

Quality Diversity for Variational Quantum Circuit Optimization

Maximilian Zorn¹, Jonas Stein^{1,2}, Maximilian Balthasar Mansky¹,
Philipp Altmann¹, Michael Kölle¹, Claudia Linnhoff-Popien¹

¹LMU Munich

²Aqarios GmbH

{maximilian.zorn, jonas.stein, philipp.altmann, maximilian-balthasar.mansky, michael.koelle, linnhoff}@ifi.lmu.de

Abstract

Optimizing the architecture of variational quantum circuits (VQCs) is crucial for advancing quantum computing (QC) towards practical applications. Current methods range from static ansatz design and evolutionary methods to machine learned VQC optimization, but are either slow, sample inefficient or require infeasible circuit depth to realize advantages. Quality diversity (QD) search methods combine diversity-driven optimization with user-specified features that offer insight into the optimization quality of circuit solution candidates. However, the choice of quality measures and the representational modeling of the circuits to allow for optimization with the current state-of-the-art QD methods like covariance matrix adaptation (CMA), is currently still an open problem. In this work we introduce a directly matrix-based circuit engineering, that can be readily optimized with QD-CMA methods and evaluate heuristic circuit quality properties like expressivity and gate-diversity as quality measures. We empirically show superior circuit optimization of our QD optimization w.r.t. speed and solution score against a set of robust benchmark algorithms from the literature on a selection of NP-hard combinatorial optimization problems.

Introduction

As the field of quantum computing (QC) continues to grow – even with the limitations of recent noisy intermediate scale quantum (NISQ) hardware (Preskill 2018) – ideas for utilizing the potential computing speedup of quantum computing have been tested for a wide range of problems (Rønnow et al. 2014). Some examples include application in many different fields of study, from financial prediction (Egger et al. 2020), fraud detection (Kyriienko and Magnusson 2022), image classification (Senokosov et al. 2023) and quantum machine learning (QML) (cf. Biamonte et al. (2017)), material science (Kandala et al. 2017), chemistry simulations (Cao et al. 2019) and combinatorial optimization (Khairy et al. 2020; Li et al. 2020).

In this paper we focus on combinatorial optimization (CO), as a branch of optimization that seeks to find the best solution from a finite set of discrete possibilities. CO plays a critical role in various fields, including computer science, operations research and engineering. In particular, research

communities using planning, scheduling or resource allocation algorithms could benefit from a speedup in the form of a ‘quantum routine’ for CO solutions, e.g., for algorithms based on SAT (Robinson et al. 2009), Knapsack (Mirkis and Domshlak 2013), or versions of Vertex Cover (Zhang et al. 2022), among many other integrations.

Despite its wide applicability, combinatorial optimization often faces significant computational challenges, necessitating the development of efficient algorithms and heuristics to achieve optimal or near-optimal solutions. However, leveraging the potential capabilities of quantum hardware for CO may eventually lead to tangible speedups, as long as quantum circuits enabling this speedup can be constructed. Current methods for designing such circuits range from analytic ansatz design to dynamically constructed variational quantum circuits (VQCs) (Cerezo et al. 2021). VQCs are characterized by their use of parameterized rotation gates, the positioning, and (parameter) tuning of which is usually done via classical optimization methods. VQC optimization can be mainly divided in gradient-based and gradient-free methods. While the former is effective for small circuits, the phenomenon of *barren plateaus* (Larocca et al. 2024), where the optimization landscape of gradient-based optimizers becomes mostly flat and featureless and therefore hard to adjust, makes the optimization of deeper circuits a challenge, in particular in combination with the current noisy hardware (Wang et al. 2021). Gradient-free methods, on the other hand, can avoid this optimization pitfall of barren-plateaus through diverse populations of individuals or evolutionary methods (e.g., Sünkel et al. (2023); Giovagnoli et al. (2023)), but are often slow and sample inefficient, which makes testing both time- and cost-inefficient.

In this work, we enable the integration of quality diversity (QD) driven optimization to the problem of optimizing VQCs. In essence, QD is a universally applicable search method that explores diverse solutions for a given objective function based on low dimensional user-defined characteristic (quality) measures. While QD has many recent examples of successful utilization (Fioravanzo and Iacca 2019; Cully et al. 2015; Zhang et al. 2024), to the best of our knowledge, QD has not yet been developed for the optimization of VQCs. We suspect this is due to a lack of low dimensional selection of quality measures and the way circuits are represented, which are essential for optimization

using current state-of-the-art quality diversity methods like Covariance Matrix Adaptation (CMA). As such, VQC QD-design has remained an unresolved problem, possibly due to the absence of a standardized circuit encoding that would facilitate more efficient circuit search.

In this paper, we resolve both these issues to enable VQC optimization via quality diversity CMA methods. We see the key contributions of this work in the following aspects:

- We propose a lightweight but efficient 1 : 1 encoding of whole quantum circuits into optimizable matrices, which we utilize for the optimization of VQCs on a set of multiple well-known combinatorial optimization problems. We leverage recent state-of-the-art QD methods into our search method, defining NISQ-friendly quality metrics for *circuit expressivity* – heuristically approximated by *circuit sparsity* – and *gate diversity*. The encoding itself, additionally, is designed to be independent of specific quantum hardware, problem specific aspects or domain knowledge and is thus generally available for any kind of optimization.
- We demonstrate the capability of our approach by achieving superior results on combinatorial optimization problems – both in terms of objective function quality and search speed – in direct comparison to two kinds of state-of-the-art circuit construction approaches, a different gradient-free evolutionary VQC optimization method (QNEAT) and the analytically designed circuit approximation optimization algorithm (QAOA).

Related Work

The field of quantum computing is actively exploring hybrid quantum-classical approaches, particularly for circuit construction and optimization, leveraging classical machine learning techniques to navigate the complexities of near-term quantum hardware (Nielsen and Chuang 2010). While research exists in quantum machine learning (Biamonte et al. 2017), quantum neural networks (Beer et al. 2020), and quantum reinforcement learning (QRL/RL) intended to run entirely on quantum computers (Dong et al. 2008; Dunjko, Taylor, and Briegel 2017), practical implementations often rely on classical computation for key optimization tasks due to current hardware limitations (Tilly et al. 2022). Our work contributes to this hybrid paradigm but distinguishes itself from several common strategies.

One prominent application area for automated circuit design is *quantum state preparation*. Several studies utilize QRL (Wu et al. 2020; Gabor, Zorn, and Linnhoff-Popien 2022) or RL combined with specific gate sets (Mackeprang, Dasari, and Wrachtrup 2020) to construct circuits that generate desired target quantum states, such as maximally entangled states. These methods focus on achieving a specific state vector as the primary output objective. In our work we construct circuits intended to solve a given optimization problem directly – skipping any state-preparation steps – where the circuit’s output or measurement statistics encode the solution quality.

Another distinct category involves the *sequential, gate-by-gate construction*, often employing gradient-based (RL) policy updates or related value-based methods. For example, Ostaszewski et al. (2021) used RL to learn an optimal composition of rotation gates by treating the circuit as a grid and placing gates incrementally. Similarly, Kölle et al. (2024) frames gate selection as a classification task managed by an RL agent, placing gates at specific circuit positions sequentially. Techniques like predicting time-sensitive gate controls also fall broadly within this incremental optimization philosophy (An and Zhou 2019). In contrast, our QD methodology optimizes the entire circuit structure, represented as a matrix, in a single holistic step per iteration of our evolutionary algorithm, rather than modifying the circuit one gate or operation at a time based on sequential decisions.

Finally, a third related direction focuses on *optimizing an initially given circuit instance* iteratively, often leveraging gradient-based machine learning techniques. This includes e.g., VQCs, where an initial circuit structure is fixed, and its continuous parameters are tuned using classical gradient-based optimizers to minimize a cost function for tasks like solving linear equations (Bravo-Prieto et al. 2019) or training RL agents (Chen et al. 2020). Other approaches might start with, for instance, randomly sampled circuits and use RL agents to apply discrete transformations to iteratively improve the existing structure (Fösel et al. 2021). We, on the other hand, employ an evolutionary approach that re-samples complete circuit solution candidates from an evolving distribution guided by the optimization objective, rather than performing iterative, gradient-informed updates on a single circuit instance.

Background

Quantum Circuits

Current quantum computing (cf. Nielsen and Chuang (2010)) and their computational circuits rely on the successive application of gates to states. A state carries information, while the gates manipulate the information in correspondence to a quantum algorithm. Numerically, states are represented as complex-valued vectors of dimension 2^n , n the number of qubits. The quantum operation U can be represented by a matrix of $2^n \times 2^n$ dimensions. Individual gates within the operation often have lower-dimensional representations that are concatenated with the tensor product.

Our work approaches this tensored product of gates. In the graphical picture of quantum circuits, our approach can be thought of as the horizontal concatenation of L layers of n vertical tensored gates, for a dense grid of operations. This grid structure is natural to conventional optimization and abstracts the quantum circuit construction.

Moving beyond a specific number of qubits, we generalize our model to a n -qubit setting. In this more general framework, instead of considering only a fixed number of amplitudes or states, we must account for the probability of measuring any of the 2^n possible state combinations in the quantum system. The state of a n -qubit quantum circuit can be expressed as: $|\theta\rangle = \sum_{i=0}^{2^n-1} \alpha_i |i\rangle$, where $|i\rangle$ represents each possible basis state of the n -qubit system,

encoded in binary notation (e.g., $|0\rangle, |1\rangle, |2\rangle, \dots, |2^n - 1\rangle$). Here, α_i are the complex amplitudes corresponding to each state, and i ranges over all possible states. The vector representation of the corresponding Dirac notation for this state is then $|\theta\rangle = [\alpha_0, \alpha_1, \dots, \alpha_{2^n-1}]^\top$.

Finally, to satisfy the fundamental rule of probability, the sum of the squares of the magnitudes of these amplitudes must equal 1, ensuring the state vector is properly normalized. This condition is formulated as $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$.

Combinatorial Optimization

In this paper we will evaluate our QD approach on the four following combinatorial binary optimization graph problems, all of which are frequently utilized as quantum optimization benchmarks. To test for selection or inclusion from the set of vertices $V(G)$ or set of edges $E(G)$ (or its complement $E(\bar{G})$), we measure our solution candidate circuits in basis $Z_i \rightarrow [-1, 1]$, as it is commonly done, where Z_i is the Pauli-Z observable acting on the qubit corresponding to the vertex i of a graph G . The tensor product $Z_i \otimes Z_j$ computes the interaction between two vertices connected by the edge (i, j) , represented on wires i, j .

- The maximum cut (*MaxCUT*) problem divides nodes on a graph into two separate sets (with labels -1 or 1) so that the splitting line goes through the maximum number of edges in the graph G . The cost hamiltonian to maximize is given as:

$$\max -H_{MC} = \sum_{(i,j) \in E(G)} \frac{1}{2} (Z_i \otimes Z_j - 1)$$

- The minimum vertex cover (*MinVEC*) problem involves identifying the smallest set of vertices in a graph such that every edge is incident to at least one vertex in the set, via cost hamiltonian:

$$\max -H_{VC} = 3 \sum_{(i,j) \in E(G)} (Z_i \otimes Z_j + Z_i + Z_j) - \sum_{i \in V(G)} Z_i$$

- The maximum independent set (*MaxIND*) problem seeks the largest subset of vertices in a graph where no two vertices are adjacent with:

$$\max -H_{IS} = 3 \sum_{(i,j) \in E(G)} (Z_i \otimes Z_j - Z_i - Z_j) + \sum_{i \in V(G)} Z_i$$

- The maximum clique (*MaxCLI*) problem aims to find the largest complete subgraph within the given graph, where every pair of vertices in the subgraph is directly connected by an edge. The cost hamiltonian penalizes the complement of all selected edges $E(\bar{G})$ with:

$$\max -H_{CL} = 3 \sum_{(i,j) \in E(\bar{G})} (Z_i \otimes Z_j - Z_i - Z_j) + \sum_{i \in V(G)} Z_i$$

Quality Diversity Methods

Quality diversity methods are a family of gradient-free single-objective optimization algorithms. The diversity aspect of QD is inspired by evolutionary algorithms with diversity optimization such as novelty search (Lehman, Stanley et al. 2008; Lehman and Stanley 2011; Conti et al. 2018),

which similarly optimizes an objective function to diversify a set of novelty measure functions in order to generate a diverse collection of high-quality solutions.

QD algorithms have many different interpretations, incorporating different optimization methods, such as topological search (cf. NEAT (Stanley and Miikkulainen 2002)), gradient descent (Fontaine and Nikolaidis 2021, 2023) or model-based surrogates (Gaier, Asteroth, and Mouret 2018). Many of the more recent QD methods are based on the concept of *MAP-Elites* (Mouret and Clune 2015), which moves away from pure novelty search in favor of exploring a high-dimensional space (of an objective function) with the intention of finding high-performing solutions at each point in a low-dimensional quality-measure space, where the user gets to choose quality dimensions of interest. The collection of diverse and qualitative solutions is referred to as an *archive*, containing the best performing combination of measure-representatives, called *elites*, in each discretized cell. Additionally, the *QD-score* measures the quality and diversity of the archive by summing the objective values of elites in the archive. After a fixed amount of exploration, the set of elites is returned. Application of MAP-Elites can be found in, e.g., constrained optimization (Fioravanzo and Iacca 2019), robotic adaptability (Cully et al. 2015) or arbitrarily scalable environment generators (Zhang et al. 2024).

Covariance Matrix Adaptation From the literature of QD methods, we explore the methods of Covariance Matrix Adaptation (CMA) (Hansen 2016) as our circuit optimization algorithm. CMA, initially based on evolutionary strategies (ES), maintains a population of solution samples (generation) and moves each iteration toward the center of the highest objective evaluation.

CMA, in its simplest form **CMA-ES**, models the sampling distribution of the population as a multivariate normal distribution $\mathcal{N}(m, C)$, where m is the distribution mean and C is its covariance matrix. The main mechanisms steering CMA-ES are the selection and ranking of the μ fittest solutions, which update the next generation's next sampling distribution. A history of aggregate changes to m (the evolution path) provides information about the search process similar to momentum in stochastic gradient descent. To avoid the quick convergence of CMA-ES, the exploration aspects of the Map-Elites algorithm were integrated into the **CMA-ME** version (Fontaine et al. 2020) to create a population of modified CMA-ES instances called *emitters*, that each performs a search with feedback gained from interacting with the archive. The concept of emitters extends CMA-ES by adjusting the ranking rules that form the covariance matrix update to maximize the likelihood that future steps in a given direction result in archive improvements with respect to the quality measures.

Finally, CMA MAP-Annealing (**CMA-MAE**) (Fontaine et al. 2020; Fontaine and Nikolaidis 2023) has emerged with state-of-the-art performance in continuous domains. CMA-MAE extends MAP-Elites by incorporating the self-adaptation mechanisms of CMA-ES. CMA-ES maintains a Gaussian distribution, samples from it for new solutions, evaluates them, and then updates the distribution towards the high-objective region of the search space. Furthermore,

an *optimization archive* updating mechanism to balance exploitation and exploration of the measure space is maintained alongside the *result archive*. The mechanism introduces a threshold value t_e (independent of the objective measure) to each cell e in the archive, which determines whether a new solution θ' should be added. New solution θ' is then accepted iff. $f(\theta') > t_e$, where $f(\theta')$ is the objective evaluation of θ' . The threshold values are iteratively updated via an archive learning rate $\alpha \in [0; 1]$ (to infer the ‘improvement rate’ of the search at cell e), calculated as $\Delta = f(\theta') - t_e$. Upon acceptance in the respective cell, t_e is updated via $t_e \leftarrow (1 - \alpha)t_e + \alpha f(\theta')$, controlled by learning rate α .

In choosing α , we can adjust the CMA framework on a ‘scale’ from CMA-ES to CMA-MA: With $\alpha = 1$, CMA-MAE behaves like CMA-ME, with the improvement control greedily moving away from diminishing improving iterations. On the other hand, with $\alpha = 0$, CMA-MAE behaves like CMA-ES since improvement values always correspond to the objective values in the t_e update term. As the threshold will never change, CMA-MAE will only optimize the objective, akin to the evolutionary strategy CMA-ES. With α values between 0 and 1, CMA-MAE will gradually anneal the exploration through t_e and ‘linger’ around promising solutions, even if their improvement rate diminishes.

Method

Quantum Circuit Encoding

We therefore propose a compact circuit encoding for variational and non-variational gate-based quantum circuits. The value *and* type of each gate are left to an evolutionary algorithm. This means that rather than working with a fixed ansatz with variable parameters, the structure of the circuit is optimized concurrently with the parameters. The encoding of each gate $g_{i,l} \in [0; |\mathcal{GS}|)$, $0 \leq i < n$, $0 \leq l < L$ (i.e., the gate on wire i in layer l) is implemented as one individual mapping scalar, where we split the integer and the decimal parts of the floating-point value to realize the integer as the discrete choice of the gate-kind (mapped to a preselected, ordered gate-set \mathcal{GS} of length $|\mathcal{GS}|$). We then use the decimal of the scalar to complete the selected gate according to one of three options:

- 1 Gates with controls (e.g., *CNOT*) at wire i for gate g_i place the control-target at the qubit corresponding to the normalized segment of the decimal, where each segment has an equal width $\frac{1}{n}$ and the range for segment $j \in [0; n - 2)$ is defined as $[\frac{j}{n}, \frac{j+1}{n}) \in [0; 1]$, for all segments $[\text{segment}_0; \text{segment}_{n-2})$. In other words, the decimal selects the normalized segment (qubit), which corresponds to the normalized magnitude of the decimal $\in [0; 1]$ between qubits 0 to n . To safely include the boundary value 1, the last segment $i = n - 1$ is adjusted to $\text{segment}_{(n-1)} = [\frac{n-1}{n}, 1]$.

In the case that the corresponding target segment is the gate’s wire itself, the action choice becomes the uncontrolled version of the gate (e.g., $\text{CNOT}_{(i,j)} \rightarrow X_i$, $\iff i = j$).

- 2 Variational gates (e.g., $R_X(\beta)_i$) at wire i for gate g_i treat the decimal as the gate angle α . To avoid over-rotation, in the realization of the gates, we remap the decimal value to the interval $[0; 1] \rightarrow [0, 2\pi]$.
- 3 Fixed operator gates (e.g., Hadamard H) without angles or targets discard the decimal, i.e., only the gate choice for gate g_i remains.

The gate encoding of $n \times L$ scalars may then be flattened layer-by-layer (i.e., column-wise), to form our circuit encoding of dimensionality $n * L$, to be optimized by any search based optimization method. Since the only assumption made here is the preselection of a desirable gate set, this encoding is task agnostic and is generalizable to any ML/RL optimization or quantum control algorithm.

For the gate-selection we test the following gate-set \mathcal{GS} combinations:

$$\begin{aligned} \mathcal{GS}_{\text{CliffordT}} &:= \{\text{CNOT}, H, S, T, I\} \\ \mathcal{GS}_{\text{RotCNOT}} &:= \{R_X, R_Y, R_Z, \text{CNOT}, I\} \\ \mathcal{GS}_{\text{TinyH}} &:= \{R_X, H, \text{CNOT}, I\} \\ \mathcal{GS}_{\text{Tiny}} &:= \{R_X, \text{CNOT}, I\} \end{aligned}$$

where `CliffordT` is the universal quantum gate set (cf. Williams (2010)) and `RotCNOT`, `Tiny`, `TinyH` are practically applied reductions, i.e., commonly used VQC building-blocks (cf. Tilly et al. (2022)). All gate sets include the identity operator I to allow for potentially sparse layer designs.

The quantum circuit is then constructed by mapping each element of the vector to a gate in the quantum circuit. Element e_i of the vector maps to a gate at a fixed position within the circuit. In the $\mathcal{GS}_{\text{Tiny}}$ framework, a value of 3 or greater maps to an identity in the circuit at that position. If the value is $2 \leq e_i \leq 3$, the inserted gate is an R_X rotation gate with parameter $(e_i - 2) \cdot 2\pi$. In the last case, $1 \leq e_i < 2$, a *CNOT* gate is inserted, with the target qubit number uniformly mapped to the fractional part. The other gate sets act similarly, with differences in the density of changes in the fractional parts.

Quality Measures

Defining our QD measures, the objective function will be the combinatorial optimization hamiltonian cost objectives as described in the background section on CO. As our quality metrics, we choose *circuit expressivity* – heuristically approximated by *circuit sparsity* – and *gate diversity* to search for performant circuits potentially resistant against barren plateaus for the following reasons: Over-parameterization of gates, in particular with the noisy NISQ era QC hardware, has not only been shown to make VQCs sensitive to barren plateaus (Wang et al. 2021) but also can – depending on the ansatz design – hinder the circuit expressivity w.r.t. state reachability in the Hilbert space (Larocca et al. 2024). As such, we want to explore the sparsity of our circuit solutions in the range $[0; n * L]$, where 0 sparsity implies a fully parameterized qubit-to-layer mapping and $n * L$ sparsity represents a completely empty circuit.

Formally, the QD measure sparsity of solution θ is defined as the count of

$$\text{sparsity}(\theta) = \sum_{l=0}^L \sum_{i=0}^n \mathbb{I}[\theta_{l,i} \neq I],$$

where \mathbb{I} is the indicator function and I is the Identity gate (i.e., an empty position for wire i on layer l).

Similarly, the concept of unitary symmetry, i.e., of the unitary spanned by the realized state of the circuit in the Hilbert space, is thought to relate closely to classical simulability (Cerezo et al. 2023), where completely symmetric unitaries offer no quantum effects, and, hence, are classically computable. Therefore a certain degree of non-uniformity in the circuit is desirable, which we express formally as the QD measure *gate diversity* of solution θ , i.e.,

$$\text{gate diversity}(\theta) = \sum_{l=0}^L |\{\theta_l\} \setminus \{I\}|,$$

where $|\{\theta_l\}|$ is the size of the unique element set of all gates on layer l of solution θ (excluding Identity gates). This measure spans the range $[0; \min(|\mathcal{GS}| * L, n * L)]$, where a gate diversity of 0 implies no gates in the circuit, up to having every layer, respectively, consist of (all) different gates-types on each circuit wire. Naturally, at most n gates can differ on each layer l in the case that $|\mathcal{GS}| > n$.

Our circuit samples are then optimized via QD-CMA search, using the respective CO Hamiltonian as objective function, and the sparsity and gate diversity as quality measures. We test our approach with different learning rates α to employ variations of CMA-ES, CMA-ME or CMA-MAE, respectively, for the circuit search problem.¹

Baselines

Since there are numerous possibilities of iterating circuits, and testing all recent VQC proposals is out of scope for this paper (cf. Sim, Johnson, and Aspuru-Guzik (2019) for a selection), we instead conform to the following two well-known, respectively state-of-the-art methods for both constructing and optimizing quantum circuits: For a gradient-based optimizer we choose the analytic Quantum Approximate Optimization Algorithm (QAOA) ansatz as well know benchmark algorithm (Farhi, Goldstone, and Gutmann 2014). For a gradient-free quantum optimization method, there are many recent examples, (Sünkel et al. 2023; Lukac et al. 2003; Ding, Jin, and Yang 2008; Chen et al. 2020), however, we pick the quantum NEAT (QNEAT) (Giovagnoli et al. 2023) algorithm as a baseline. QNEAT is based on NEAT (Stanley and Miikkulainen 2002) for classical topological architecture search and has shown strong competitive performance compared to QAOA and comparable setups. For further details to both QAOA and QNEAT we refer the reader to their respective publications.

¹Implementation provided at github.com/m-zorn/qd4vqc

v=12	maxCLI	maxCUT	maxIND	minVER
cma-es	0.985	0.995	0.998	0.987
cma-mae	0.973	0.993	0.996	0.976
cma-me	0.984	0.997	0.998	0.979
v=14	maxCLI	maxCUT	maxIND	minVER
cma-es	0.972	0.987	0.998	0.955
cma-mae	0.965	0.980	0.995	0.942
cma-me	0.982	0.979	0.996	0.956
v=16	maxCLI	maxCUT	maxIND	minVER
cma-es	0.962	0.973	0.993	0.933
cma-mae	0.961	0.968	0.992	0.900
cma-me	0.964	0.976	0.991	0.912

Table 1: Scaling of different CMA-methods for graphs with $v = 12, 14, 16$ vertices (4 layers). Each cell represents the average approximation ratio of 50 randomly generated Erdos-Reny graph instances over 100 optimization steps.

Experiments

Gate-Set Choice We begin our experimental evaluation with an ablation on the suitability of QD-optimization for the four different gate sets introduced previously, `CliffordT`, `RotCNOT`, `TinyH` and `Tiny`. We test the three CMA variants, CMA-MAE, CMA-ES, and CMA-ME, on solutions with $L = 4$ circuit layers, 20 emitters (batch size 5 each). In this case study, we consider MaxCUT problem graphs, with 8 vertices, i.e., the solution circuits are of size $8 \times 4 = 32$. The three test graph types (*Barbell*, *Ladder*, *Caveman*) are considered in reference to (Khairy et al. 2020) as well as for direct comparability to (Giovagnoli et al. 2023). *Barbell*, *Ladder* and *Caveman* are characteristic graph instances in that they each represent distinct graph properties, i.e., complete graphs that are sparsely connected (*Barbell*, with sub-graphs of size p_B), uniformly structured graphs (*Ladder*, with p_L repetitions), and graphs consisting of cliques (*Caveman*, with c cliques of size $|c|$, i.e., $p_C = (c, |c|)$). For gate-set suitability, we show the optimization quality via the solution approximation ratio on these three representative graph instances.

The graphs and results are shown in Figure 1. We observe that the variational gate sets `RotCNOT`, `TinyH` and `Tiny` significantly outperform the `CliffordT` set, with `Tiny` emerging as the variational gate set with the most variance, but also the best optimization potential. This result can be expected since `CliffordT` consists of mainly fixed gate operators, i.e., without parameterization, and is effectively only using half the scalar encoding compared to the parameterized control that variational gate sets allow. We also note that simply removing the Hadamard H operator (from the `TinyH` set) seems to improve the CMA optimization process, indicating that small and basic operator sets like `Tiny` are favorable choices for the CMA methods.

QD-measure Exploration Considering three (singular but representative) optimization runs of the CMA-MAE method on the `TinyH` set in Figure 2, we can also observe which directions of the QD measures the CMA meth-

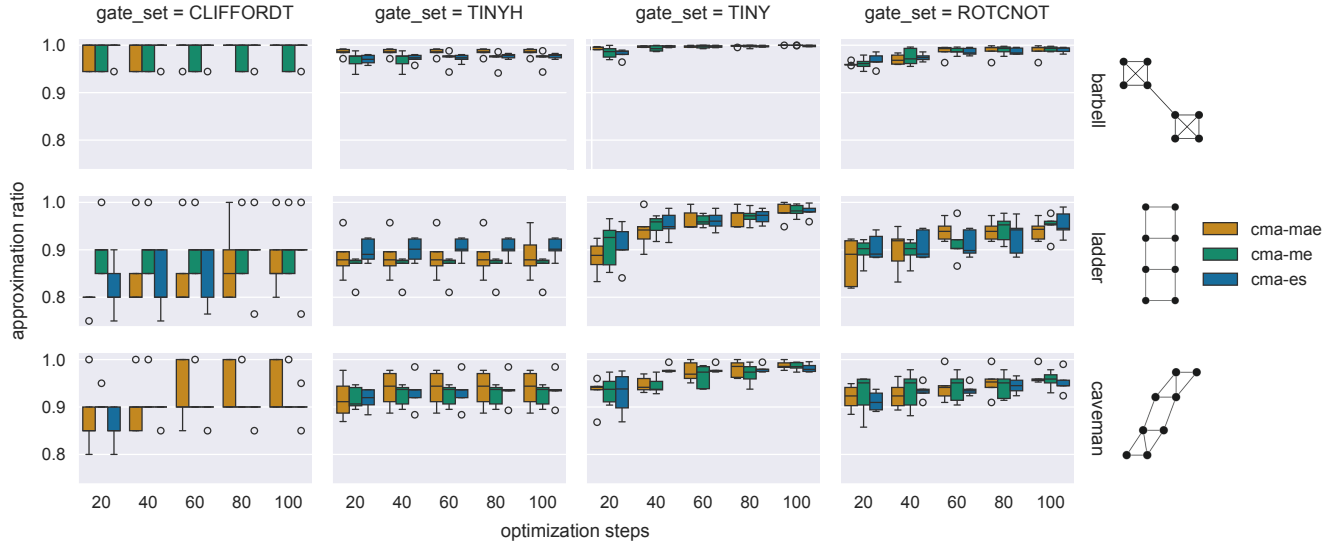


Figure 1: Ablation on the optimizability of different gate sets (from left to right: CliffordT, TinyH, Tiny, RotCNOT) on the three example graphs barbell (top), ladder (center), caveman (bottom) after 100 optimization steps (x-axis). The approximation ratio to the optimal objective energy is shown on the (y-axis). Boxplot error bars show the 95% confidence interval over 5 runs each.

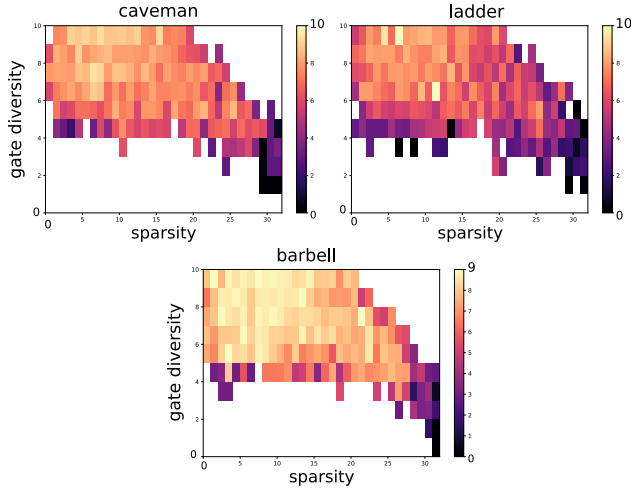
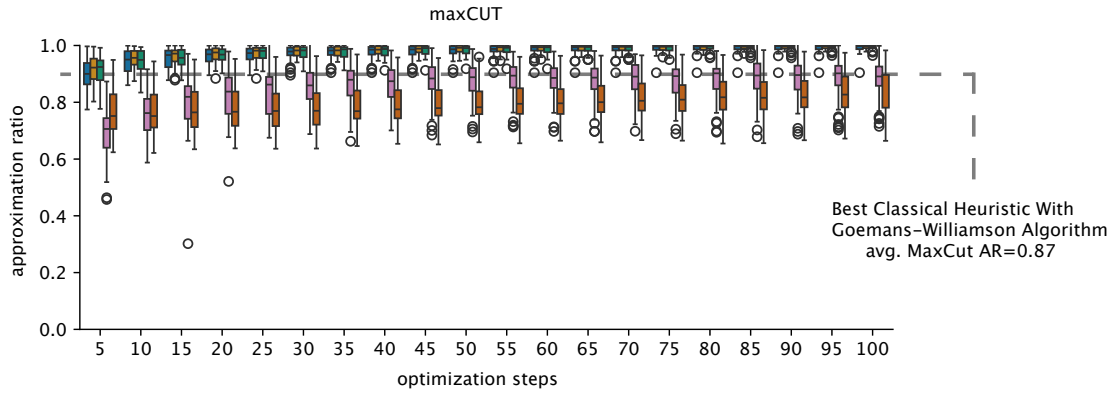


Figure 2: Single MaxCUT run heatmaps of the elite archives after 100 steps of CMA-MAE optimization on the example graphs (barbell, bottom), (caveman, top-left), (ladder, top-right). Lighter colors indicate higher objective function values for the quality criteria (sparsity, x-axis) and (gate diversity, y-axis). Axes start with (0,0) bottom left.

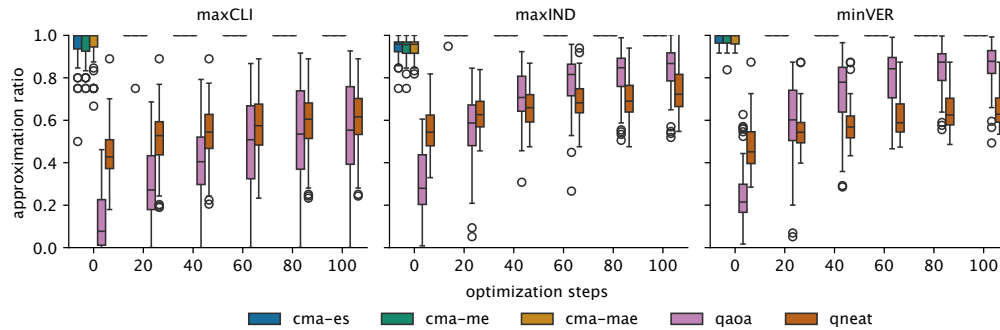
ods choose to explore and exploit. The heatmaps represent the result maps of the found elites (measure combinations), with color indicating their respective MaxCUT objective score. In all three cases we find that CMA-MAE tends to explore towards sparser circuits with a high degree of gate variability. We conclude that the CMA, as such, naturally explores circuits less prone to barren plateaus (avoiding

over-parameterization) but still leveraging quantum effects (with diverse layers, i.e., lowered symmetry). Heatmaps for CMA-ES and CMA-MA show similar trends, but not as pronounced as with CMA-MAE.

Baseline Evaluation Finally, we conduct a direct baseline comparison to QAOA and QNEAT. We compare all three CMA variants (CMA-MAE, CMA-ES, and CMA-ME) to QAOA with $p = 4$ layers, as well as QNEAT, with population sizes of 100 individuals and the setup detailed in (Giovagnoli et al. 2023). We show the average approximation ratio (best-solution objective value / optimal-solution objective value) of 50 randomly generated Erdos-Reny graph instances over 100 optimization steps in Figure 3, first in detail for MaxCUT in Figure 3a and as trend for the other three combinatorial optimization problems in Figure 3b. We note that all CMA variants clearly and quickly outperform both baselines. All baselines do eventually find optimal solution circuits; however, QAOA only slowly, and QNEAT is characterized by very high variance. In comparison, CMA-MAE, CMA-ES, and CMA-ME perform relatively similarly, with CMA-MAE again slightly more optimal. We also note that QD methods are vastly more sample efficient, often finding optimal (or almost optimal) solutions within 20 optimization steps. Efficiency on this scale can be considered quite valuable in both evaluation time and cost, considering that application on actual quantum hardware is still very expensive. A small scaling ablation of the CMA methods for problems graphs of $v = 12, 14, 16$ vertices can be found in Table 1 for comparison. While a slight degradation in approximation-ratio with increased graph-size is noticeable, the general performance remains strong across the three CMA methods.



(a) Evaluation MaxCUT problem



(b) Evaluation MaxCLI, maxIND, minVer problems

Figure 3: Comparison of the optimization progress for the different combinatorial optimization problems, each on of 50 randomly generated erdos-ryen graphs (connected), with 8 vertices (average edge-density 0.635 ± 0.22). Optimization steps over time are shown per graph, QNEAT with a population of 100 individuals, QAOA with $p = 4$, CMA-MAE ($\alpha = 0.3$), CMA-ES ($\alpha = 0$), CMA-ME ($\alpha = 1$) producing 4 layer VQCs using the TINY gate set. Boxplots show optimization over 100 steps, at every n -th step (x-axis), and the approximation ratios (y-axis) of the best solution energies found until this step / respective optimal solution energies. Error bars show the 95% confidence interval. **(Top:)** Comparison of the optimization progress for the MaxCUT problem in greater detail (every $n = 5$ optimization steps), **(Bottom:)** shows the MaxCLI, maxIND, minVer problems at every $n = 20$ -th step.

Hardware Details

All experiments were simulated on two identical Linux (Debian 6.1) workstations, with AMD Threadripper 5995WX CPUs and NVIDIA 4090 GPUs with 1024gb RAM.

Limitations and Future Work

In this work, we have proposed and detailed a compact but expressive circuit encoding and shown that quality-diversity driven optimization with covariance matrix adaptation methods notably outperforms current state-of-the-art optimizers on a set of combinatorial optimization benchmark problems. We find that by appropriate design of our quality metrics (circuit sparsity and gate variance), QD methods will naturally explore and find circuit solutions that prevent the current NISQ era problems, such as barren plateaus and insufficient quantum effect through overly symmetric circuits. However, our work is still limited in scope, yielding many directions for future work. More extensive evaluation

against QML-based methods on a variety of other benchmarks would round out this study with even more significance. Secondly, the choice of quality metrics could also be further explored, with concepts like *circuit expressivity* and *circuit capacity*, which are also linked to barren plateaus (Larocca et al. 2024).

Acknowledgments

We thank the anonymous reviewers for their valuable comments. This paper was partially funded by the German Federal Ministry for Economic Affairs and Climate Action through the funding program "Quantum Computing – Applications for the industry" (Project QCHALLENGE, contract number: 01MQ22008A) and through the Munich Quantum Valley, supported by the Bavarian state government's Hightech Agenda Bayern Plus.

References

- An, Z.; and Zhou, D. 2019. Deep reinforcement learning for quantum gate control. *EPL (Europhysics Letters)*, 126(6): 60002.
- Beer, K.; Bondarenko, D.; Farrelly, T.; Osborne, T. J.; Salzmann, R.; Scheiermann, D.; and Wolf, R. 2020. Training deep quantum neural networks. *Nature communications*, 11(1): 1–6.
- Biamonte, J.; Wittek, P.; Pancotti, N.; Rebentrost, P.; Wiebe, N.; and Lloyd, S. 2017. Quantum machine learning. *Nature*, 549(7671): 195–202.
- Bravo-Prieto, C.; LaRose, R.; Cerezo, M.; Subasi, Y.; Cincio, L.; and Coles, P. J. 2019. Variational quantum linear solver. *arXiv preprint arXiv:1909.05820*.
- Cao, Y.; Romero, J.; Olson, J. P.; Degroote, M.; Johnson, P. D.; Kieferová, M.; Kivlichan, I. D.; Menke, T.; Peropadre, B.; Sawaya, N. P.; et al. 2019. Quantum chemistry in the age of quantum computing. *Chemical reviews*, 119(19): 10856–10915.
- Cerezo, M.; Arrasmith, A.; Babbush, R.; Benjamin, S. C.; Endo, S.; Fujii, K.; McClean, J. R.; Mitarai, K.; Yuan, X.; Cincio, L.; et al. 2021. Variational quantum algorithms. *Nature Reviews Physics*, 3(9): 625–644.
- Cerezo, M.; Larocca, M.; García-Martín, D.; Diaz, N.; Braccia, P.; Fontana, E.; Rudolph, M. S.; Bermejo, P.; Ijaz, A.; Thanasilp, S.; et al. 2023. Does provable absence of barren plateaus imply classical simulability? Or, why we need to rethink variational quantum computing. *arXiv preprint arXiv:2312.09121*.
- Chen, S. Y.-C.; Yang, C.-H. H.; Qi, J.; Chen, P.-Y.; Ma, X.; and Goan, H.-S. 2020. Variational quantum circuits for deep reinforcement learning. *IEEE Access*, 8: 141007–141024.
- Conti, E.; Madhavan, V.; Petroski Such, F.; Lehman, J.; Stanley, K.; and Clune, J. 2018. Improving exploration in evolution strategies for deep reinforcement learning via a population of novelty-seeking agents. *Advances in neural information processing systems*, 31.
- Cully, A.; Clune, J.; Tarapore, D.; and Mouret, J.-B. 2015. Robots that can adapt like animals. *Nature*, 521(7553): 503–507.
- Ding, S.; Jin, Z.; and Yang, Q. 2008. Evolving quantum circuits at the gate level with a hybrid quantum-inspired evolutionary algorithm. *Soft Computing*, 12: 1059–1072.
- Dong, D.; Chen, C.; Li, H.; and Tarn, T.-J. 2008. Quantum reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(5): 1207–1220.
- Dunjko, V.; Taylor, J. M.; and Briegel, H. J. 2017. Advances in quantum reinforcement learning. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 282–287. IEEE.
- Egger, D. J.; Gambella, C.; Marecek, J.; McFaddin, S.; Mevissen, M.; Raymond, R.; Simonetto, A.; Woerner, S.; and Yndurain, E. 2020. Quantum computing for finance: State-of-the-art and future prospects. *IEEE Transactions on Quantum Engineering*, 1: 1–24.
- Farhi, E.; Goldstone, J.; and Gutmann, S. 2014. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*.
- Fioravanzo, S.; and Iacca, G. 2019. Evaluating MAP-Elites on constrained optimization problems. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion, GECCO '19*, 253–254. New York, NY, USA: Association for Computing Machinery. ISBN 9781450367486.
- Fontaine, M.; and Nikolaidis, S. 2021. Differentiable quality diversity. *Advances in Neural Information Processing Systems*, 34: 10040–10052.
- Fontaine, M.; and Nikolaidis, S. 2023. Covariance matrix adaptation map-annealing. In *Proceedings of the Genetic and Evolutionary Computation Conference*, 456–465.
- Fontaine, M. C.; Togelius, J.; Nikolaidis, S.; and Hoover, A. K. 2020. Covariance matrix adaptation for the rapid illumination of behavior space. In *Proceedings of the 2020 genetic and evolutionary computation conference*, 94–102.
- Fösel, T.; Niu, M. Y.; Marquardt, F.; and Li, L. 2021. Quantum circuit optimization with deep reinforcement learning. *arXiv preprint arXiv:2103.07585*.
- Gabor, T.; Zorn, M.; and Linnhoff-Popien, C. 2022. The applicability of reinforcement learning for the automatic generation of state preparation circuits. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 2196–2204.
- Gaier, A.; Asteroth, A.; and Mouret, J.-B. 2018. Data-efficient design exploration through surrogate-assisted illumination. *Evolutionary computation*, 26(3): 381–410.
- Giovagnoli, A.; Tresp, V.; Ma, Y.; and Schubert, M. 2023. Qneat: Natural evolution of variational quantum circuit architecture. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation*, 647–650.
- Hansen, N. 2016. The CMA evolution strategy: A tutorial. *arXiv preprint arXiv:1604.00772*.
- Kandala, A.; Mezzacapo, A.; Temme, K.; Takita, M.; Brink, M.; Chow, J. M.; and Gambetta, J. M. 2017. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *nature*, 549(7671): 242–246.
- Khairy, S.; Shaydulin, R.; Cincio, L.; Alexeev, Y.; and Balaprakash, P. 2020. Learning to optimize variational quantum circuits to solve combinatorial problems. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 2367–2375.
- Kölle, M.; Schubert, T.; Altmann, P.; Zorn, M.; Stein, J.; and Linnhoff-Popien, C. 2024. A Reinforcement Learning Environment for Directed Quantum Circuit Synthesis. *arXiv preprint arXiv:2401.07054*.
- Kyriienko, O.; and Magnusson, E. B. 2022. Unsupervised quantum machine learning for fraud detection. *arXiv preprint arXiv:2208.01203*.
- Larocca, M.; Thanasilp, S.; Wang, S.; Sharma, K.; Biamonte, J.; Coles, P. J.; Cincio, L.; McClean, J. R.; Holmes, Z.; and Cerezo, M. 2024. A Review of Barren Plateaus in Variational Quantum Computing. *arXiv preprint arXiv:2405.00781*.

- Lehman, J.; and Stanley, K. O. 2011. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary computation*, 19(2): 189–223.
- Lehman, J.; Stanley, K. O.; et al. 2008. Exploiting open-endedness to solve problems through the search for novelty. In *ALIFE*, 329–336.
- Li, Y.; Tian, M.; Liu, G.; Peng, C.; and Jiao, L. 2020. Quantum optimization and quantum learning: A survey. *Ieee Access*, 8: 23568–23593.
- Lukac, M.; Perkowski, M.; Goi, H.; Pivtoraiko, M.; Yu, C. H.; Chung, K.; Jeech, H.; Kim, B.-G.; and Kim, Y.-D. 2003. Evolutionary approach to quantum and reversible circuits synthesis. *Artificial Intelligence Review*, 20: 361–417.
- Mackeprang, J.; Dasari, D. B. R.; and Wrachtrup, J. 2020. A reinforcement learning approach for quantum state engineering. *Quantum Machine Intelligence*, 2(1): 1–14.
- Mirkis, V.; and Domshlak, C. 2013. Abstractions for over-subscription planning. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 23, 153–161.
- Mouret, J.-B.; and Clune, J. 2015. Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909*.
- Nielsen, M. A.; and Chuang, I. L. 2010. *Quantum computation and quantum information*. Cambridge university press.
- Ostaszewski, M.; Trenkwalder, L.; Masarczyk, W.; Scerri, E.; and Dunjko, V. 2021. Reinforcement learning for optimization of variational quantum circuit architectures. *Advances in Neural Information Processing Systems*, 34.
- Preskill, J. 2018. Quantum computing in the NISQ era and beyond. *Quantum*, 2: 79.
- Robinson, N.; Gretton, C.; Pham, D. N.; and Sattar, A. 2009. SAT-based parallel planning using a split representation of actions. In *Proceedings of the international conference on automated planning and scheduling*, volume 19, 281–288.
- Rønnow, T. F.; Wang, Z.; Job, J.; Boixo, S.; Isakov, S. V.; Wecker, D.; Martinis, J. M.; Lidar, D. A.; and Troyer, M. 2014. Defining and detecting quantum speedup. *science*, 345(6195): 420–424.
- Senokosov, A.; Sedykh, A.; Sagingalieva, A.; Kyriacou, B.; and Melnikov, A. 2023. Quantum machine learning for image classification. *Machine Learning: Science and Technology*.
- Sim, S.; Johnson, P. D.; and Aspuru-Guzik, A. 2019. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Advanced Quantum Technologies*, 2(12): 1900070.
- Stanley, K. O.; and Miikkulainen, R. 2002. Evolving neural networks through augmenting topologies. *Evolutionary computation*, 10(2): 99–127.
- Sünkel, L.; Martyniuk, D.; Mattern, D.; Jung, J.; and Paschke, A. 2023. GA4QCO: genetic algorithm for quantum circuit optimization. *arXiv preprint arXiv:2302.01303*.
- Tilly, J.; Chen, H.; Cao, S.; Picozzi, D.; Setia, K.; Li, Y.; Grant, E.; Wossnig, L.; Rungger, I.; Booth, G. H.; et al. 2022. The variational quantum eigensolver: a review of methods and best practices. *Physics Reports*, 986: 1–128.
- Wang, S.; Fontana, E.; Cerezo, M.; Sharma, K.; Sone, A.; Cincio, L.; and Coles, P. J. 2021. Noise-induced barren plateaus in variational quantum algorithms. *Nature communications*, 12(1): 6961.
- Williams, C. P. 2010. *Explorations in quantum computing*. Springer Science & Business Media.
- Wu, S.; Jin, S.; Wen, D.; Han, D.; and Wang, X. 2020. Quantum reinforcement learning in continuous action space. *arXiv preprint arXiv:2012.10711*.
- Zhang, H.; Salzman, O.; Kumar, T. S.; Felner, A.; Ulloa, C. H.; and Koenig, S. 2022. A* pex: Efficient approximate multi-objective search on graphs. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 32, 394–403.
- Zhang, Y.; Fontaine, M.; Bhatt, V.; Nikolaidis, S.; and Li, J. 2024. Arbitrarily Scalable Environment Generators via Neural Cellular Automata. *Advances in Neural Information Processing Systems*, 36.