

SCALAR: A Neurosymbolic Framework for Automated Conjecture and Reasoning in Quantum Circuit Analysis

Sean Feeney^{1*}, Pooja Rao², Andreas Klappenecker¹, Reuben Tate³, Yuri Alexeev², Stefano Mensa², Elica Kyoseva², and Stephan Eidenbenz³

¹*Department of Computer Science, Texas A&M University, College Station, USA*

²*NVIDIA Corporation, Santa Clara, USA*

³*Computing and Artificial Intelligence Division (CAI-3), Los Alamos National Laboratory, Los Alamos, USA*

Abstract—In this paper, we present SCALAR (Symbolic Conjecture and LLM-Assisted Reasoning), a neurosymbolic framework for automated conjecture generation in quantum circuit analysis built on top of the CUDA-Q open source framework. The system integrates quantum simulation, symbolic conjecture generation, and LLM-based interpretation. We evaluate SCALAR on 82 MaxCut instances from the MQLib benchmark dataset and extend the analysis to 2,000 randomly generated graphs across four topologies: regular, Erdős-Rényi, Barabási-Albert, and Watts-Strogatz. The framework generates conjectured bounds relating optimal QAOA parameters to graph invariants, including known relationships such as periodicity constraints on the phase separation parameter γ . SCALAR also recovers previously reported parameter transfer phenomena across structurally similar instances. Additionally, the system identifies correlations between graph structural features and optimization landscape properties, which we characterize through invariant-based descriptors. Using CUDA-Q tensor network simulator, we scale experiments to instances of up to 77 qubits. We discuss the accuracy, generality, and limitations of the generated conjectures, including sensitivity to graph class and quantum circuit depth.

Index Terms—Neurosymbolic, Quantum architecture search, Parameterized quantum circuits, Conjecture generation, MaxCut, QAOA

I. INTRODUCTION

Quantum computing and artificial intelligence represent two of the most rapidly advancing frontiers in computational science. Their intersection has emerged as a substantial area of interest for researchers in academia, industry, and government [1], [2]. A central challenge in near-term quantum computing is the classical optimization burden imposed by variational quantum algorithms: finding optimal parameters requires many expensive quantum evaluations, and the optimization landscape is often non-convex and prone to barren plateaus [3], [4]. Understanding how properties of the problem instance (e.g., interaction structure of the cost Hamiltonian) govern these

landscapes, and whether optimal parameters can be inferred without repeated optimization, is therefore of fundamental importance.

Artificial intelligence and machine learning approaches for scientific and mathematical discovery have driven breakthroughs in domains ranging from protein structure prediction [5] to advancing research-level mathematics [6]–[8]. In the quantum computing domain, these methods have been applied to circuit synthesis and compilation, the task of translating a target quantum operation into a sequence of hardware-native gates that can be executed on a real device [9]. They have also been applied to circuit generation, where rather than starting from a known target operation, the goal is to search over candidate circuit architectures to find one that performs well for a given problem [10], [11]. However, these approaches treat parameter optimization as a black box: they can construct or optimize circuits, but offer little insight into *why* optimal parameters take the values they do, or how problem structure governs the optimization landscape. More broadly, the field lacks tools for automated, scalable reasoning about quantum circuit structure and behavior. This work is a first step toward filling this gap.

In this paper we introduce **SCALAR** (Symbolic Conjecture and LLM-Assisted Reasoning Loop), an iterative neurosymbolic framework for quantum circuit analysis. We use the term neurosymbolic to denote the integration of symbolic conjecture generation with neural reasoning components in an iterative discovery process, where each informs subsequent experimental design. We demonstrate its application to the Quantum Approximate Optimization Algorithm (QAOA) [12] on the MaxCut optimization problem. Our framework proceeds iteratively: quantum circuit simulations produce structured data for a knowledge base containing problem instance invariants (i.e., n the number of nodes), and circuit simulation results; automated conjecture generation (via the txGraffiti

* sfeeney@tamu.edu

system [13]) conjectures both linear and non-linear bounds over this table. An LLM layer then interprets and ranks conjectures by tightness. Those conjectures are then used to inform the next experimental design cycle, (e.g., the optimal parameter γ^* initialized as a function of another optimal parameter β^*). A key methodological insight of our work is that violation patterns, rows in the knowledge table that violate the bound, in failed conjectures carry scientific signal as well. The central empirical finding of this paper, that optimal QAOA parameters (γ^*, β^*) are predictable from a small set of graph-structural invariants for graphs sharing a structural fingerprint, emerged directly from analyzing which instances violated a conjectured bound on γ^* , revealing that all violating instances shared the same values for their graph invariants and, upon inspection, identical optimal parameters.

The contributions of this paper are as follows:

- 1) We introduce **SCALAR**, a novel, iterative neurosymbolic framework for quantum circuit analysis, combining quantum simulation, automated symbolic conjecture generation with TxGraffiti, and LLM-assisted interpretation. A key design principle is that instances violating a conjecture are treated as experimental signals rather than failures, and are used to guide the next iteration of the discovery loop.
- 2) We apply SCALAR to QAOA for MaxCut on 82 MQLib [14] benchmark instances, producing a family of analytical bounds relating optimal parameters (γ^*, β^*) to graph-theoretic invariants, including a novel lower bound on γ^* from chromatic number alone, and rediscovering the periodicity of QAOA angles as an emergent conjecture.
- 3) We present QAOA parameter predictability: optimal parameters (γ^*, β^*) are determined by four cheap graph invariants ($n, \bar{d}, \bar{c}, \alpha_{mis}$) across graphs sharing a structural fingerprint at circuit depths $p \leq 2$ (where p denotes the number of QAOA rounds, formally defined in Section III-B), a finding that emerged directly from analyzing conjecture violations.
- 4) We apply SCALAR to 1000 randomly generated graphs across four topology models (Barabási-Albert, Watts-Strogatz, GNM, and regular), extending the parameter predictability finding to diverse graph topologies and revealing topology-specific conjecture structure, including an $O(n^2)$ empirical lower bound on optimizer calls and a sign reversal in the degree- γ^* relationship unique to small-world graphs.

The remainder of this paper is organized as follows. Section II reviews related work. Section III provides background on QAOA, MaxCut, and txGraffiti. Section IV describes the framework and experimental setup. Section V presents our conjecture and universality results. Section VI discusses implications, limitations, and future directions.

The SCALAR framework and experimental code are publicly available at: <https://github.com/sfeeney1897/SCALAR>

II. RELATED WORK

A. QAOA Parameter Concentration and Transferability

A growing body of work has studied the phenomenon of QAOA parameter concentration, wherein optimal parameters for large random instances of a problem class concentrate around a common value [15], [16]. Parameter transferability studies have extended this, showing that parameters optimized on small or random instances can be transferred to related instances with limited performance loss, with transferability explained by local subgraph structure and graph parity [17]. Our work is complementary: rather than studying concentration in the probabilistic or thermodynamic limit, we study the structural conditions, expressible as four graph invariants computable in polynomial time, under which graphs sharing a common fingerprint, exhibit matching optimal parameters at $p = 1$ across graphs that are not necessarily isomorphic, with evidence that the required invariant set grows with circuit depth.

B. Automated Mathematical Conjecture Generation

The Graffiti program of Fajtlowicz [18] and its successors, including Graffiti.pc [19] and txGraffiti [13], established the paradigm of data-driven symbolic conjecture generation in graph theory. These systems filter a database of objects and their invariants (a knowledge table) through heuristics such as Dalmatian filtering to produce non-trivial symbolic inequalities. The GraphCalc package [20] extends this infrastructure with efficient computation of graph invariants. More broadly, machine intelligence for mathematical conjecture generation has been studied in [7], and AI-guided mathematical discovery at research level has been demonstrated by Davies et al. [6] and Feng et al. [8]. Our work applies this conjecture-generation paradigm to a new domain: the knowledge table of quantum circuit behavior.

C. Quantum Architecture Search and Circuit Discovery

Quantum architecture search (QAS) aims to automate the design of quantum circuit ansätze [21]. Related work on circuit synthesis and compilation [9] and generative approaches such as Q-Fusion [10] and QAOA-GPT and the Generative Quantum Eigensolver [11], [22] similarly seek to reduce the manual effort of quantum circuit design. AlphaEvolve [23] demonstrates the broader power of AI-driven discovery in algorithmic and scientific settings. In contrast to prior data-driven approaches in quantum architecture search—such as circuit synthesis and generative design, which focus on constructing circuit architectures, our framework targets automated reasoning about quantum circuit behavior. Through an iterative loop of symbolic conjecture generation and LLM-based interpretation, it explores relationships between problem instance structure and the resulting optimization landscape, rather than searching over circuit designs.

III. BACKGROUND

A. Classical MaxCut

The MaxCut problem is a fundamental combinatorial optimization problem. Given an unweighted graph $G = (V, E)$, the goal is to partition the vertex set V into two disjoint subsets S and \bar{S} such that the number of edges crossing the partition is maximized. For a graph with $n = |V|$ vertices, a feasible solution is encoded as a bit string $b \in \{0, 1\}^n$, where the j -th bit indicates to which subset the vertex j belongs to.

Formally, the MaxCut problem can be expressed as:

$$\text{MaxCut}(G) = \max_{b \in \{0, 1\}^n} \text{cut}(b), \quad (1)$$

where

$$\text{cut}(b) = \sum_{(i,j) \in E} \mathbf{1}[b_i \neq b_j]. \quad (2)$$

Here, $\mathbf{1}[b_i \neq b_j]$ is the indicator function that evaluates to 1 if vertices i and j are assigned to different subsets, and 0 otherwise. In this work we consider only unweighted graphs, so all edge weights are unity. The exact MaxCut value is computed by brute-force enumeration over all 2^n bitstrings, which is feasible for the problem sizes considered here ($n \leq 20$).

B. QAOA for MaxCut

The Quantum Approximate Optimization Algorithm (QAOA) [12] is a variational quantum-classical hybrid algorithm designed for combinatorial optimization. QAOA is defined by a cost Hamiltonian H_C encoding the classical objective, a mixing Hamiltonian H_M , a circuit depth p , parameter vectors $\gamma = (\gamma_1, \dots, \gamma_p)$ and $\beta = (\beta_1, \dots, \beta_p)$, and an initial state $|s_0\rangle$ taken to be the uniform superposition $|+\rangle^{\otimes n}$.

For MaxCut on a graph $G = (V, E)$, the cost Hamiltonian is

$$H_C = \sum_{(u,v) \in E} \frac{1}{2}(1 - Z_u Z_v), \quad (3)$$

and the mixer is the standard transverse-field Hamiltonian

$$H_M = \sum_{v \in V} X_v, \quad (4)$$

where X_v and Z_v denote the Pauli- X and Pauli- Z operators on qubit v . This mixer generates transitions between computational basis states and preserves exploration of the full Hilbert space. More general mixers may be employed provided they do not commute with H_C , though in this work we use the standard transverse-field formulation.

The QAOA circuit of depth p prepares the state

$$|\gamma, \beta\rangle = e^{-i\beta_p H_M} e^{-i\gamma_p H_C} \dots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_C} |+\rangle^{\otimes n}, \quad (5)$$

with parameters¹ $\gamma \in [0, \pi]^p$ and $\beta \in [0, \pi/2]^p$ optimized classically to maximize the expected cut value

$$\langle H_C \rangle = \langle \gamma, \beta | H_C | \gamma, \beta \rangle.$$

¹See Section 7 of the Supplementary Materials of [24] for a proof that these parameter ranges are sufficient.

In this work, parameters are optimized using the Nelder-Mead method, initialized uniformly at random within the admissible ranges, with these bounds enforced throughout optimization. No post-processing or angle normalization (e.g., periodic wrapping) is applied; all reported parameters correspond directly to the optimizer output. The approximation ratio is defined as

$$r = \frac{\langle H_C \rangle_{\text{opt}}}{\text{MaxCut}(G)}. \quad (6)$$

In this work, we use (γ^*, β^*) to denote the numerically optimized parameters found via Nelder-Mead [25], a gradient-free method, applied to the variational objective, with simulations carried out using CUDA-Q [26]. Due to the complexity of the optimization landscape, we note that Nelder-Mead may not always find the globally optimal values of γ and β .

C. txGraffiti: Automated Conjecture Generation

TxGraffiti is a heuristics-based, data-driven Python library for automated mathematical conjecture generation [13]. The name pays homage to a lineage of conjecture generation programs rooted in graph theory, originating with Fajtlowicz's Graffiti [18] and continuing with Graffiti.pc [19], both developed at the University of Houston.

The core input to txGraffiti is a *knowledge table*: a structured dataset in which rows represent mathematical objects and columns represent their properties, typically invariants or any computable feature of interest to the researcher. The core output is a set of *symbolic conjecture expressions*: inequalities of the form $f(\text{invariants}) \leq \text{target}$ or $f(\text{invariants}) \geq \text{target}$ that hold (subject to configurable violation tolerance) across all objects in the table.

A central challenge in conjecture generation is filtering the combinatorially large space of candidate inequalities. TxGraffiti addresses this through Dalmatian filtering and the Morgan filter [13], which suppress trivially true or uninteresting conjectures and enforce that each conjecture must be tight on a minimum number of instances (`min_touches`). Two conjecture modes are used in this work: Graffiti3 (polynomial and composite functional forms) and ConjecturePlayground (convex hull and ratio methods).

In our application, each row of the knowledge table is a quantum circuit instance (a graph G with its QAOA simulation results), and the columns contain both static features (graph instance-theoretic invariants) and dynamic features (circuit output quantities). This is described in detail in Section IV.

D. Graph-Theoretic Invariants

The following graph invariants are used as features in our knowledge table and in the conjectures discovered by the framework. These invariants were selected to provide a compact, interpretable set of structural descriptors capturing complementary aspects of graph topology, including density, local structure, coloring complexity, and independence structure, while remaining computationally tractable across all instances.

For a graph $G = (V, E)$ with $n = |V|$ vertices and $m = |E|$ edges:

- **Mean degree** $\bar{d} = \frac{2m}{n}$: average vertex degree, capturing overall graph density.
- **Mean clustering coefficient** \bar{c} : average local clustering over all vertices, measuring the prevalence of triangle structure [27].
- **Chromatic number** $\chi(G)$: minimum number of colors needed for a proper vertex coloring, reflecting global constraint complexity.
- **Maximum independent set ratio** $\alpha_{\text{mis}} = \alpha(G)/n$: normalized size of the maximum independent set, capturing sparsity and complement structure.
- **Degree assortativity** r_{assort} : Pearson correlation of degrees across edges, measuring degree–degree correlations [28].

These invariants form the core feature set used in the initial knowledge table construction. While additional invariants are available (e.g., through MQLib), we restrict to a compact subset to balance interpretability and computational cost, and to avoid overfitting conjectures to a large feature space.

For smaller instances ($n \leq 20$), invariants such as the maximum independent set are computed approximately using NetworkX [29]. For larger-scale experiments, including the MQLib [14] benchmark instances and randomly generated graph datasets, we incorporate precomputed invariants when available and otherwise restrict to efficiently computable features.

We note that not all invariants are included in the structural fingerprint used for parameter predictability. In particular, chromatic number $\chi(G)$ is excluded from the initial fingerprint due to its higher computational cost and limited additional discriminative power relative to the selected invariants in the low-depth regime; however, it is retained in the knowledge table for conjecture generation and analysis.

IV. METHODOLOGY

A. Framework Overview

Our framework is an iterative pipeline consisting of five stages, illustrated in Fig. 1.

- 1) **Simulation.** Quantum circuit simulations are executed using CUDA-Q [26] with either a statevector backend (exact expectation values, feasible at $n \leq 20$) or a tensor network backend (approximate but scalable to $n \lesssim 100$), producing numerically optimized parameters (γ^*, β^*) for each instance.
- 2) **Knowledge table construction.** Graph-theoretic (static features) and circuit invariants (dynamic features) are assembled into a structured knowledge table, one row per problem instance.
- 3) **Conjecture generation.** TxGraffiti (Graffiti3 and ConjecturePlayground modes) is run over the knowledge table, generating both linear and non-linear symbolic inequality conjectures relating invariants to target quan-

ties (e.g., γ^* , $|\beta^*|^2$), subject to configurable violation tolerance. In Phase 1, TxGraffiti generates on the order of 10^3 candidate conjectures, which are filtered down to a smaller set (tens to low hundreds) based on violation tolerance and slack criteria.

- 4) **Reasoning and interpretation.** An LLM-based reasoning layer interprets the conjecture output, ranks conjectures by slack and tightness, highlights informative bounds, and flags violation patterns for further investigation. In this work we instantiate this layer using Claude Sonnet 4.6 (Anthropic), though the framework is agnostic to the choice of reasoning system. Formal methods, human expert review, or other automated reasoners could serve equally well. The LLM-based layer further prioritizes a subset of conjectures as informative based on tightness and structural interpretability.
- 5) **Violation feedback.** Instances that violate or nearly violate conjectures are analyzed structurally. These violation patterns are treated as informative signals rather than failures, guiding the design of follow-on experiments, including the selection of new graph instances and the introduction of additional invariants.

A key design principle of the framework is that *violations are not failures*. When a conjecture nearly holds but is violated by a consistent set of instances, these violations may indicate that the current feature set is incomplete or that the dataset does not sufficiently capture the relevant structural variation. Analyzing such cases guides both the introduction of additional invariants and the design of new experiments, including expanding the set of graph instances. In our setting, this process led directly to the identification of parameter predictability, as described in Section V-B.

B. Dataset

Our experiments are conducted in two phases, reflecting the iterative nature of the SCALAR framework.

Phase 1: MQLib benchmark instances. We construct the Phase 1 dataset by filtering instances from the MQLib benchmark suite. First, we select all instances with $n \leq 20$ vertices, so that exact MaxCut values can be computed by brute-force enumeration and QAOA can be simulated using a statevector backend on a standard laptop. From this subset, we further restrict to instances with unit edge weights to simplify the MaxCut Hamiltonian. This filtering procedure yields a total of 82 graph instances.

For each instance G , we run QAOA at depths $p = 1$ and $p = 2$ using CUDA-Q with a statevector backend. Parameters (γ^*, β^*) are obtained via Nelder-Mead optimization. We report $|\beta^*|$ throughout for consistency, as the optimization is performed over the restricted domain $\beta \in [0, \pi/2]$, and the sign of β^* is therefore not explored in our parameterization.

Phase 2: Scale-up to diverse graph models. To validate conjecture generalizability beyond the MQLib benchmark, we

²We report $|\beta^*|$ due to symmetry in the QAOA landscape: for the MaxCut Hamiltonian with the standard mixer, the objective is invariant under $\beta \mapsto -\beta$, making the sign of β^* non-identifying for the relationships studied.

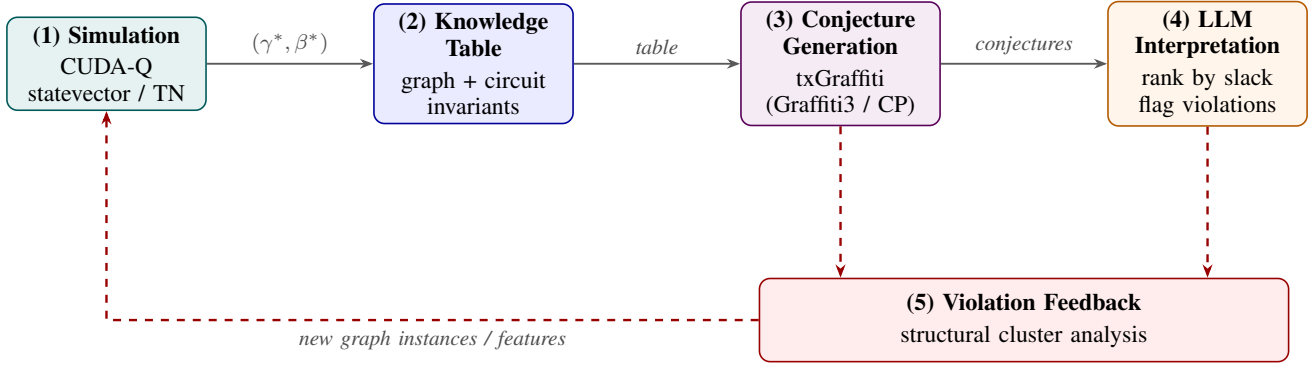


Fig. 1. The proposed human-in-the-loop AI-symbolic reasoning framework. Quantum circuit simulations (Step 1) populate a knowledge table of graph-theoretic and circuit invariants (Step 2), which is processed by txGraffiti to generate symbolic conjecture expressions (Step 3). An LLM interprets and ranks conjectures by slack and tightness, flagging violation patterns (Step 4). Crucially, violation patterns are analyzed structurally and fed back into the next experimental design cycle (Step 5, dashed). The parameter predictability finding reported in Section V-B emerged directly from this feedback loop.

use NetworkX to generate 1000 random graphs across four topology models: Barabási–Albert (scale-free), Watts–Strogatz (small-world), GNM/Erdős–Rényi (structureless random), and d -regular (maximally uniform), with n ranging from 6 to 20. For each model, standard parameterizations are used (e.g., fixed attachment number for Barabási–Albert, rewiring probability for Watts–Strogatz, edge probability for Erdős–Rényi, and fixed degree for regular graphs), with parameters chosen to produce connected graphs; any disconnected instances are discarded. For Phase 2 instances, QAOA is run to depth $p = 5$ using the Nelder–Mead optimizer.

C. Knowledge Table Construction

The knowledge table is the central data structure of the SCALAR framework. Each row corresponds to a problem instance, and columns represent graph-theoretic invariants (static features) together with circuit-derived quantities (dynamic features).

The size and composition of the knowledge table evolve across experimental phases. In Phase 1, the table contains 82 rows corresponding to the filtered MQLib benchmark instances. In Phase 2, it is expanded to include 2000 randomly generated graphs across multiple topology classes.

The core feature set used in the initial table consists of a small number of interpretable graph invariants, including n , m , mean degree \bar{d} , mean clustering coefficient \bar{c} , chromatic number χ , maximum independent set ratio α_{mis} , and degree assortativity r_{assort} , along with circuit-derived quantities such as approximation ratio r and optimized QAOA parameters $(\gamma^*, |\beta^*|)$.

Although the MQLib dataset provides a large number of graph invariants, we restrict to a compact subset to balance interpretability and computational cost. While some invariants such as chromatic number and maximum independent set are NP-hard in general, we compute them approximately or leverage precomputed values for small instances. We also limit the feature set to avoid including highly correlated invariants, as many graph invariants capture overlapping structural properties, which can introduce redundancy and reduce the inter-

pretability of generated conjectures. This improves conjecture quality by focusing on a smaller set of complementary features rather than a large, redundant feature space.

Importantly, the feature set is not fixed a priori. As part of the SCALAR feedback loop, violation patterns identified during conjecture analysis motivate the introduction of additional invariants beyond the initial feature set. For example, degree standard deviation σ_d is introduced in Phase 2 to resolve structural ambiguities between graph classes that share identical values of the initial invariants.

D. Conjecture Generation Configuration

TxGraffiti was run with the following configuration choices, refined through iterative experimentation:

- `max_violations=2-4`: A tolerance of zero violations is too strict for quantum simulation data, which contains small numerical errors from grid-search discretization. Allowing 2–4 violations produces tight, near-universal conjectures while tolerating numerical noise.
- `min_touches=1`: Required to ensure all 82 instances participate in conjecture generation; higher values suppressed valid conjectures.
- `Graffiti3 mode`: Used for polynomial and composite functional forms, targeting γ^* as the output quantity with features including $|\beta^*|$, χ , n , m , and \bar{d} .
- `ConjecturePlayground mode`: Used with convex hull and ratio methods, targeting γ^* with features $|\beta^*|$, \bar{d} , m , and n .

To quantify how well conjectures are satisfied, we use the notion of slack. For a conjecture of the form $f(x) \leq y$, slack is defined as $y - f(x)$; for $f(x) \geq y$, slack is defined as $f(x) - y$. Negative slack indicates a violation.

V. RESULTS

A. Analytical Bounds from Conjecture Generation

Running the SCALAR pipeline on the 82-instance Phase 1 knowledge table yields a set of symbolic conjectures relating γ^* to graph-theoretic and circuit-derived quantities. Table I

summarizes the primary conjectures, together with their slack statistics. We summarize several observed properties of these conjectures.

Linear upper bound (C7). The bound $\gamma^* \leq 0.155|\beta^*| + 2.904$ holds across all 82 instances with near-zero minimum slack. The expression depends only on $|\beta^*|$ and does not include graph-structural quantities such as n or m .

Chromatic-number bound (C2). The inequality $\gamma^* \geq \frac{45}{13}\chi^2 - 4\chi + \frac{5}{2}$ involves only the chromatic number χ and no circuit-derived quantities. This indicates that, within this dataset, chromatic number alone provides a lower bound on γ^* .

Trivial constraint (C4). The bound $\gamma^* \leq 22/7 \approx \pi$ reflects the parameter range imposed during optimization. For unweighted maxcut instances this γ is known to be $\gamma \in [0, \pi]$ [30]. Its appearance serves as a consistency check on the conjecture generation process.

Degree-dependent bounds (C5–C6). These inequalities include negative coefficients on \bar{d} , indicating that larger mean degree is associated with smaller lower bounds on γ^* within this dataset.

B. QAOA Parameter Predictability

This analysis is motivated by investigating violations of conjecture C1. Slack analysis showed that all violating instances shared the same values of $(n, \bar{d}, \bar{c}, \alpha_{\text{mis}}) = (17, 0.5, 0.429, 0.176)$. These graphs are topologically distinct (i.e., not guaranteed to be isomorphic) and exhibit indistinguishable optimized parameters (γ^*, β^*) at $p = 1$. In some cases, graphs within a fingerprint group are isomorphic, for which parameter agreement is expected; however, we also observe agreement across instances that are not trivially identical under isomorphism.

This observation motivates a systematic analysis. We partition the 82 instances by the structural fingerprint $(n, \bar{d}, \bar{c}, \alpha_{\text{mis}})$ and compute the standard deviation of γ^* and β^* within each group of size ≥ 2 . The results are shown in Table II.

Across the 14 repeated-fingerprint groups, all but one group have near-zero variance ($\sigma \approx 0$) in both γ^* and β^* at $p = 1$ and $p = 2$. These groups span problem sizes $n \in \{6, 8, 10, 11, 12, 13, 17, 18, 20\}$; the one exceptional group has problem size $n = 14$.

We summarize this observation as an empirical pattern:

Empirical observation (Parameter predictability).

For $p = 1$ and $p = 2$ QAOA on unweighted MaxCut, graphs sharing the same values of the structural fingerprint $(n, \bar{d}, \bar{c}, \alpha_{\text{mis}})$ often exhibit numerically similar optimized parameters (γ^*, β^*) , up to optimization tolerance and potential landscape degeneracy.

This suggests that, within the Phase 1 dataset, optimal parameters are largely determined by a small set of graph invariants. One possible implication is that parameters may be reused across instances with matching fingerprints, reducing the need for repeated optimization. However, this behavior is

not uniform across all instances and is examined further in the Phase 2 analysis.

A limitation of this result is that many fingerprint groups in the Phase 1 dataset contain isomorphic graphs, for which parameter agreement is expected [31]. Only two groups ($n = 13$ and $n = 18$) are confirmed to contain non-isomorphic graphs with matching optimized parameters.

The previously mentioned $n = 14$ exception is discussed in Section V-C, and the generality of this pattern is evaluated in Section V-D.

C. The $n = 14$ Exception and Multi-Basin Landscapes

A notable exception to the parameter predictability pattern occurs at $n = 14$, $\bar{d} = 0.692$. Instances in this fingerprint group do not converge to a single consistent set of optimized parameters (γ^*, β^*) . Instead, optimization results cluster into two distinct regions of parameter space. We verify that these clusters are not related by simple periodic symmetries of the QAOA parameters (e.g., shifts by π or sign flips), suggesting that they correspond to distinct optima under the chosen parameterization.

This behavior indicates that the four-invariant fingerprint $(n, \bar{d}, \bar{c}, \alpha_{\text{mis}})$ is insufficient to uniquely determine the optimized parameters for this class of graphs. We emphasize that this observation does not distinguish between multiple possible explanations: the underlying optimization landscape may contain multiple basins, or different instances may exhibit distinct optimal parameters despite sharing the same fingerprint. Our results provide evidence for the latter, namely that the chosen invariant set does not fully capture the structural variation relevant to QAOA parameter selection.

This behavior suggests that the four-invariant fingerprint $(n, \bar{d}, \bar{c}, \alpha_{\text{mis}})$ is insufficient to fully characterize the optimization landscape for this class of graphs. Additional structural features, such as degree assortativity or chromatic number, may distinguish between the observed basins, though identifying the minimal distinguishing set remains an open question.

D. Phase 2: Generalization and Invariant Refinement

The Phase 1 analysis suggests that QAOA parameters are largely determined by a small set of graph invariants. We next test whether this pattern generalizes to a broader set of graph structures. To do this, we apply the same four-invariant fingerprint $(n, \bar{d}, \bar{c}, \alpha_{\text{mis}})$ to the Phase 2 dataset of 1000 randomly generated graphs spanning multiple topology models. Repeating the groupby analysis shows that the consistency observed in Phase 1 does not fully carry over: at $p = 1$, the fraction of repeated-fingerprint groups with near-zero parameter variance drops from 92.9% to 54.0%.

We examine the source of this discrepancy. In many cases, graphs with identical values of $(n, \bar{d}, \bar{c}, \alpha_{\text{mis}})$ produce different optimized parameters. A common example occurs between d -regular graphs and Watts–Strogatz graphs, which can share the same mean degree and clustering coefficient but differ in the distribution of degrees.

TABLE I

CONJECTURES GENERATED ON THE 82-INSTANCE PHASE 1 DATASET. MEAN SLACK AND MINIMUM SLACK ARE REPORTED ACROSS ALL INSTANCES. NEGATIVE MINIMUM SLACK INDICATES SMALL VIOLATIONS (PERMITTED BY $\text{MAX_VIOLATIONS} > 0$). C4[†] IS INCLUDED AS A SANITY CHECK.

ID	Conjecture	Type	Mean slack	Min slack
C1	$\gamma^* \geq \frac{1}{3} \beta^* ^2 - \frac{14}{15} \beta^* + 2$	Lower (quadratic)	0.89	-0.008
C2	$\gamma^* \geq \frac{45}{13}\chi^2 - 4\chi + \frac{5}{2}$	Lower (chromatic)	0.93	-0.012
C3	$\gamma^* \leq -\frac{1}{18} \beta^* ^2 + \frac{1}{4} \beta^* + \frac{23}{8}$	Upper (quadratic)	0.49	-0.026
C4 [†]	$\gamma^* \leq \frac{22}{7} \approx \pi$	Upper (trivial)	—	0
C5	$\gamma^* \geq -2.484\bar{d} + 3.287 - 0.015m$	Lower (degree/edges)	0.791	0
C6	$\gamma^* \geq -2.270\bar{d} + 3.164 - 0.014m$	Lower (degree/edges)	0.784	0
C7	$\gamma^* \leq 0.155 \beta^* + 2.904$	Upper (linear)	0.51	≈ 0

TABLE II

STANDARD DEVIATION OF OPTIMAL QAOA PARAMETERS WITHIN EACH STRUCTURAL FINGERPRINT GROUP $(n, \bar{d}, \bar{c}, \alpha_{\text{mis}})$ HAVING TWO OR MORE INSTANCES (14 GROUPS TOTAL). ALL σ VALUES ARE COMPUTED ACROSS TOPOLOGICALLY DISTINCT GRAPHS SHARING THE SAME FINGERPRINT. THIRTEEN OF 14 GROUPS SHOW $\sigma \approx 0$ AT BOTH $p = 1$ AND $p = 2$, CONFIRMING PARAMETER UNIVERSALITY. THE SINGLE EXCEPTION ($n = 14$, $\bar{d} = 0.692$, HIGHLIGHTED) EXHIBITS TWO DISTINCT OPTIMIZATION LANDSCAPE BASINS AND IS DISCUSSED IN SECTION V-C.

n	\bar{d}	\bar{c}	α_{mis}	count	$\sigma(\gamma_1^*)$	$\sigma(\beta_1^*)$	$\sigma(\gamma_2^*)$	$\sigma(\beta_2^*)$
6	0.600	0.000	0.500	3	9.7×10^{-5}	6.7×10^{-5}	1.6×10^{-4}	2.8×10^{-4}
8	0.429	0.000	0.500	3	8.8×10^{-6}	3.5×10^{-5}	1.6×10^{-4}	1.5×10^{-4}
8	0.750	0.600	0.375	2	4.4×10^{-4}	3.6×10^{-5}	4.5×10^{-4}	5.1×10^{-4}
10	0.333	0.000	0.400	5	5.0×10^{-4}	2.0×10^{-4}	3.5×10^{-4}	5.6×10^{-4}
11	0.364	0.000	0.455	2	0	0	0	0
12	0.455	0.500	0.250	5	9.0×10^{-8}	1.6×10^{-9}	4.5×10^{-8}	2.8×10^{-7}
13	0.500	0.452	0.231	2	2.4×10^{-3}	2.0×10^{-3}	7.5×10^{-4}	2.4×10^{-3}
14	0.692	0.648	0.143	2	5.2×10^{-1}	1.6×10^0	3.9×10^{-1}	1.3×10^{-1}
17	0.500	0.429	0.176	6	5.1×10^{-4}	1.2×10^{-4}	1.0×10^{-3}	5.4×10^{-4}
18	0.314	0.533	0.278	2	1.0×10^{-3}	1.1×10^{-3}	1.0×10^{-6}	5.8×10^{-5}
20	0.137	0.000	0.500	2	0	0	0	0
20	0.158	0.000	0.350	2	0	0	0	0
20	0.158	0.000	0.400	3	0	0	0	0
20	0.263	0.300	0.250	4	4.1×10^{-4}	6.9×10^{-4}	3.0×10^{-6}	1.0×10^{-4}

TABLE III

UNIVERSALITY RATES (FRACTION OF REPEATED-FINGERPRINT GROUPS WITH $\sigma(\gamma^*, \beta^*) < \varepsilon$) FOR FINGERPRINTS OF INCREASING SIZE, EVALUATED AT CIRCUIT DEPTHS $p = 1$, $p = 2$, AND $p = 5$ ON THE PHASE 2 DATASET. HERE, ε DENOTES A SMALL TOLERANCE THRESHOLD (SET TO 10^{-3} IN OUR EXPERIMENTS). RESULTS ARE COMPUTED WITHIN EACH GRAPH MODEL SEPARATELY (I.E., INSTANCES ARE GROUPED BY BOTH FINGERPRINT AND TOPOLOGY CLASS). ADDING σ_d ALONE INCREASES $p = 1$ UNIVERSALITY FROM 60.9% TO 96.6%.

Fingerprint	$p = 1$	$p = 2$	$p = 5$
4-invariant: $(n, \bar{d}, \bar{c}, \alpha_{\text{mis}})$	60.9%	16.3%	8.7%
5-invariant: $+ \sigma_d$	96.6%	27.6%	11.5%
5-invariant: $+ r_{\text{assort}}$	—	—	—
6-invariant: $+ \sigma_d + r_{\text{assort}}$	98.6%	39.4%	19.7%

This observation suggests that the four-invariant fingerprint does not capture all relevant structural variation. In particular, degree standard deviation σ_d distinguishes these cases: it is zero for regular graphs and non-zero for Watts–Strogatz graphs with the same mean degree. Incorporating σ_d as an additional invariant resolves most of these discrepancies. Table III summarizes the effect of extending the fingerprint with additional invariants across multiple circuit depths.

Several observations follow from Table III.

Degree variance improves separation. Adding degree

standard deviation σ_d increases the fraction of repeated-fingerprint groups with near-zero parameter variance from 60.9% to 96.6% at $p = 1$, resolving most discrepancies observed with the four-invariant fingerprint.

Performance degrades with depth. Even with additional invariants, this consistency decreases as circuit depth increases, dropping from 98.6% at $p = 1$ to 39.4% at $p = 2$ and 19.7% at $p = 5$. This suggests that higher-depth QAOA parameters depend on more detailed aspects of graph structure. One possible explanation is that increasing circuit depth expands the effective interaction radius of the algorithm, allowing it to capture more global features of the graph. In contrast, many of the invariants used in our feature set (e.g., mean degree, clustering coefficient) primarily encode local or aggregate structural properties. As a result, these invariants may be sufficient to characterize low-depth behavior but become less informative as deeper circuits incorporate longer-range correlations and more complex global structure.

First-layer parameters are more stable. At $p = 5$, restricting attention to (γ_1^*, β_1^*) yields higher consistency (32.5%) than considering all parameters jointly (19.7%).

Cross-model agreement at $p = 1$. For groups containing graphs from different topology models, adding σ_d yields consistent parameters at $p = 1$ across all observed cases, whereas the four-invariant fingerprint does not.

E. Additional Patterns at Scale

Applying the framework to the larger Phase 2 dataset also reveals several recurring patterns across graph models and sizes. We summarize the most consistent observations.

Scaling pattern in γ^* . Across multiple conjectures, upper bounds on γ^* consistently include a leading term proportional to n , for example:

$$\gamma^* \leq \frac{1}{4}n + \frac{8}{13}\sqrt{n}, \quad (7)$$

$$\gamma^* \leq \frac{1}{4}n + \frac{7}{6}\sqrt{d}. \quad (8)$$

However, these bounds become vacuous for larger graph sizes. In particular, the right-hand side exceeds the trivial constraint $\gamma^* \leq \pi$ for sufficiently large n (e.g., $n \geq 7$ for the first bound), indicating that these expressions are primarily informative in the small- n regime. This reflects the scale of the instances considered and suggests that the observed scaling behavior is most relevant for small to moderate problem sizes. More generally, it highlights a limitation of unconstrained conjecture generation, which may produce bounds that are locally tight but do not remain informative asymptotically.

Approximation ratio bounds. The conjectures relate approximation ratio r to structural properties such as assortativity and independence number. For example:

$$r \leq 1 - \frac{3}{28}\sqrt{r_{\text{assort}}}, \quad (9)$$

$$r \leq \left\lfloor \frac{1}{12}n + \sqrt{\alpha_{\text{mis}}} \right\rfloor. \quad (10)$$

These expressions indicate that approximation quality varies with graph structure in a measurable way. We note that degree assortativity r_{assort} may take negative values. The above expression is reported as generated by TxGraffiti from the observed dataset, and is therefore interpreted over the range of values present in that data, not a general prediction.

Optimizer call scaling. The conjectures include lower bounds on the number of objective function calls required by Nelder–Mead optimization:

$$\frac{22}{17}n^2 + 4n + 4 \leq \text{obj}. \quad (11)$$

This is consistent with the quadratic scaling of simplex-based methods with respect to the number of optimization parameters. [25]

Topology-dependent behavior. When conjectures are evaluated within individual graph models, differences in structure become apparent. For example, in Watts–Strogatz graphs, bounds on γ^* include a negative coefficient on mean degree, whereas other models exhibit positive coefficients. This indicates that the relationship between graph structure and optimized parameters depends on topology.

F. Scaling Demonstration with Tensor Network Simulation

To complement the preceding analysis, we perform a proof-of-feasibility experiment using the CUDA-Q tensor network backend on a single instance with 77 qubits. The SCALAR pipeline is successfully executed in this setting, producing a valid objective value and optimized parameters using the same optimization procedure as in smaller-scale experiments. We

emphasize that this is a single-instance demonstration rather than a systematic scaling study. Nevertheless, it illustrates that the framework can be instantiated beyond the exact statevector regime.

VI. DISCUSSION

A. Main Takeaways

We summarize the main findings of this work. Conjecture-driven analysis produces interpretable relationships. Applying SCALAR to QAOA simulations yields symbolic bounds relating optimal parameters to graph structure. These conjectures provide a compact description of patterns in the data.

Low-depth QAOA parameters exhibit structured regularity. At $p = 1$ and $p = 2$, many graph instances with matching values of a small set of invariants produce numerically similar optimized parameters. This suggests that, in this regime, QAOA behavior is largely determined by coarse structural features.

Invariant choice is critical and problem-dependent. The Phase 2 analysis shows that initial invariants are not sufficient to generalize across graph families. Introducing additional features, such as degree variance, improves consistency but does not fully resolve the problem, especially at higher circuit depths.

QAOA behavior becomes more instance-specific with depth. As circuit depth increases, the consistency of optimized parameters across similar instances decreases. This indicates that higher-depth QAOA depends on finer-grained structural properties not captured by simple invariant sets.

Automated conjecture generation can guide analysis. The SCALAR workflow identifies patterns, highlights exceptions, and suggests directions for refinement, providing a structured way to explore relationships between quantum circuits and problem structure.

B. Limitations and Caveats

This work represents an initial step toward automated reasoning about quantum circuits, and several limitations should be noted.

All conjectures and patterns reported are empirical and derived from finite datasets. They should not be interpreted as general statements about QAOA beyond the tested instances and parameter ranges.

The analysis is restricted to unweighted MaxCut at low circuit depths ($p \leq 2$), where the strongest patterns are observed. At higher depths, parameter behavior becomes less consistent, and the current invariant sets do not fully capture the underlying structure.

The results depend on the choice of graph invariants used to construct the knowledge table. While additional invariants can improve performance, there is no guarantee that a small set of features will suffice in general.

Optimized parameters are obtained through numerical methods and may be affected by local minima, initialization, and optimizer settings. Observed agreement between instances

is therefore subject to optimization tolerance and potential landscape degeneracy.

Although SCALAR automates conjecture generation, the identification and interpretation of meaningful invariants currently requires human input. The system is not fully autonomous and relies on domain knowledge to guide refinement.

Finally, the datasets considered are limited in scope. Phase 1 uses a small set of benchmark instances, and Phase 2 relies on synthetic graph models. While these provide useful test cases, they do not cover the full range of problem instances encountered in practice.

C. Future Work

The conjectures that SCALAR finds with the help of txGraffiti [13] can be exported to the Lean 4 [32] proof assistant. Then a formal proof of these conjectures can be formulated and checked in Lean. However, Lean will in general not be able to automatically find proofs of these conjectures, except in very simple cases. One reason is that the grind tactic tries to avoid computationally expensive graph invariants, and other tactics follow a similar approach for efficiency reasons. However, Lean is starting to offer more support for graphs and graph invariants. Writing custom tactics for automated theorem proving of graphs conjectures is an interesting direction to pursue in the future.

VII. CODE AND DATA AVAILABILITY

The code, datasets, and experiment configurations used in this work are available at: <https://github.com/sfeeney1897/SCALAR>

ACKNOWLEDGMENTS

The research presented in this article was supported by the NNSA's Advanced Simulation and Computing Beyond Moore's Law Program at Los Alamos National Laboratory. LANL report LA-UR-26-23374.

REFERENCES

- [1] Y. Alexeev, M. H. Farag, T. L. Patti, M. E. Wolf, N. Ares, A. Aspuru-Guzik, S. C. Benjamin, Z. Cai, S. Cao, C. Chamberland *et al.*, "Artificial intelligence for quantum computing," *Nature Communications*, vol. 16, no. 1, p. 10829, 2025.
- [2] U.S. Department of Energy, "Genesis Mission: A national mission to accelerate science through artificial intelligence," <https://genesis.energy.gov/>, 2025, accessed: April 2026.
- [3] M. Larocca, S. Thanasilp, S. Wang, K. Sharma, J. Biamonte, P. J. Coles, L. Cincio, J. R. McClean, Z. Holmes, and M. Cerezo, "Barren plateaus in variational quantum computing," *Nature Reviews Physics*, vol. 7, no. 4, pp. 174–189, 2025.
- [4] E. R. Anschuetz and B. T. Kiani, "Quantum variational algorithms are swamped with traps," *Nature Communications*, vol. 13, no. 1, p. 7760, 2022.
- [5] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko *et al.*, "Highly accurate protein structure prediction with alphafold," *nature*, vol. 596, no. 7873, pp. 583–589, 2021.
- [6] A. Davies, P. Veličković, L. Buesing, S. Blackwell, D. Zheng, N. Tomašev, R. Tanburn, P. Battaglia, C. Blundell, A. Juhász *et al.*, "Advancing mathematics by guiding human intuition with ai," *Nature*, vol. 600, no. 7887, pp. 70–74, 2021.
- [7] C. Mishra, S. R. Moulik, and R. Sarkar, "Mathematical conjecture generation using machine intelligence," 2023. [Online]. Available: <https://arxiv.org/abs/2306.07277>
- [8] T. Feng, T. Trinh, G. Bingham, J. Kang, S. Zhang, S.-h. Kim, K. Barreto, C. Schildkraut, J. Jung, J. Seo *et al.*, "Semi-autonomous mathematics discovery with gemini: A case study on the erdős problems," *arXiv preprint arXiv:2601.22401*, 2026.
- [9] G. Yan, W. Wu, Y. Chen, K. Pan, X. Lu, Z. Zhou, Y. Wang, R. Wang, and J. Yan, "Quantum circuit synthesis and compilation optimization: Overview and prospects," 2025. [Online]. Available: <https://arxiv.org/abs/2407.00736>
- [10] C. Beaudoin and S. Ghosh, "Q-fusion: Diffusing quantum circuits," 2025. [Online]. Available: <https://arxiv.org/abs/2504.20794>
- [11] I. Tyagin, M. H. Farag, K. Sherbert, K. Shirali, Y. Alexeev, and I. Safro, "Qaoa-gpt: Efficient generation of adaptive and regular quantum approximate optimization algorithm circuits," in *2025 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 1. IEEE, 2025, pp. 1505–1515.
- [12] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *arXiv preprint arXiv:1411.4028*, 2014.
- [13] R. Davila, "Automated conjecturing in mathematics with TxGraffiti," 2024. [Online]. Available: <https://arxiv.org/abs/2409.19379>
- [14] I. Dunning, S. Gupta, and J. Silberholz, "What works best when? a systematic evaluation of heuristics for max-cut and qubo," *INFORMS Journal on Computing*, vol. 30, no. 3, pp. 608–624, 2018.
- [15] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, "Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices," *Physical Review X*, vol. 10, no. 2, p. 021067, 2020.
- [16] F. G. Brandao, M. Broughton, E. Farhi, S. Gutmann, and H. Neven, "For fixed control parameters the quantum approximate optimization algorithm's objective function value concentrates for typical instances," *arXiv preprint arXiv:1812.04170*, 2018.
- [17] A. Galda, E. Gupta, J. Falla, X. Liu, D. Lykov, Y. Alexeev, and I. Safro, "Similarity-based parameter transferability in the quantum approximate optimization algorithm," *Frontiers in Quantum Science and Technology*, vol. 2, p. 1200975, 2023.
- [18] S. Fajtlowicz, "On conjectures of graffiti," in *Graph Theory and Applications*, ser. Annals of Discrete Mathematics, J. Akiyama, Y. Egawa, and H. Enomoto, Eds. Elsevier, 1988, vol. 38, pp. 113–118. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167506008707763>
- [19] E. DeLaViña, "Graffiti.pc: A variant of graffiti," in *Graphs and Discovery*, ser. DIMACS Series in Discrete Mathematics and Theoretical Computer Science. Providence, RI: American Mathematical Society, 2005, vol. 69, pp. 71–79.
- [20] R. Davila, "Graphcale: A python package for computing graph invariants in automated conjecturing systems," *Journal of Open Source Software*, vol. 10, no. 112, p. 8383, 2025. [Online]. Available: <https://doi.org/10.21105/joss.08383>
- [21] D. Martyniuk, J. Jung, and A. Paschke, "Quantum architecture search: A survey," in *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 01, 2024, pp. 1695–1706.
- [22] K. Nakaji, L. B. Kristensen, R. Kemmoku, J. A. Campos-Gonzalez-Angulo, M. G. Vakili, H. Huang, M. Bagherimehrab, C. Gorgulla, F. Wong, A. McCaskey *et al.*, "The generative quantum eigensolver (gqe) and its application for ground state search," *arXiv preprint arXiv:2401.09253*, 2024.
- [23] A. Novikov, N. Vü, M. Eisenberger, E. Dupont, P.-S. Huang, A. Z. Wagner, S. Shirobokov, B. Kozlovskii, F. J. Ruiz, A. Mehrabian *et al.*, "Alphaevolve: A coding agent for scientific and algorithmic discovery," *arXiv preprint arXiv:2506.13131*, 2025.
- [24] R. Tate and S. Eidenbenz, "Theoretical approximation ratios for warm-started qaoa on 3-regular max-cut instances at depth $p=1$," *Theoretical Computer Science*, vol. 1059, p. 115571, 2026. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0304397525005092>
- [25] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 01 1965. [Online]. Available: <https://doi.org/10.1093/comjnl/7.4.308>
- [26] The CUDA-Q development team, "CUDA-Q," [Online]. Available: <https://github.com/NVIDIA/cuda-quantum>
- [27] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *nature*, vol. 393, no. 6684, pp. 440–442, 1998.

- [28] M. E. J. Newman, "Mixing patterns in networks," *Phys. Rev. E*, vol. 67, p. 026126, Feb 2003. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevE.67.026126>
- [29] A. Hagberg, P. J. Swart, and D. A. Schult, "Exploring network structure, dynamics, and function using networkx," Los Alamos National Laboratory (LANL), Tech. Rep., 2007.
- [30] S. H. Sack and M. Serbyn, "Quantum annealing initialization of the quantum approximate optimization algorithm," *quantum*, vol. 5, p. 491, 2021.
- [31] R. Shaydulin, S. Hadfield, T. Hogg, and I. Safro, "Classical symmetries and the quantum approximate optimization algorithm," *Quantum Information Processing*, vol. 20, no. 11, p. 359, 2021.
- [32] L. de Moura and S. Ullrich, "The Lean 4 theorem prover and programming language," in *International Conference on Automated Deduction*. Springer, 2021, pp. 625–635.