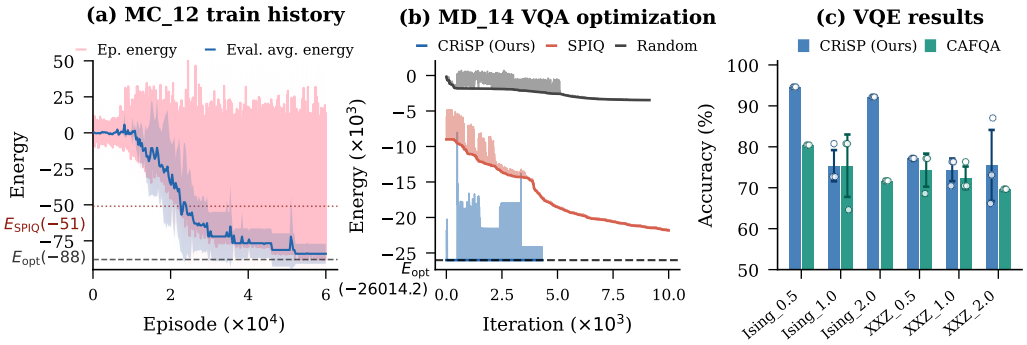

Figure 4: (a) MaxCut_12 training history. E_{opt} and E_{SPIQ} denote the ground-state and the best SPIQ energies, respectively. The shaded areas denote std across three seeds. (b) VQA optimization trajectories of the MedicalDense_14 instance, starting from CRiSP, SPIQ, and random initialization. Solid lines show the best energy found so far for each method, while the faint lines trace the raw per-iteration energy values. (c) Initialization accuracy for VQE tasks. The bar indicates the average accuracy, the error bar displays std, and the points correspond to results from three independent runs.

For the MedicalSparse tasks specifically, we observe a wider accuracy distribution among different random seeds, with certain CRiSP instances occasionally underperforming SPIQ. While consistent outperformance across all seeds is ideal, identifying even a single high-quality solution holds immense practical value in realistic quantum computing use cases, as it drastically reduces subsequent quantum resource expenditures. Consequently, the fact that the best-performing seeds consistently originate from CRiSP is of greater practical utility than strictly outperforming the baseline on average. We hypothesize that performance on these sparse tasks is primarily bottlenecked by the inherent difficulty of initial exploration rather than the search policy itself. This is evidenced by the substantially larger performance margins that CRiSP demonstrates on denser benchmarks, where non-zero reward signals become more readily available to guide learning.

Finally, the performance gap between the two methods notably widens as problem dimensionality scales. While SPIQ degrades significantly on larger instances, CRiSP exhibits strong scalability, maintaining average energy accuracies above 92% throughout all Knapsack and MedicalDense benchmarks. This consistent improvement is further corroborated by the convergence behavior illustrated in Fig. 4(a), which confirms that CRiSP progressively and reliably converges toward low-energy (high-return) configurations given sufficient training budgets.

4.3 Downstream VQA optimization results

To evaluate whether better initialization translates into better VQA outcomes, we run downstream optimization from CRiSP, SPIQ, and random initializations under an identical budget of 10^4 quantum circuit evaluations. Fig. 4(b) shows the optimization trajectories on the MedicalDense_14 instance using the Constrained Optimization BY Linear Approximations (COBYLA) optimizer (Zhang, 2023), tracking the sample with the best final energy for each method.

CRiSP reaches a substantially lower final energy than SPIQ and random initialization. In fact, CRiSP's initial energy is already lower than the final energy of either baseline, so its advantage does not depend on additional quantum evaluations. Because this starting point is already highly accurate, the further gain from optimization is limited; it is nonzero but less visible at the scale of the figure. The same trend holds across all benchmark categories (Appendix C.1): CRiSP reaches a lower final energy, and in cases where the baselines come close, they require many more quantum evaluations to do so. These results indicate that a better initialization leads to better downstream performance by starting the optimization in a favorable region of the landscape. This is consistent with prior work showing that VQA landscapes contain many poor local minima (Anschuetz & Kiani, 2022) and that good initialization substantially improves QAOA performance (Zhou et al., 2020). How much of this benefit is realized also depends on the choice of continuous optimizer; to our knowledge, this remains an open question, which we view as orthogonal to finding a good initialization.

4.4 FRAMEWORK GENERALIZATION TO VQE

To investigate the applicability of CRiSP beyond the QAOA domain, we preliminarily evaluate the framework on VQE benchmarks using two physical models: the one-dimensional Transverse-Field Ising Model (TFIM) (Pfeuty, 1970) and the XXZ model (Bonechi et al., 1992), a specific variant of the Heisenberg model. Detailed Hamiltonian formulations for these models are provided in Section D.5. Both models are evaluated at $N = 10$ across three coupling strengths, $J \in \{0.5, 1.0, 2.0\}$.

We compare CRiSP against a scalable variant of CAFQA (Ravi et al., 2023) (implemented via a GA, as employed in Clapton (Seifert et al., 2024)), a Clifford initialization method designed for VQE, under equal circuit evaluation budgets. Both approaches utilize a circular hardware-efficient ansatz (Kandala et al., 2017) with one repetition, resulting in 40 parameters per circuit. Given the exploratory nature of this experiment, we adopt the closest QAOA benchmark configuration of comparable problem size without VQE-tailored algorithmic setup, performing a grid search over the per-round training epochs $K \in \{12, 16, 20, 24\}$ and the value coefficient $c_v \in \{1, 2\}$.

As presented in Fig. 4(c), CRiSP achieves comparable or higher accuracy than CAFQA across all benchmarks, with GeoMeans of $1.1\times$ in average accuracy and $1.11\times$ in best-achieved accuracy (see Appendix E for details). On Ising_0.5 and Ising_2.0, CRiSP consistently outperforms CAFQA by $1.18\times$ and $1.29\times$, respectively. On the XXZ models, although the variance across seeds increases with the coupling strength J, CRiSP successfully identifies the best circuit configurations across all benchmarks, some of which remain unreachable by CAFQA. While additional hyperparameter tuning would likely further enhance performance, these preliminary results indicate that CRiSP generalizes to VQE settings without requiring task-specific modifications, supporting its potential applicability to a broader class of VQA problems.

5 CONCLUSIONS, LIMITATIONS, AND FUTURE OUTLOOK

We introduced CRiSP, a reinforcement learning framework for classically simulable state preparation that addresses the resource cost of VQAs on early fault-tolerant hardware. By casting Clifford initialization as a sequential decision-making problem, CRiSP uses Transformer-guided Monte Carlo Tree Search to navigate the exponentially large Clifford search space. Across QAOA benchmarks with up to 22 qubits and $1{,}370$ parameters, as well as VQE, CRiSP consistently outperforms the state-of-the-art heuristic baseline, improving average energy accuracy by a mean of $3.19\times$ under the same circuit evaluation budget.

Our work has some limitations. First, although the empirical results in this paper and in prior work (Bharadwaj et al., 2026) clearly show that lower initial energies lead to better results after continuous optimization, there is no formal guarantee that this holds in general. Second, as with many near-term algorithms, VQAs are not guaranteed to achieve practical advantage over advanced classical methods, though their noise resilience and flexibility justify continued study. Third, while each stabilizer simulation remains polynomial and our curriculum handles large parameter counts, the Clifford search space still grows exponentially, so training cost increases with system size.

We highlight some directions for future work:

Landscape-aware initialization: Reward signals could incorporate landscape properties such as gradient or Hessian magnitudes, guiding the agent toward low-energy states in trainable regions and accelerating subsequent optimization.

Near-Clifford state preparation: Allowing a small budget of non-Clifford gates (e.g., T gates) expands the reachable state space, at a simulation cost that grows exponentially only in the number of non-Clifford gates. Extending CRiSP to this action space could yield higher-quality initial states.

Zero-shot or few-shot transfer: A model trained on a set of problem instances that generalizes to unseen ones would reduce per-instance training cost.

Generalization beyond VQAs: Replacing the energy objective with state or unitary fidelity would adapt our sequential construction framework to approximate state preparation and circuit compilation, for example to reduce T-count or multi-qubit gate count.

AI USE STATEMENT

In this work, we have not used generative AI tools for helping to develop theoretical models or conceptual frameworks, designing or providing feedback on research methodology or experiments, implementing methods, assisting with translation, cleaning or reformatting datasets, supporting qualitative and thematic data analysis, and interpreting results. Generating synthetic data sets, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, and proposing or refining hypotheses are not applicable to this work. Additionally, we used generative AI tools for sourcing/searching for information, editing a research paper to improve readability, and editing software code (basic code assistance). We have reviewed all AI-assisted work. For example, LLM-edited code was verified and tested for correctness, and the authors have checked the AI-edited text and verified information obtained from AI-assisted searches. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

The research pursued in this paper is foundational and focuses on using ML to improve the execution of specific quantum algorithms on quantum systems. While one could always use these quantum algorithms to construct malicious applications, this is not a direct jump from our research, so we do not identify any direct concerns with respect to the Code of Ethics.

REPRODUCIBILITY STATEMENT

We present full implementation details and hyperparameters in the main paper and appendix. For instance, see Sec. 4.1, Appendix B, Appendix D.4, and Appendix D.5. We also provide code as supplemental material.

REFERENCES

Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Physical Review A—Atomic, Molecular, and Optical Physics*, 70(5):052328, 2004.

Eric R Anschuetz and Bobak T Kiani. Quantum variational algorithms are swamped with traps. *Nature Communications*, 13(1):7760, 2022.

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

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

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

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009.

Dhanvi Bharadwaj, Yuewen Hou, Guang-Yi Li, and Gokul Subramanian Ravi. Scalable clifford-based classical initialization for the quantum approximate optimization algorithm. *arXiv preprint arXiv:2602.14327*, 2026. To appear in 2026 IEEE International Conference on Quantum Computing and Engineering (QCE).

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

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

F Bonechi, E Celeghini, R Giachetti, E Sorace, and M Tarlini. Heisenberg xxz model and quantum galilei group. *Journal of Physics A: Mathematical and General*, 25(15):L939, aug 1992. doi: 10.1088/0305-4470/25/15/007. URL https://dx.doi.org/10.1088/0305-4470/25/15/007.

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

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

Sergey Bravyi and Alexei Kitaev. Universal quantum computation with ideal clifford gates and noisy ancillas. *Physical Review A—Atomic, Molecular, and Optical Physics*, 71(2):022316, 2005.

Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43, 2012.

Marco Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, et al. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021.

MH Cheng, KE Khosla, CN Self, M Lin, BX Li, AC Medina, and MS Kim. Clifford circuit initialization for variational quantum algorithms. *Physical Review A*, 111(6):062413, 2025.

Chi-Ning Chou, Peter J. Love, Juspreet Singh Sandhu, and Jonathan Shi. Limitations of Local Quantum Algorithms on Random Max-k-XOR and Beyond, February 2022.

Barry A. Cipra. An introduction to the ising model. *The American Mathematical Monthly*, 94 (10):937–959, 1987. doi: 10.1080/00029890.1987.12000742. URL https://doi.org/10.1080/00029890.1987.12000742.

Siddharth Dangwal, Gokul Subramanian Ravi, Poulami Das, Kaitlin N. Smith, Jonathan M. Baker, and Frederic T. Chong. Varsaw: Application-tailored measurement error mitigation for variational quantum algorithms, 2023.

Maxime Dupont, Nicolas Didier, Mark J Hodson, Joel E Moore, and Matthew J Reagor. Entanglement perspective on the quantum approximate optimization algorithm. *Physical Review A*, 106 (2):022423, 2022.

E. Farhi, J. Goldstone, and S. Gutmann. A Quantum Approximate Optimization Algorithm Applied to a Bounded Occurrence Constraint Problem. *arXiv*, June 2015. doi: 10.48550/arXiv.1412.6062.

Bryan Eastin and Emanuel Knill. Restrictions on transversal encoded quantum gate sets. *Physical review letters*, 102(11):110502, 2009.

Jeffrey M Epstein, Andrew W Cross, Easwar Magesan, and Jay M Gambetta. Investigating the limits of randomized benchmarking protocols. *Physical Review A*, 89(6):062321, 2014.

Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Leo Zhou. The Quantum Approximate Optimization Algorithm and the Sherrington-Kirkpatrick Model at Infinite Size. *Quantum*, 6:759, July 2022. ISSN 2521-327X. doi: 10.22331/q-2022-07-07-759.

Alhussein Fawzi, Matej Balog, Aja Huang, Thomas Hubert, Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Francisco J R. Ruiz, Julian Schrittwieser, Grzegorz Swirszcz, et al. Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature*, 610(7930):47–53, 2022.

Fuchang Gao and Lixing Han. Implementing the nelder-mead simplex algorithm with adaptive parameters. *Comput. Optim. Appl.*, 51(1):259–277, January 2012. ISSN 0926-6003. doi: 10.1007/s10589-010-9329-3. URL https://doi.org/10.1007/s10589-010-9329-3.

Craig Gidney. Stim: a fast stabilizer circuit simulator. *Quantum*, 5:497, July 2021. ISSN 2521-327X. doi: 10.22331/q-2021-07-06-497. URL https://doi.org/10.22331/q-2021-07-06-497.

Craig Gidney, Noah Shutty, and Cody Jones. Magic state cultivation: growing t states as cheap as cnot gates. *arXiv preprint arXiv:2409.17595*, 2024.

Tudor Giurgica-Tiron, Yousef Hindy, Ryan LaRose, Andrea Mari, and William J Zeng. Digital zero noise extrapolation for quantum error mitigation. In *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 306–316. IEEE, 2020.

Fred Glover, Gary Kochenberger, and Yu Du. A tutorial on formulating and using qubo models. *arXiv preprint arXiv:1811.11538*, 2019. URL https://arxiv.org/abs/1811.11538.

Matthew L Goh, Martin Larocca, Lukasz Cincio, Marco Cerezo, and Frédéric Sauvage. Lie-algebraic classical simulations for quantum computing. *Physical Review Research*, 7(3):033266, 2025.

Google. Google quantum computing roadmap. https://quantumai.google/roadmap, 2024. Accessed: 2024-11-22.

Daniel Gottesman. The heisenberg representation of quantum computers. *arXiv preprint quant-ph/9807006*, 1998.

Edward Grant, Leonard Wossnig, Mateusz Ostaszewski, and Marcello Benedetti. An initialization strategy for addressing barren plateaus in parametrized quantum circuits. *Quantum*, 3:214, 2019.

Harper R. Grimsley, Sophia E. Economou, Edwin Barnes, and Nicholas J. Mayhall. An adaptive variational algorithm for exact molecular simulations on a quantum computer. *Nature Communications*, 10(1), Jul 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-10988-2. URL http://dx.doi.org/10.1038/s41467-019-10988-2.

Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pp. 212–219, 1996.

Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2024. URL https://www.gurobi.com.

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

Tianyi Hao, Amanda Xu, and Swamit Tannu. Reducing t gates with unitary synthesis. *arXiv preprint arXiv:2503.15843*, 2025.

Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. Array programming with numpy. *nature*, 585(7825):357–362, 2020.

M. B. Hastings. Classical and Quantum Bounded Depth Approximation Algorithms. *arXiv*, August 2019. doi: 10.48550/arXiv.1905.07047.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Rebekah Herrman, Phillip C Lotshaw, James Ostrowski, Travis S Humble, and George Siopsis. Multi-angle quantum approximate optimization algorithm. *Scientific Reports*, 12(1):6781, 2022.

Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*, pp. 492–518. Springer, 1992.

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

IBM. Ibm quantum computing roadmap. https://www.ibm.com/roadmaps/quantum, 2024. Accessed: 2024-11-22.

J. Preskill. Beyond nisq: The megaquop machine. *ACM Transactions on Quantum Computing*, 6 (3):1–7, April 2025. ISSN 2643-6817. doi: 10.1145/3723153. URL http://dx.doi.org/10.1145/3723153.

Yuwei Jin, Zirui Li, Fei Hua, Tianyi Hao, Huiyang Zhou, Yipeng Huang, and Eddy Z Zhang. Tetris: A compilation framework for vqa applications in quantum computing. In *2024 ACM/IEEE 51st Annual International Symposium on Computer Architecture (ISCA)*, pp. 277–292. IEEE, 2024.

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

Amara Katabarwa, Katerina Gratsea, Athena Caesura, and Peter D. Johnson. Early fault-tolerant quantum computing. *PRX Quantum*, 5:020101, Jun 2024. doi: 10.1103/PRXQuantum.5.020101. URL https://link.aps.org/doi/10.1103/PRXQuantum.5.020101.

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. Evidence for the utility of quantum computing before fault tolerance. *Nature*, 618(7965):500–505, 2023.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL https://arxiv.org/abs/1412.6980.

Vadym Kliuchnikov, Dmitri Maslov, and Michele Mosca. Fast and efficient exact synthesis of single qubit unitaries generated by clifford and t gates. *arXiv preprint arXiv:1206.5236*, 2012.

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

David Kremer, Ali Javadi-Abhari, and Priyanka Mukhopadhyay. Optimizing the non-clifford-count in unitary synthesis using reinforcement learning. *arXiv preprint arXiv:2509.21709*, 2025.

Wim Lavrijsen, Ana Tudor, Juliane Müller, Costin Iancu, and Wibe de Jong. Classical optimizers for noisy intermediate-scale quantum devices. In *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 267–277, 2020. doi: 10.1109/QCE49297.2020.00041.

Xinwei Lee, Ningyi Xie, Dongsheng Cai, Yoshiyuki Saito, and Nobuyoshi Asai. A Depth-Progressive Initialization Strategy for Quantum Approximate Optimization Algorithm. *Mathematics*, 11(9):2176, 2023. doi: 10.3390/math11092176.

Xinwei Lee, Xinjian Yan, Ningyi Xie, Dongsheng Cai, Yoshiyuki Saito, and Nobuyoshi Asai. Iterative layerwise training for the quantum approximate optimization algorithm. *Physical Review A*, 109(5):052406, 2024.

Gushu Li, Anbang Wu, Yunong Shi, Ali Javadi-Abhari, Yufei Ding, and Yuan Xie. Paulihedral: a generalized block-wise compiler optimization framework for quantum simulation kernels. In *Proceedings of the 27th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 554–569, 2022.

Cedric Yen-Yu Lin and Yechao Zhu. Performance of QAOA on Typical Instances of Constraint Satisfaction Problems with Bounded Degree. *arXiv*, January 2016. doi: 10.48550/arXiv.1601.01744.

Daniel Litinski. Magic state distillation: Not as costly as you think. *Quantum*, 3:205, 2019.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Anqi Mao, Mehryar Mohri, and Yutao Zhong. Cross-entropy loss functions: Theoretical analysis and applications. In *International conference on Machine learning*, pp. 23803–23828. pmlr, 2023.

Kunal Marwaha. Local classical MAX-CUT algorithm outperforms $p=2$ QAOA on high-girth regular graphs. *Quantum*, 5, April 2021. ISSN 2521-327X. doi: 10.22331/q-2021-04-20-437.

Nina Mazyavkina, Sergey Sviridov, Sergei Ivanov, and Evgeny Burnaev. Reinforcement learning for combinatorial optimization: A survey. *Computers & Operations Research*, 134:105400, 2021.

Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics*, 18(2):023023, 2016.

Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nat. Commun.*, 9(1):4812, 2018.

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. Quantum optimization using variational algorithms on near-term quantum devices. *Quantum Science and Technology*, 3, 06 2018. doi: 10.1088/2058-9565/aab822.

Kouhei Nakaji, Lasse Bjørn Kristensen, Ryota Kemmoku, Jorge A Campos-Gonzalez-Angulo, Mohammad Ghazi Vakili, Haozhe Huang, Mohsen Bagherimehrab, Christoph Gorgulla, FuTe Wong, Alex McCaskey, et al. The generative quantum eigensolver (gqe) and its application for ground state search. *arXiv preprint arXiv:2401.09253*, 2024.

Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

Julius Beneoluchi Odili. Combinatorial optimization in science and engineering. *Current Science*, pp. 2268–2274, 2017.

Joe O'Gorman and Earl T. Campbell. Quantum computation with realistic magic-state factories. *Physical Review A*, 95(3), Mar 2017. ISSN 2469-9934. doi: 10.1103/physreva.95.032338. URL http://dx.doi.org/10.1103/PhysRevA.95.032338.

Seohong Park, Kevin Frans, Deepinder Mann, Benjamin Eysenbach, Aviral Kumar, and Sergey Levine. Horizon reduction makes rl scalable. *arXiv preprint arXiv:2506.04168*, 2025.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

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

Pierre Pfeuty. The one-dimensional ising model with a transverse field. *ANNALS of Physics*, 57(1): 79–90, 1970.

John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, 2018.

Qiskit contributors. Qiskit: An open-source framework for quantum computing, 2023.

Ronald L Rardin. *Optimization in operations research*, volume 166. Prentice Hall Upper Saddle River, NJ, 1998.

Gokul Subramanian Ravi, Kaitlin N. Smith, Pranav Gokhale, Andrea Mari, Nathan Earnest, Ali Javadi-Abhari, and Frederic T. Chong. Vaqem: A variational approach to quantum error mitigation, 2021.

Gokul Subramanian Ravi, Kaitlin N. Smith, Jonathan M. Baker, Tejas Kannan, Nathan Earnest, Ali Javadi-Abhari, Henry Hoffmann, and Frederic T. Chong. Navigating the dynamic noise landscape of variational quantum algorithms with qismet, 2022. URL https://arxiv.org/abs/2209.12280.

Gokul Subramanian Ravi, Pranav Gokhale, Yi Ding, William Kirby, Kaitlin Smith, Jonathan M Baker, Peter J Love, Henry Hoffmann, Kenneth R Brown, and Frederic T Chong. Cafqa: A classical simulation bootstrap for variational quantum algorithms. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 1*, pp. 15–29, 2023.

A. H. G. Rinnooy Kan and G. T. Timmer. Stochastic global optimization methods part i: Clustering methods. *Mathematical Programming*, 39:27–56, 1987.

Neil J Ross and Peter Selinger. Optimal ancilla-free clifford+ t approximation of z-rotations. *Quantum Inf. Comput.*, 16(11&12):901–953, 2016.

Manuel S Rudolph, Jacob Miller, Danial Motlagh, Jing Chen, Atithi Acharya, and Alejandro Perdomo-Ortiz. Synergistic pretraining of parametrized quantum circuits via tensor networks. *Nature Communications*, 14(1):8367, 2023.

Francisco JR Ruiz, Tuomas Laakkonen, Johannes Bausch, Matej Balog, Mohammadamin Barekatain, Francisco JH Heras, Alexander Novikov, Nathan Fitzpatrick, Bernardino Romera-Paredes, John Van De Wetering, et al. Quantum circuit optimization with alphatensor. *Nature Machine Intelligence*, 7(3):374–385, 2025.

Stefan H. Sack and Maksym Serbyn. Quantum annealing initialization of the quantum approximate optimization algorithm. *Quantum*, 5:491, July 2021. ISSN 2521-327X. doi: 10.22331/q-2021-07-01-491. URL http://dx.doi.org/10.22331/q-2021-07-01-491.

Abdelkader Sbihi and Richard W Eglese. Combinatorial optimization and green logistics. *Annals of Operations Research*, 175(1):159–175, 2010.

Lennart Maximilian Seifert, Siddharth Dangwal, Frederic T Chong, and Gokul Subramanian Ravi. Clapton: Clifford assisted problem transformation for error mitigation in variational quantum algorithms. In *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 4*, pp. 47–62, 2024.

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. Toward quantum-enabled biomarker discovery: An outlook from q4bio, 2025. URL https://arxiv.org/abs/2509.25904.

Peter W Shor. Fault-tolerant quantum computation. In *Proceedings of 37th conference on foundations of computer science*, pp. 56–65. IEEE, 1996.

Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, Oct 1997. ISSN 1095-7111. doi: 10.1137/s0097539795293172. URL http://dx.doi.org/10.1137/S0097539795293172.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

Volodymyr Sivak, Alexis Morvan, Michael Broughton, Matthew Neeley, Alec Eickbusch, Dmitry Abanin, Amira Abbas, Rajeev Acharya, Laleh Aghababaie Beni, Georg Aigeldinger, et al. Reinforcement learning control of quantum error correction. *arXiv preprint arXiv:2511.08493*, 2025.

James C Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE transactions on automatic control*, 37(3):332–341, 2002.

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

Richard S Sutton, Andrew G Barto, et al. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H. Booth, and Jonathan Tennyson. The variational quantum eigensolver: A review of methods and best practices. *Physics Reports*, 986:1–128, 2022. ISSN 0370-1573. doi: https://doi.org/10.1016/j.physrep.2022.08.003. URL https://www.sciencedirect.com/science/article/pii/S0370157322003118. The Variational Quantum Eigensolver: a review of methods and best practices.

Matthew Treinish, Ivan Carvalho, Georgios Tsilimigkounakis, and Nahum Sá. rustworkx: A high-performance graph library for python. *arXiv preprint arXiv:2110.15221*, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Hanrui Wang, Yongshan Ding, Jiaqi Gu, Zirui Li, Yujun Lin, David Z. Pan, Frederic T. Chong, and Song Han. Quantumnas: Noise-adaptive search for robust quantum circuits, 2022.

Samson Wang, Piotr Czarnik, Andrew Arrasmith, Marco Cerezo, Lukasz Cincio, and Patrick J Coles. Can error mitigation improve trainability of noisy variational quantum algorithms? *arXiv preprint arXiv:2109.01051*, 2021.

Zhihui Wang, Stuart Hadfield, Zhang Jiang, and Eleanor G. Rieffel. Quantum Approximate Optimization Algorithm for MaxCut: A Fermionic View. *Physical Review A*, 97(2):022304, February 2018. ISSN 2469-9926, 2469-9934. doi: 10.1103/PhysRevA.97.022304.

Mathias Weiden, Justin Kalloor, John Kubiatowicz, and Costin Iancu. Making neural networks more suitable for approximate clifford+ t circuit synthesis. In *2025 IEEE International Conference on Quantum Computing and Engineering (QCE)*, volume 1, pp. 501–510. IEEE, 2025.

Jiahao Yao, Haoya Li, Marin Bukov, Lin Lin, and Lexing Ying. Monte carlo tree search based hybrid optimization of variational quantum circuits. In *Mathematical and Scientific Machine Learning*, pp. 49–64. PMLR, 2022.

Zaikun Zhang. PRIMA: Reference Implementation for Powell's Methods with Modernization and Amelioration. available at http://www.libprima.net, DOI: 10.5281/zenodo.8052654, 2023.

Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Phys. Rev. X*, 10:021067, Jun 2020. doi: 10.1103/PhysRevX.10.021067. URL https://link.aps.org/doi/10.1103/PhysRevX.10.021067.

Ciyou Zhu, Richard H. Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Trans. Math. Softw.*, 23(4): 550–560, December 1997. ISSN 0098-3500. doi: 10.1145/279232.279236. URL https://doi.org/10.1145/279232.279236.

APPENDICES

A ADDITIONAL RELATED WORK

Beyond Clifford-based initialization, several other strategies aim to give VQAs a better starting point, each targeting a different regime. Tensor-network-based pretraining (Rudolph et al., 2023) represents the circuit state with a truncated bond dimension. This allows arbitrary non-Clifford states but caps how much entanglement the state can carry, which is the opposite trade-off from Clifford-based methods: those are restricted to stabilizer states but place no cap on entanglement. The entanglement cap matters for QAOA in particular. Even when the optimal solution is a product state, the QAOA circuit can pass through a volume-law entanglement barrier in the middle of its evolution (Dupont et al., 2022). An entanglement-limited representation cannot faithfully capture these intermediate states, which can limit the quality of the initial state it produces.

Lie-algebraic simulation and initialization techniques (Goh et al., 2025) take a different route. When the dynamical Lie algebra generated by the ansatz is polynomially sized and contains the observable, expectation values can be computed efficiently on a classical computer. In this regime, however, the entire variational problem becomes classically tractable, so these methods effectively replace the quantum algorithm rather than warm-start it. Outside this regime, they do not apply.

Adaptive ansatz approaches such as ADAPT-VQE (Grimsley et al., 2019) and generative circuit synthesis models such as GQE (Nakaji et al., 2024) instead search over the circuit structure itself rather than warm-starting a fixed ansatz. ADAPT-VQE grows the ansatz one operator at a time based on measured energy gradients, while GQE trains a generative model to propose circuits scored by their energies. Both typically rely on quantum hardware evaluations during the search, which incurs the same per-evaluation cost that motivates classical initialization in the first place. In contrast, CRiSP warm-starts a fixed ansatz and performs its entire search classically via stabilizer simulation (Gidney, 2021), so no quantum evaluations are spent before the VQA begins.

B EXPERIMENTAL DETAILS

MCTS trajectory collection is distributed across 30 parallel workers, forming 30 episodes per training round. The training mini-batch size is set to 512 or 1024 depending on the problem size. We employ a two-stage simulation schedule, in which the agent begins with $m = 50$ simulations per step during the warmup period, which is the first 10% of the total episodes, to allow rapid exploration, then transitions to a standard phase with $m = 100$ simulations to enhance search quality. The evaluation simulation count m_{eval} is set equal to that of the standard phase. The model is trained using AdamW (Loshchilov & Hutter, 2017) with a peak learning rate of 10^{-4} and a linear learning rate warmup (Vaswani et al., 2017) during the initial stage with a reduced simulation count. To facilitate convergence, a step-wise exponential temperature decay (Silver et al., 2017) is applied to τ after the warmup episodes, starting from $\tau = 1.0$ at step 25 until a horizon-dependent threshold is met, after which τ is held at 0.75. When the incremental horizon curriculum is applied, we use reduced c_{puct} to enhance convergence on large tasks, while a grid search over $c_{\text{puct}} \in \{1, 1.5\}$ is performed in sparse-reward environments. To encourage exploration upon each horizon expansion, the temperature is temporarily set to $\tau = 1.2$ for the subsequent 5% of episodes, applied only to gate selection steps within the newly expanded portion of the search horizon. Depending on the problem size, hyperparameters including simulation counts, training episodes, epochs, and learning rates are adjusted based on resource constraints.

The implementation uses Qiskit 2.3.0 (Qiskit contributors, 2023) for general quantum circuit operations, Pytorch 2.5.1 (Paszke et al., 2019) for model training, Stim 1.15.0 (Gidney, 2021) for classical stabilizer tableau simulation to compute the expected energy of the synthesized quantum circuits, NumPy 2.3.3 (Harris et al., 2020) for numerical computation, and rustworkx 0.17.1 (Treinish et al., 2021) for graph operations, all running on Python 3.12. All experiments are conducted on a single NVIDIA A100 GPU with 32 cores of a 64-core AMD EPYC 7763 CPU and up to 64GB memory. Execution time ranges from 10 hours to 80 hours, depending on the simulation scale and budget. Hyperparameters are reported in Table 1. For VQE experiments, the values of K and c_v are listed in Table 2, with all remaining hyperparameters set to MaxCut_8 configuration in Table 1.

Table 1: Hyperparameters for CRiSP training and architecture. When indicating benchmarks, MC, Kn, MD, and MS refer to MaxCut, Knapsack, MedicalDense, and MedicalSparse, respectively.

Category	Hyperparameter	Value
Architecture	Transformer layers / heads	2 / 4
	Transformer input / FF dimension	128 / 256
	State / Positional embedding dimension	32 / 96
	Positional embedding / size	Sinusoidal (Vaswani et al., 2017)
	Hamiltonian embedding size	[256, 128]
	Policy / Value head size	[128, 128, 24] / [256, 128, 64, 1]
	Context length (c_{ctx})	64
	Activation	GELU (Hendrycks & Gimpel, 2016)
	Normalization	LayerNorm (Ba et al., 2016)
MCTS	c_{puct}	1.0 (default), 0.8 (MD_12/14/20), 1.5 (MS_15_1)
	Dirichlet noise (α_D, ϵ_D)	0.15, 0.2 (Eval: 0.15, 0.1)
	Warmup / Standard simulations (m)	50 / 100 (default), 30 / 50 (MC_20, Kn_16, MD_12/14/20)
	Parallel workers	30
	Initial / Decayed temperature τ	1.0 / 0.75
Optimization	Total episodes	6×10^4 (default), 3×10^4 (MC_20, Kn_12/16, MD_12), 1.5×10^4 (MD_14), 6×10^3 (MD_20)
	Training epochs (K)	16 (default), 12 (MC_8), 24 (MC_20, Kn_9/12/16), 32 (MD_12/14), 40 (MD_20)
	Optimizer	AdamW (Loshchilov & Hutter, 2017)
	Peak learning rate	10^{-4} (default), 3×10^{-4} (MD_20)
	Weight decay	10^{-6}
	Value coefficient (c_v)	2.0
	Mini-batch size	1024 (default), 512 (MC_8/12, Kn_4, MS_15_1/15_2)
	Gradient clipping	1.0
	Replay buffer size	5×10^5 (default), 10^5 (MC_20, Kn_9/12/16, MD_12, MS_20), 1.5×10^5 (MD_14), 4×10^5 (MD_20)
	Best game buffer size	10^3 (default), 2×10^3 (Kn_9/12), 3×10^3 (MC_20, Kn_16, MD_12), 5×10^3 (MD_14), 10^4 (MD_20)
	Training sample ratio (replay, best game)	(0, 0) (initial), (1.0, 0) (after 300 episodes), (0.9, 0.1) (after 600 episodes)

Table 2: Hyperparameters for VQE training across different tasks.

Task	Training epochs (K)	Value coefficient (c_v)
Ising_0.5	12	1.0
Ising_1.0	16	2.0
Ising_2.0	24	2.0
XXZ_0.5	16	1.0
XXZ_1.0	12	1.0
XXZ_2.0	16	1.0

C Additional experiment results

C.1 Additional downstream VQA optimization results

Figure 5 presents additional downstream optimization results for MaxCut_16, Knapsack_9, MedicalDense_12, and MedicalSparse_15_2, starting from CRiSP, SPIQ (Bharadwaj et al., 2026), and random initializations. All methods receive the same quantum circuit evaluation budget for a given instance: $4{,}000$ for MaxCut_16 and 10^4 for the others. Across all four instances, CRiSP achieves a lower final energy than both baselines. In MedicalDense_12, SPIQ approaches CRiSP's final energy, but it requires substantially more quantum evaluations to do so.

For Knapsack_9 and MedicalDense_12, CRiSP's initialization is already highly accurate, so the subsequent optimization gain (up to an order of magnitude lower error) is not clearly visible at the scale of the figure. For MedicalSparse_15_2, optimization does not improve the CRiSP result, and the same holds for SPIQ. This may be due to a flat landscape around the initial point, such as a barren plateau, and highlights room for further research on post-initialization optimization. Even so, CRiSP's energy remains substantially lower than SPIQ's.

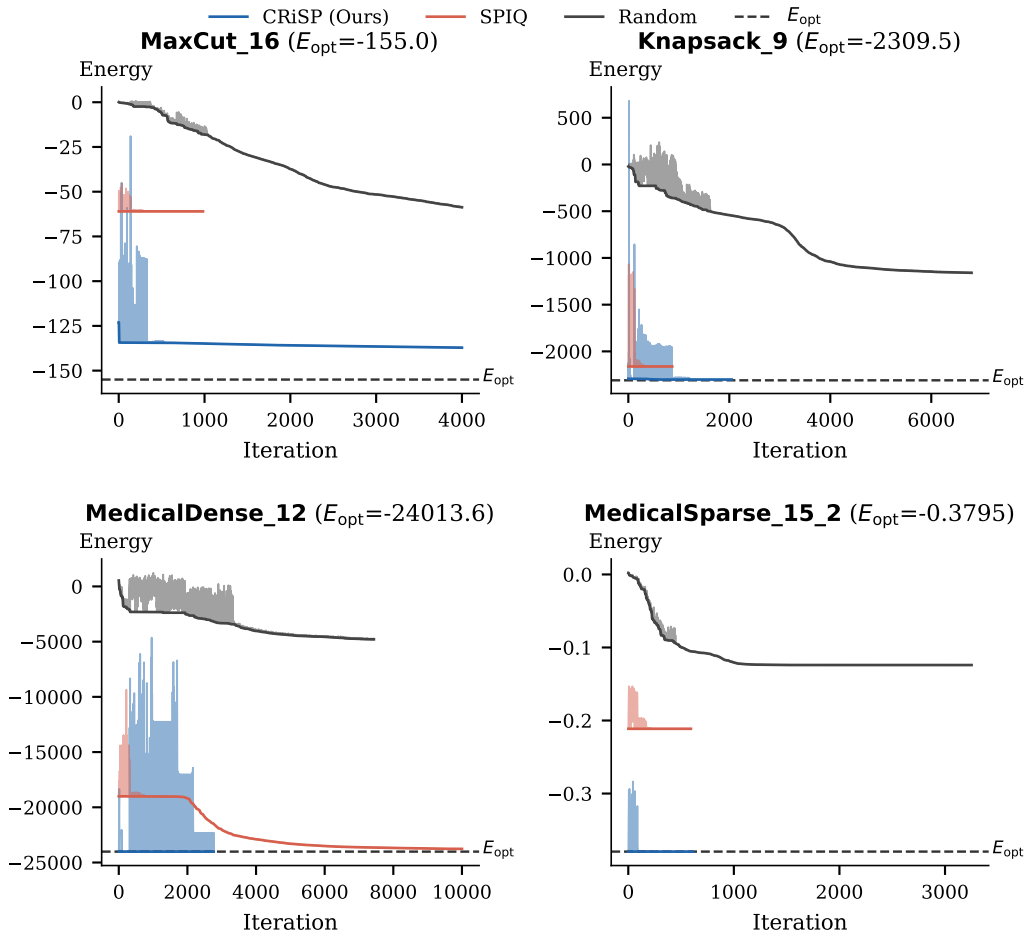

Figure 5: Downstream VQA optimization results, comparing random initialization, SPIQ, and CRiSP as starting points for continuous COBYLA optimization (Zhang, 2023). Circuit evaluation budgets are matched within an instance, set on a similar scale to the budgets used in Bharadwaj et al. (2026). Solid lines show the best energy found so far for each method, while the faint lines trace the raw per-iteration energy values. Lower energy with fewer iterations is better.

Table 3: Time-matched comparison results on MaxCut_12, MaxCut_20, Knapsack_16, and MedicalDense_20. SPIQ-T denotes a variant of SPIQ (Bharadwaj et al., 2026) whose wall-time is matched to the average time CRiSP took to reach its best solution. E_{opt} indicates the optimal energy. Accuracy results are reported as mean $\pm$ std across three independent runs, where higher is better. Bold values denote the best average performance for each task.

Task	E_{opt}	CRiSP (Ours) (%)	SPIQ-T (%)
MaxCut_12	-88	**100** $\pm$ 0	52.27 $\pm$ 11.97
MaxCut_20	-192	**53.99** $\pm$ 6.1	40.63 $\pm$ 6.53
Knapsack_16	-19538.5	**93.61** $\pm$ 2.85	24.52 $\pm$ 21.22
MedicalDense_20	-32023.8	**93.71** $\pm$ 6.25	23.94 $\pm$ 3.6

C.2 Additional QAOA results under time-matched budgets

Table 3 reports the time-matched QAOA results on four representative benchmarks (MaxCut_12, MaxCut_20, Knapsack_16, MedicalDense_20). We evaluate SPIQ-T, a time-matched variant of SPIQ (Bharadwaj et al., 2026), under a wall-time budget equal to the average time CRiSP took to

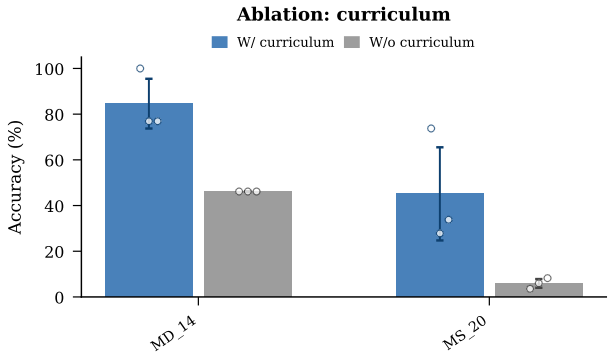

Figure 6: Ablation study with and without incremental horizon curriculum, showing CRiSP accuracy. The bar indicates the average accuracy, the error bar displays std, and the points correspond to results from three independent runs.

reach its best solution. Under this equal-time comparison, SPIQ-T reaches 52.27%, 40.63%, 24.52% and 23.94% average accuracy on MaxCut_12, MaxCut_20, Knapsack_16 and MedicalDense_20, respectively, while CRiSP achieves 100%, 53.99%, 93.61% and 93.71% average accuracy. This indicates that CRiSP's advantage comes from a more effective search within the Clifford space, not merely from spending more classical compute.

C.3 Ablation study on incremental horizon curriculum

To examine the contribution of the incremental horizon curriculum, we test CRiSP with and without the curriculum on two representative benchmarks where the total episode length is large and the energy landscape is highly barren, namely MedicalDense_14 and MedicalSparse_20. Due to the substantial computational cost of full-horizon training on MedicalDense tasks, the ablation is conducted under a reduced setting with $3{,}000$ training episodes, a simulation count of 30 for both the warmup and standard phases, and a learning rate of 3×10^{-4}, while keeping all other hyperparameters the same as the configuration for the main results. Both variants are evaluated under this same setting to ensure a fair comparison. The ablation for MedicalSparse_20 is performed using the identical experimental setup as the main experiments.

As shown in Fig. 6, without the curriculum, CRiSP struggles to effectively explore the full search space within the allocated training budget, causing substantial performance degradation on both benchmarks. On MedicalDense_14, the curriculum pushes the average accuracy from 46.14% to 84.59%. This indicates that the gradual expansion of the search horizon enables the agent to uncover partially promising configurations in the early stages of training, and to further focus its exploration on those regions as the horizon extends toward its full length. The performance gap becomes even more drastic on MedicalSparse_20, where the reward landscape is extremely sparse. Here, the mean accuracy improves $7.62\times$, from 5.92% to 45.11%, because the curriculum allows the agent to find positive reward signals during early episodes within confined search spaces—regions that would otherwise remain unexplored during full-horizon exploration. These results confirm that the incremental horizon curriculum is highly beneficial in addressing tasks with large parameter counts and challenging energy landscapes.

C.4 Ablation study for CRiSP's components

To examine which components drive CRiSP's gains, we conduct an ablation on MaxCut_12 under a matched stabilizer simulation budget. We evaluate variants that remove CRiSP's best-game buffer, temperature decay, or Dirichlet noise, as well as variants that replace its Transformer-based policy with a simpler two-layer linear architecture, or that remove policy guidance entirely via unguided MCTS or random search. As shown in Table 4, ablating any of these factors leads to a clear degradation in performance compared to the full model ($100\% \pm 0\%$ average accuracy). This confirms that each component meaningfully contributes to CRiSP's search capability.

Table 4: Ablation comparing CRiSP with variants that remove the best-game buffer, temperature decay, or Dirichlet noise, or that replace its Transformer-based policy with a simpler policy architecture (two linear layers), or that remove policy guidance entirely via unguided MCTS or random search. Evaluated on MaxCut_12 ($E_{\text{opt}} = -88$), averaged over three seeds (mean $\pm$ std), under the same total evaluation budget as CRiSP's training (60,000 stabilizer evaluations).

	Accuracy	Avg. Energy
CRiSP (Ours)	**100% $\pm$ 0%**	**-88**
Unguided MCTS	$85.23\% \pm 4.95\%$	-75
No Dirichlet noise	$83.33\% \pm 17.69\%$	-73.33
No temperature decay	$79.17\% \pm 4.59\%$	-69.67
Simpler policy	$77.27\% \pm 7.10\%$	-68
No best-game buffer	$70.83\% \pm 11.04\%$	-62.33
Random search	$38.26\% \pm 0.66\%$	-33.67

D Quantum computing background

D.1 Qubits, quantum gates, and the Clifford group

At the core of quantum computation lies the quantum bit, or *qubit*. In stark contrast to deterministic classical bits, a single qubit can occupy a linear combination, or superposition, of its computational basis states, denoted mathematically as $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$. This principle scales exponentially: the state of an N-qubit register is represented by a joint superposition across all 2^N possible classical bitstrings within $\{0,1\}^N$. To evolve these quantum states, one applies continuous unitary transformations referred to as *quantum gates*. Theoretical and experimental focus is frequently directed toward single- and two-qubit operations, as they collectively provide a universal basis capable of executing any arbitrary quantum algorithm Nielsen & Chuang (2010).

Within quantum information theory, the single-qubit Pauli operators, denoted as $\mathcal{P} = \{I, X, Y, Z\}$, serve as a foundational toolkit. The identity matrix I acts trivially by preserving the state, whereas the X operator induces a direct bit flip ($|0\rangle \leftrightarrow |1\rangle$). Conversely, the Z operator functions as a phase flip, appending a negative sign strictly to the excited state ($Z\,|1\rangle = -\,|1\rangle$). The Y gate effectively couples both bit and phase flips, defined as $Y = iXZ$. To describe multi-qubit interactions, the N-qubit Pauli group $\mathcal{P}_N$ is constructed via the N-fold tensor product of these fundamental single-qubit operators, such that $\mathcal{P}_N = \mathcal{P}^{\otimes N}$.

Expanding on this foundation yields the *Clifford group* $\mathcal{C}_N$, a paramount algebraic structure in quantum computing. For an N-qubit system, $\mathcal{C}_N$ encompasses all quantum operations C that act as normalizers for the Pauli group $\mathcal{P}_N$. This requirement dictates that mapping a Pauli operator through a Clifford gate must return another valid Pauli operator, expressed mathematically as:

$$CPC^\dagger \in \pm\mathcal{P}_N \quad \forall\, P \in \mathcal{P}_N. \tag{2}$$

The transformation $CPC^\dagger$ is formally known as the *conjugation* of a Pauli matrix P by a Clifford operation C, which maps to a new Pauli string, potentially with an inverted sign. Because Pauli operators trivially normalize themselves, the Pauli group inherently exists as a strict subgroup within the broader Clifford group ($\mathcal{P}_N \subset \mathcal{C}_N$).

D.2 Universal quantum gate sets and associated costs

To perform arbitrary quantum computations, a universal gate set is necessary to approximate any unitary operator to a desired precision. In the context of fault-tolerant quantum computing (FTQC), the discrete Clifford+T set is a standard choice. By themselves, Clifford operations—such as the Pauli, Hadamard, and CNOT gates—are non-universal. However, supplementing them with the non-Clifford T gate unlocks universality, allowing for the precise approximation of continuous operations (Kliuchnikov et al., 2012). Tools like Gridsynth (Ross & Selinger, 2016) are used to compile continuous arbitrary gates into discrete Clifford+T sequences. The drawback is that achieving higher

mathematical precision necessitates exponentially longer gate sequences, leading to a massive computational overhead that is especially prohibitive in the early FTQC era.

This bottleneck is a direct consequence of the Eastin-Knill theorem (Eastin & Knill, 2009), which establishes that no quantum error correction (QEC) code can natively and fault-tolerantly implement a universal gate set. As a result, standard QEC schemes like the surface code can easily protect Clifford operations but fundamentally cannot correct errors for non-Clifford gates such as the T gate or continuous $R_z(\theta)$ rotations.

To execute these non-Clifford operations, FTQC architectures rely on a resource-intensive procedure known as *magic state distillation* (or T state distillation) (Bravyi & Kitaev, 2005; Litinski, 2019). This method generates high-fidelity resource states, such as the T state $|T\rangle = (1/\sqrt{2})(|0\rangle + \exp\left(i\frac{\pi}{4}\right)|1\rangle)$. Once prepared, these low-error T states are consumed alongside an arbitrary data qubit to successfully apply a T gate to the circuit, though this process incurs substantial qubit-cycle penalties.

The physical implementation of this process occurs in "magic state factories" (Litinski, 2019). The procedure begins by initializing several surface code patches in the $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$ state and applying a series of logical rotations. A designated subset of these qubits is then measured to extract an error syndrome. If this signature flags an error during the logical rotations, the entire batch is discarded, and the protocol restarts. If the signature is clear, the remaining unmeasured qubits yield distilled T states. Because the syndrome measurements cannot catch every possible error type, the final T states are of exceptionally high fidelity, but not strictly perfect.

The footprint of a T state distillation protocol is largely defined by its input-to-output patch ratio, alongside the spatial and temporal code distances (d_X, d_Z, and d_m). For example, a standard (15-to-1)$_{7,3,3}$ factory consumes 15 raw input patches to generate a single distilled T state, utilizing distances of $d_X = 7$, $d_Z = 3$, and $d_m = 3$. Suppressing the error rate of the output T states requires scaling up these dimensions, which directly inflates the space-time volume of the factory. Consequently, protocols designed for higher output yields—such as a (20-to-4) factory—demand significantly greater hardware resources than a basic (15-to-1) configuration operating under the same code distances.

D.3 Achieving utility with VQAs

Although near-term quantum platforms are inherently noisy, specific algorithmic frameworks—most notably variational quantum algorithms (VQAs)—offer a potential pathway to surpassing classical computational capabilities. At its core, a VQA evaluates a parameterized quantum circuit (the ansatz) with respect to a problem-specific Hermitian operator (the Hamiltonian) to compute a scalar objective function, typically the system's energy. This energy landscape is then navigated by classical optimization routines, such as Nelder-Mead or SPSA (Tilly et al., 2022), which iteratively update the circuit parameters to locate the minimum. A key theoretical advantage of VQAs is their Optimal Parameter Resilience (OPR) (Wang et al., 2021), a property suggesting that the optimal parameter configurations discovered under noisy hardware conditions generally align with the ideal, noiseless solutions. In practice, however, the sheer magnitude of noise in the Noisy Intermediate-Scale Quantum (NISQ) era frequently overwhelms this inherent robustness, necessitating explicit error reduction techniques to solve complex, practical problems.

Historically, efforts to suppress errors on near-term hardware have heavily favored quantum error mitigation. Although a vast array of mitigation protocols has been developed for NISQ architectures (Dangwal et al., 2023; Ravi et al., 2021; 2023; 2022; Li et al., 2022; Jin et al., 2024; Seifert et al., 2024; Giurgica-Tiron et al., 2020), these techniques alone have proven insufficient to bridge the gap to practical utility. To genuinely surpass classical computational capabilities, many VQA applications demand extremely high operational fidelities—often exceeding 99%—which mitigation alone struggles to deliver under severe hardware noise. Consequently, while advancing novel error mitigation strategies remains important, there is a growing consensus that transitioning toward early, lightweight, or partial implementations of quantum error correction is an unavoidable prerequisite for sufficiently suppressing noise and realizing true quantum advantage.

D.4 SOLVING COMBINATORIAL OPTIMIZATION PROBLEMS WITH QAOA

Combinatorial optimization fundamentally involves identifying the optimal configuration from an exponentially large discrete search space. This paradigm underpins critical applications across logistics (Sbihi & Eglese, 2010), finance (Odili, 2017), and operations research (Rardin, 1998). Because these problems are overwhelmingly NP-hard, finding exact classical solutions becomes computationally intractable as system sizes scale. In this context, problem formulation plays a pivotal role. The most common formulation is Quadratic Unconstrained Binary Optimization (QUBO), defined as $\min x^T Q x$ where $x \in \{0,1\}^n$ and $Q \in \mathbb{R}^{n\times n}$ is symmetric. This can be extended to Polynomial Unconstrained Binary Optimization (PUBO) to capture higher-order interactions ($\min \sum_i a_i x_i + \sum_{i<j} b_{ij} x_i x_j + \sum_{i<j<k} c_{ijk} x_i x_j x_k + \cdots$), and further generalized to Polynomial Constrained Binary Optimization (PCBO) by incorporating constraints. While PCBO models offer greater expressivity, they introduce significant computational complexity.

Classical exact solvers like CPLEX and Gurobi (IBM, 2024; Gurobi Optimization, LLC, 2024) guarantee optimality through branch-and-bound and relaxation methods, but they suffer from exponential scaling. Conversely, heuristic approaches like simulated annealing and genetic algorithms (Boros et al., 2007) provide approximate scalability for larger problem instances. Often, PCBO structures are quadratized into standard QUBO forms (Boros & Hammer, 2002; Anthony et al., 2017) to leverage existing solvers, though this comes at the expense of introducing auxiliary variables. Consequently, practically scaling these solutions continues to depend heavily on heuristic and approximation frameworks (Kochenberger et al., 2014; Glover et al., 2019).

To address these scaling limitations computationally, the Quantum Approximate Optimization Algorithm (QAOA) (Farhi et al., 2014) offers a hybrid quantum-classical approach tailored for optimization tasks. A standard QAOA circuit is constructed by alternately applying p layers of two parameterized unitaries: a problem-specific cost operator $U(\gamma, H_C) = e^{-iH_C\gamma}$ and a mixing operator $U(\beta, H_M) = e^{-iH_M\beta}$. Starting from an initial uniform superposition of all computational basis states, $|s\rangle = \frac{1}{\sqrt{2^n}} \sum_z |z\rangle$, the system is variationally evolved for p repetitions to approximate the low-energy configurations of the target cost Hamiltonian H_C.

Unlike the Variational Quantum Eigensolver (VQE), which typically targets physical ground states, QAOA is specifically designed to minimize classical discrete cost functions. However, because the standard QAOA ansatz restricts expressivity through its rigid, problem-specific layered structure, navigating its non-convex, instance-dependent optimization landscape is notoriously difficult (Wang et al., 2018; Farhi et al., 2022; Barak et al., 2015; Hastings, 2019; Marwaha, 2021; Chou et al., 2022; Lin & Zhu, 2016; E. Farhi, J. Goldstone, and S. Gutmann, 2015; Hadfield et al., 2019; Streif & Leib, 2020). While increasing the circuit depth (p) theoretically improves expressivity, it concurrently amplifies noise susceptibility on near-term hardware.

An alternative strategy to enhance expressivity without deepening the ansatz is to expand the classical parameter space per layer via multi-angle QAOA (ma-QAOA) (Herrman et al., 2022). Rather than tying all operations in a layer to a single pair of variables, ma-QAOA assigns independent variational angles to each distinct term within the cost and mixer Hamiltonians:

$$U(H_C, \vec{\gamma_\ell}) = e^{-i\sum_{a=1}^{m} H_{C,a}\gamma_{\ell,a}} = \prod_{a=1}^{m} e^{-iH_{C,a}\gamma_{\ell,a}}, \tag{3}$$

$$U(H_M, \vec{\beta_\ell}) = e^{-i\sum_{b=1}^{n} H_{M,b}\beta_{\ell,b}} = \prod_{b=1}^{n} e^{-iH_{M,b}\beta_{\ell,b}}. \tag{4}$$

For a system with m clauses and n qubits, this expands the parameter count from the standard $2p$ to $(m+n)p$. Standard QAOA naturally emerges as a special case of this formulation, whereas ma-QAOA allows classical optimizers to explore a vastly richer, multidimensional solution landscape.

Despite this increased expressivity, QAOA remains highly sensitive to parameter initialization (Sack & Serbyn, 2021; Blekos et al., 2024). Poor starting configurations frequently trap the optimizer in suboptimal local minima or barren plateaus lacking useful gradient information (Lee et al., 2023). Existing initialization heuristics, such as INTERP (Zhou et al., 2020) or TQA (Sack & Serbyn, 2021), are typically customized for specific problem structures and struggle to generalize or scale seamlessly. This lack of robust, universally applicable initialization remains a major barrier to deploying QAOA for complex, real-world tasks.

Once a viable starting point is established, the continuous optimization loop updates the parameters within the interval $[-\pi, \pi)$—extending the search well beyond discrete Clifford angles. Depending on the landscape's smoothness and the presence of hardware noise, this loop may employ gradient-free algorithms like COBYLA (Zhang, 2023), SPSA (Spall, 2002), or Nelder-Mead (Gao & Han, 2012), or gradient-based methods like L-BFGS-B (Zhu et al., 1997) and Adam Kingma & Ba (2017). This iterative refinement proceeds until the algorithm converges or triggers a predefined iteration limit to conserve quantum resources.

Given the extreme non-convexity of the QAOA landscape, a single optimization trajectory—even one initialized with a high-quality warm-start—carries a substantial risk of converging to an unfavorable local minimum (Sack & Serbyn, 2021; Lee et al., 2024). To mitigate this vulnerability, multi-start optimization strategies (Rinnooy Kan & Timmer, 1987) are widely employed, launching several independent optimization routines from a diverse collection of initial seeds to ensure robust convergence.

To empirically assess our work in the QAOA setting, we select a diverse set of optimization tasks. First, we leverage MaxCut as our primary QUBO benchmark. For a graph G with edge set $E(G)$, the corresponding cost Hamiltonian is defined as $H_C = \sum_{(i,j)\in E(G)} \frac{1}{2} w_{ij} (I - Z_i Z_j)$, where w_{ij} denotes the weight of edge (i, j), and Z_i and Z_j are Pauli-Z operators acting on the qubits associated with vertices i and j, respectively. MaxCut serves as an effective benchmark for evaluating our framework, enabling systematic analysis across varied graph topologies under a unified optimization setting. Specifically, we evaluate on weighted complete graphs, which exhibit dense connectivity, leading to highly entangled and globally coupled optimization landscapes.

For PCBO evaluation, we consider the Knapsack problem and a biomedical high-order interaction task Shah et al. (2025) as representative benchmarks. The Knapsack problem is mapped to a QUBO Hamiltonian using the Qiskit Optimization module Qiskit contributors (2023), where constraint satisfaction (e.g., capacity limits) is enforced through penalty terms. Problem instances are generated with randomized item weights and values to ensure a range of nontrivial optimization scenarios.

For the biomedical benchmark Shah et al. (2025), each binary variable corresponds to a gene-associated feature, with bit assignments indicating selection within a subset. The PCBO representation captures both individual feature relevance and higher-order statistical dependencies—including pairwise and triple-wise mutual information—modeled as a weighted hypergraph. The objective is to select M features from a total of N candidates to maximize predictive relevance while reducing redundancy. This selection constraint is imposed via a Hamming-weight condition, enforced through an additional Lagrangian penalty term. In our experiments, we set $M = 4$ for the MedicalDense tasks, $M = 8$ for the 15-qubit MedicalSparse tasks, and $M = 10$ for the MedicalSparse_20 task. Further details regarding these QAOA benchmarks can be found in Bharadwaj et al. (2026).

D.5 Solving scientific problems with VQE

A prominent implementation of the VQA paradigm is the Variational Quantum Eigensolver (VQE) Tilly et al. (2022), which is explicitly designed to estimate the ground-state energy—or the minimum eigenvalue—of a target Hamiltonian $\mathcal{H}$. VQE has become a cornerstone algorithm in computational physics and chemistry, routinely applied to compute molecular dissociation curves as a function of bond length or to resolve the minimum energy configurations of quantum spin models. Because of its practical importance, optimizing the reliable execution of VQE has been the subject of extensive architectural and systems-level research Ravi et al. (2023); Grimsley et al. (2019); Wang et al. (2022); Ravi et al. (2021). In this study, we leverage VQE to investigate fundamental physical systems, detailing the specific target models below.

Within the realm of statistical mechanics, the Ising Cipra (1987) and Heisenberg Bonechi et al. (1992) models serve as foundational frameworks for describing interacting magnetic spin systems. These models are renowned for hosting rich quantum phenomena, including dimension-dependent phase transitions and highly entangled, non-classical ground states. Our analysis specifically targets one-dimensional configurations governed by uniform coupling parameters. The first is the one-dimensional transverse-field *Ising model*, defined by the Hamiltonian:

$$\mathcal{H} = J \sum_{i=1}^{N-1} X_i X_{i+1} + \sum_{i=1}^{N} Z_i. \tag{5}$$

This operator models a chain of N coupled spins (qubits) where nearest-neighbor interactions along the x-axis are governed by a coupling strength J, while each spin simultaneously interacts with a uniform external magnetic field of unit strength aligned along the z-axis.

Additionally, we examine a restricted variant of the field-free Heisenberg model, which generalizes spin-spin coupling across all spatial dimensions. Specifically, we focus on the Hamiltonian:

$$\mathcal{H} = \sum_{i=1}^{N-1} (JX_iX_{i+1} + JY_iY_{i+1} + Z_iZ_{i+1}). \tag{6}$$

In this formulation, the system's energy scale is normalized by fixing the ZZ-interaction at unit strength. Because the transverse interactions along both the X and Y axes are constrained to share an identical coupling constant J, this specific parameterization is conventionally identified as the *XXZ model*.

E NUMERICAL RESULTS FOR QAOA AND VQE BENCHMARKS

Table 5 shows the numerical results for QAOA benchmarks, and the simulation outcomes for VQE tasks are provided in Table 6.

Table 5: QAOA benchmark results. n and N_{params} denote the number of qubits and circuit parameters, respectively. E_{opt} represents the ground state energy of the problem. CRiSP and SPIQ are compared under equal circuit evaluation budgets. SPIQ-R denotes SPIQ evaluated with the same number of update rounds as CRiSP. Results are reported as mean $\pm$ standard deviation (std) across three independent runs. † indicates tasks where the incremental horizon curriculum is applied. Bold values denote the best average performance for each task. MC: MaxCut, Kn: Knapsack, MD: MedicalDense, MS: MedicalSparse.

Category	Task	n	N_{params}	E_{opt}	CRiSP (Ours) (%)	SPIQ (%)	SPIQ-R (%)
MaxCut	MC_8	8	36	-46	**100** $\pm$ 0	83.33 ± 16.02	86.96 ± 18.95
	MC_12	12	78	-88	**100** $\pm$ 0	47.35 ± 9.46	51.89 ± 11.44
	MC_16	16	136	-155	**80.22** $\pm$ 0.74	33.55 ± 6.15	37.85 ± 9.51
	MC_20	20	210	-192	**53.99** $\pm$ 6.1	4.69 ± 4.45	9.55 ± 4.18
Knapsack	Kn_4	8	44	-255	**99.74** $\pm$ 0.45	91.63 ± 2.27	91.63 ± 2.27
	Kn_9	14	119	-2309.5	**94.56** $\pm$ 4.15	76.24 ± 19.86	81.50 ± 16.75
	Kn_12	18	189	-7079.5	**92.18** $\pm$ 6.59	15.34 ± 12.83	29.68 ± 4.87
	Kn_16	22	275	-19538.5	**93.61** $\pm$ 2.85	3.6 ± 6.23	4.26 ± 7.2
MedicalDense	MD_12†	12	310	-24013.6	**99.98** $\pm$ 0.01	73.58 ± 6.36	86.08 ± 10.49
	MD_14†	14	483	-26014.2	**96.14** $\pm$ 3.84	37.16 ± 4.44	47.41 ± 8.88
	MD_20†	20	1370	-32023.8	**93.71** $\pm$ 6.25	2.08 ± 3.6	4.16 ± 3.6
MedicalSparse	MS_15_1†	15	83	-0.365	**38.81** $\pm$ 45.02	21.87 ± 19.75	24.91 ± 23.19
	MS_15_2†	15	84	-0.380	**83.74** $\pm$ 22.56	49.82 ± 7.31	51.38 ± 4.74
	MS_20†	20	107	-0.168	**45.11** $\pm$ 24.96	27.17 ± 23.11	34.87 ± 17.46
				Overall Mean	**83.7**	40.53	45.87

Table 6: VQE results on Ising and XXZ models. CRiSP and CAFQA are compared under equal circuit evaluation budgets. Results are reported as mean ± std across three independent runs. Bold values indicate the best average performance for each task.

Model	Task	n	N_{param}	E_{opt}	CRiSP (Ours) (%)	CAFQA (%)
Ising	Ising_0.5	10	40	-10.570	**94.61** $\pm$ 0.00	80.42 ± 0.00
	Ising_1.0	10	40	-12.381	**75.38** $\pm$ 4.66	**75.38** $\pm$ 9.33
	Ising_2.0	10	40	-19.531	**92.16** $\pm$ 0.00	71.68 ± 0.00
XXZ	XXZ_0.5	10	40	-11.665	**77.15** $\pm$ 0.00	74.29 ± 4.95
	XXZ_1.0	10	40	-17.032	**74.37**$\pm$ 3.39	72.41 ± 3.39
	XXZ_2.0	10	40	-28.722	**75.44** $\pm$ 10.64	69.63 ± 0.00
				Overall Mean	**77.93**	73.97