

Variational Quantum Multi-Objective Optimization

Linus Ekström,¹ Hao Wang,² and Sebastian Schmitt¹

¹*Honda Research Institute Europe GmbH, Carl-Legien-Str. 30, 63073 Offenbach, Germany*

²*LIACS/aQa, Leiden University, Niels Bohrweg 1, 2333 CA Leiden, Netherlands*

(Dated: February 5, 2024)

Solving combinatorial optimization problems on near-term quantum devices has gained a lot of attraction in recent years. Currently, most works have focused on single-objective problems, whereas many real-world applications need to consider multiple, mostly conflicting objectives, such as cost and quality. We present a variational quantum optimization algorithm to solve discrete multi-objective optimization problems on quantum computers. The proposed quantum multi-objective optimization (QMOO) algorithm incorporates all cost Hamiltonians representing the classical objective functions in the quantum circuit and produces a quantum state consisting of Pareto-optimal solutions in superposition. From this state we retrieve a set of solutions and utilize the widely applied hypervolume indicator to determine its quality as an approximation to the Pareto-front. The variational parameters of the QMOO circuit are tuned by maximizing the hypervolume indicator in a quantum-classical hybrid fashion. We show the effectiveness of the proposed algorithm on several benchmark problems with up to five objectives. We investigate the influence of the classical optimizer, the circuit depth and compare to results from classical optimization algorithms. We find that the algorithm is robust to shot noise and produces good results with as low as 128 measurement shots in each iteration. These promising result open the perspective to run the algorithm on near-term quantum hardware.

I. INTRODUCTION

Over the past few years, quantum computing experiments have indicated that the threshold of practically useful quantum computation is nearing [1–4]. Realistic application problems have increasingly moved into the focus of quantum algorithm research, in particular for the currently available noisy-intermediate scale quantum (NISQ) devices [5–10]. A promising class of approaches are the variational algorithms such as the quantum approximate optimization algorithm (QAOA) [11, 12]. However, the exact nature of the potential quantum advantage for real-world combinatorial optimization problems remains an open problem [12–14].

Current versions of variational quantum optimization approaches only deal with single-objective optimization problems. The final target for the optimized quantum circuit is to produce a quantum state where the probability of sampling the optimal solution of the original problem is as large as possible. However, many realistic application problems need to consider more than one objective, such as cost and quality, simultaneously [15, 16]. Since these objectives are mostly in a trade-off relationship where increasing the performance in one objective leads to a degradation in the other objectives, finding one single optimal solution is no longer a proper target setting. Instead, the goal is to find a set of Pareto-optimal solutions that represent the best possible trade-off solutions. Solving such multi-objective optimization problems directly with a variational quantum algorithm has not been possible until now.

While most investigations focus on qubit quantum computation, qubits can lead to significant encoding overhead for integer optimization problems [7, 17, 18]. In the NISQ-era of quantum computation, such over-

heads may be prohibitive to obtaining useful results. It has been suggested some of these limitations can be handled effectively by utilizing multi-level systems called qudits. There are indications that qudit systems may provide algorithmic benefits for circuit simplification [19–21], error correction [22, 23] or algorithm speed-up [24]. Qudit-based approaches have been used for several applications like max-cut, graph-coloring, constraints handling or machine learning problems [6, 25–32]. Generally, qudits represent a very promising route to efficient quantum information processing in enlarged Hilbert spaces. [21, 33–36]. Therefore, we formulate and evaluate the proposed quantum multi-objective optimization algorithm with qudit variables.

The structure of the paper is as follows. After a brief introduction to multi-objective optimization, we repeat the basics of qudit-based formulation of quantum algorithms. Then, we describe the proposed quantum multi-objective (QMOO) algorithm and introduce several benchmark functions for evaluating its performance. We present empirical results from algorithm simulations. Finally, we discuss the results, present our conclusions from this work, and give an outlook for future directions of investigations.

II. RELATED WORK

A recent proposal for solving multi-objective optimization problems with NISQ devices [37] utilizes a scalarizing approach. The objectives are combined into several single objectives by weighted sums. Each scalarized objective is then solved using a regular QAOA to obtain one Pareto-optimal solution. Combining the solutions of several different scalarizing functions allows the recon-

struction of convex parts of the Pareto-front.

Another recent approach uses the non-dominated sorting genetic algorithm II (NSGA-II) to obtain the solutions to a constraint single-objective optimization problem [38]. The quantum circuit generates one quantum state from which the classical cost function to be minimized is calculated. The fraction of constraint-fulfilling solutions in the quantum state is also calculated as a second objective. The multi-objective NSGA-II approach is only employed to handle the constraints. Finding a single optimized minimal cost solution is targeted. Another approach [39] utilizes the NSGA-II algorithm to find the best trade-offs between circuit complexity (measured in the number of CNOT gates) and achievable minimal energy solution for each structure. In this setup, a full variational single-objective optimization of the quantum circuit is done for a given circuit structure. In contrast, in our approach, all different objectives enter the quantum circuit, and the whole set of Pareto-optimal solutions is extracted from the resulting one quantum state at once.

III. MULTI-OBJECTIVE OPTIMIZATION

A multi-objective optimization problem (MOP) involves minimizing multiple cost functions $C_k(\mathbf{x})$ simultaneously, i.e., $\vec{C} = (C_1, \dots, C_K)$ with $C_k : \mathbf{x} \in \mathcal{X} \rightarrow y_k \in \mathbb{R}$, $k \in \{1, \dots, K\}$. For every $\vec{y}^{(1)} \in \mathbb{R}^K$, we say $\vec{y}^{(1)}$ weakly dominates $\vec{y}^{(2)}$ (denoted by $\vec{y}^{(1)} \preceq \vec{y}^{(2)}$) iff. $y_k^{(1)} \leq y_k^{(2)}$, $\forall k \in \{1, \dots, K\}$. The Pareto order \prec on \mathbb{R}^K is defined: $\mathbf{y}^{(1)} \prec \mathbf{y}^{(2)}$ iff. $\vec{y}^{(1)} \preceq \vec{y}^{(2)}$ and $\vec{y}^{(1)} \neq \vec{y}^{(2)}$. A point $\mathbf{x} \in \mathcal{X}$ is (Pareto-)efficient iff. $\nexists \mathbf{x}' \in \mathcal{X} : (\vec{C}(\mathbf{x}') \prec \vec{C}(\mathbf{x}))$. The set of all efficient points in \mathcal{X} is called the *efficient set* or *Pareto set* denoted by \mathcal{PS} . The image of the efficient set under \vec{C} is called the *Pareto front*.

Arguably, the most important performance metric in multi-objective optimization is the well-known hypervolume (HV) indicator [40–44]. HV of a set $Y \subseteq \mathbb{R}^K$ is defined as the K -dimensional Lebesgue measure λ of the subset dominated by Y and bounded above by a reference point $\vec{r} \in \mathbb{R}^K$, i.e., $\text{HV}(Y, \vec{r}) = \lambda(\{\vec{y} \in \mathbb{R}^K : \vec{y} \prec \vec{r} \wedge \exists \vec{p} \in Y, \vec{p} \prec \vec{y}\})$. The HV indicator is Pareto-compliant [40, 45–47]: for two subsets A and $B \subseteq \mathbb{R}^K$, $A \prec B \implies \text{HV}(A, \vec{r}) > \text{HV}(B, \vec{r})$, given the same reference point \vec{r} . Due to this property, the HV indicator is often employed as the performance indicator of multi-objective optimization algorithms. For continuous problem domains, i.e., $\mathcal{X} \subseteq \mathbb{R}^N$, maximizing the HV indicator leads to an asymptotic distribution of points on the Pareto front [48], which depends on the curvature thereof. To our knowledge, no theoretical results have been reported for discrete problems. In this work, we deal with unconstrained combinatorial integer optimization problems where $\mathcal{X} \subseteq \{0, \dots, d-1\}^N$.

IV. QUDIT REPRESENTATION

In this section, we briefly explain qudit quantum computing and motivate its usefulness in integer optimization settings. In integer combinatorial optimization, the goal is to find a specific search variable vector that yields points on the Pareto front of the optimization problem. The search variable $\mathbf{x} \in \mathcal{X}$ is a vector of N finite-ranged integers: $\mathbf{x} = (x_1, \dots, x_N)^\top$, where each variable takes a value $x_n \in \{0, 1, \dots, d-1\}$.

In the quantum formulation, these integer variables are represented by a set of N qudits, i.e., d -level quantum systems. The computational basis is chosen such that each possible search vector corresponds to one N qudit quantum state,

$$\mathbf{x} = (x_1, \dots, x_N)^T \Leftrightarrow |\mathbf{x}\rangle = |x_1, \dots, x_N\rangle \quad (1)$$

The total dimension of the Hilbert space is $\dim \mathcal{H} = d^N$, and any state vector $|\psi\rangle \in \mathcal{H}$ can be written as

$$|\psi\rangle = \sum_{x_1=0}^{d-1} \dots \sum_{x_N=0}^{d-1} \alpha_{x_1 \dots x_N} |\mathbf{x}\rangle, \quad (2)$$

where $\alpha_{x_1 \dots x_N}$ is the complex amplitude with $\sum_{\mathbf{x}} |\alpha_{\mathbf{x}}|^2 = 1$ and the states $|\mathbf{x}\rangle = |x_1, \dots, x_N\rangle$ form an orthonormal basis, i.e., $\langle \mathbf{x} | \mathbf{x}' \rangle = \delta_{\mathbf{x}, \mathbf{x}'} = \delta_{x_1, x'_1} \dots \delta_{x_N, x'_N}$.

In general, one needs $d^2 - 1$ Hermitian basis operators to generate all unitary operations of the $SU(d)$ group, which represent all allowed actions on a single qudit in the quantum circuit [19]. However, it was shown [49, 50] that it is sufficient to consider only the three operators L_x , L_z , and L_z^2 as generators. Here, L_x and L_z are the x - and z -angular momentum operators, which act on the single qudit state as follows,

$$L_x |x\rangle = \frac{1}{2} (\gamma_{d,x+1} |x+1\rangle + \gamma_{d,x-1} |x-1\rangle) \quad (3)$$

$$L_z |x\rangle = \frac{2x-d+1}{2} |x\rangle \quad (4)$$

where $x \in \{0, \dots, d-1\}$ and $\gamma_{d,x} = \sqrt{(d-x-1)(x+1)}$. The operator L_z^2 is called squeezing or one-axis twisting operator in the context of cold-atom systems. These three generators are sufficient to implement arbitrary unitary operations by (possibly many) repeated finite rotations. This is because the iterated commutators of these three operators generate all $d^2 - 1$ Hermitian basis operators, which are then necessary to generate all unitary operations of the group $SU(d)$.

V. QUANTUM MULTI-OBJECTIVE OPTIMIZATION

In this section, we explain the construction of our algorithm to solve multi-objective combinatorial optimization problems natively on quantum hardware.

Each objective $C_k(\mathbf{x})$ is represented by a cost Hamiltonian H_k , which is diagonal in the computational basis

$|\mathbf{x}\rangle$. The eigenvalue of the basis state is given by the cost function value of that state,

$$H_k |\mathbf{x}\rangle = C_k(\mathbf{x}) |\mathbf{x}\rangle \quad (k = 1, \dots, K). \quad (5)$$

We employ a variational quantum circuit to solve many-objective optimization problems. A schematic can be seen in Fig. 1. Analogous to the traditional QAOA, the circuit is built up from L layers, where the functional form of the unitary operation in each layer is the same across all layers, but each layer has independently adjustable variational parameters β^l and γ^l with $l = 1, \dots, L$,

$$|\psi(\beta, \gamma)\rangle = U(\beta^L, \gamma^L) \cdots U(\beta^1, \gamma^1) |\psi_0\rangle. \quad (6)$$

$|\psi_0\rangle$ denotes a predefined fixed initial state. We choose the uniform superposition of all possible states,

$$|\psi_0\rangle = \frac{1}{\sqrt{d^N}} \sum_{\mathbf{x} \in \{0, \dots, d-1\}^N} |\mathbf{x}\rangle, \quad (7)$$

which is an eigenvector to the generalized Pauli X -operator [19]. Alternative initial states are also possible, such as one single basis state or an eigenstate to the x angular momentum operator L_x . The latter is typically done for qubit circuits. However, the specific choice of the initial state should not change the algorithm's performance qualitatively: the different initial states are related by unitary transformations, which can be generated implicitly by the variation quantum circuit. Therefore, the initial states merely alter the variational optimization problem slightly without changing its characteristics. We experimentally validated this argument with various initial states (not shown in this paper).

Each layer l of the circuit comprises K blocks, where K is the number of classical objectives. Each block consists of one mixing unitary U_M and one phase unitary P_k ,

$$U(\beta^l, \gamma^l) = U_M(\beta_K^l) P_K(\gamma_K^l) \cdots U_M(\beta_1^l) P_1(\gamma_1^l) \quad (8)$$

This is in contrast to the regular QAOA, where only one phase and one mixing operator are included, i.e. $K = 1$.

While, in principle, the mixing unitaries of each block in Eq. (8) could have a different structure, we choose them to be the same but allow for different variational parameters of each mixer. We focus on mixing unitaries which are generated by the x -angular momentum L_x and the local squeezing operator, L_z^2 , which is sufficient to generate all unitaries as described in Sec. IV above. This leads to the mixing unitary of block k

$$U_M(\beta_k^l) = e^{-i\beta_{1,k}^l \sum_{j=1}^N L_{x;j} - i\beta_{2,k}^l \sum_{j=1}^N L_{z;j}^2}, \quad (9)$$

where the sum in the exponent extends over all qudits $j = \{1, \dots, \dots, N\}$. Note that for qubits ($d = 2$), the local squeezing operator reduces to the identity, $L_z^2 = \frac{1}{4}\mathbb{1}$, and consequently, we drop that term for $d = 2$.

The phase operators are generated by the K cost Hamiltonians representing the classical objectives, i.e.,

$$P_k(\gamma_k^l) = e^{-i\gamma_k^l H_k}. \quad (10)$$

where $k = \{1, \dots, K\}$.

For a given set of variational parameters (β, γ) the quantum circuit generates a state as depicted in Eq. (6). Generally, this state is a superposition of all computational basis states, i.e., all classical solutions, with varying amplitudes. A set of solutions is generated by running the quantum circuit N_{shots} times and measuring one output state each time. From this set, the N_S most frequent basis states are selected

$$|\psi(\beta, \gamma)\rangle \xrightarrow{N_{\text{shots}}} \mathcal{S}_{\beta, \gamma} = \{\mathbf{x}_i\}_{i=1}^{N_S}. \quad (11)$$

In case of infinite number of shots, $N_{\text{shots}} \rightarrow \infty$, this is analogous to performing a full state tomography (see, e.g., [51–53]) and selecting the N_S states with largest spectral weights $|\xi_{\mathbf{x}}|^2$. But as we show, the algorithm works well with a finite and small number of shots, much less than the dimension of the Hilbert space.

For each of those extracted classical solutions, all classical cost objectives are calculated, which generates the set of objective vectors:

$$\mathcal{C}_{\beta, \gamma} = \{\{C_k(\mathbf{x}_i)\}_{k=1}^K\}_{i=1}^{N_S} \quad (12)$$

$$= \{\vec{C}(\mathbf{x}_i)\}_{i=1}^{N_S}. \quad (13)$$

From this set, we extract the non-dominated solutions, which approximate the Pareto front. The quality of the current approximate solution set is then evaluated by calculating the hypervolume indicator with respect to a pre-defined reference point \vec{r}

$$\text{HV}\left(\{\vec{C}(\mathbf{x}_i)\}_{i=1}^{N_S}; \vec{r}\right). \quad (14)$$

This serves as the single classical objective function, which is maximized to determine the optimal values of the parameters (β, γ) .

The idea behind this choice of the unitaries in each layer is that each phase operator produces relative phases proportional to the energies of the corresponding objective. In that way, the information of each objective is encoded in relative phases of states, contributing to the quantum state prepared by the circuit. Given the individual variational parameters for each objective, the classical optimization procedure can, therefore, in principle, discriminate between states based on the set of objective values and find optimal parameters to increase the weights of Pareto-efficient solutions in the quantum state.

The ideal quantum state of the proposed multi-objective approach is given by a superposition where Pareto optimal solutions are the predominant contribution. That is, the ideal state represents the whole set of classical solutions belonging to the Pareto set \mathcal{PS} as one superposition, i.e.

$$|\psi(\beta_{\text{ide}}, \gamma_{\text{opt}})\rangle_{\text{ideal}} = \sum_{\mathbf{x} \in \mathcal{PS}} \xi_{\mathbf{x}} |\mathbf{x}\rangle + |\delta\psi\rangle, \quad (15)$$

with $\sum_{\mathbf{x} \in \mathcal{PS}} |\xi_{\mathbf{x}}|^2 \approx 1$ and $\langle \delta\psi | \delta\psi \rangle \ll 1$. This represents a major difference to current variational algorithms like

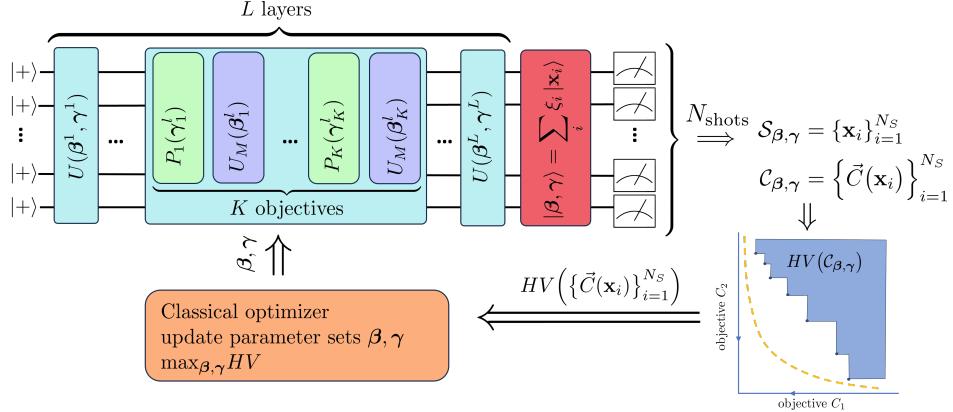


FIG. 1. Schematic overview of the QMOO algorithm. The algorithm is initialized in an even superposition state. The circuit consists of L layers (blue), each of which consists of phase- (green) and mixing-operators (purple) for each of the K objectives. The QMOO state (red) is measured N_{shots} times, and a population containing N_S solutions is generated from the output. We calculate the hypervolume from this population, which is used as a metric for the classical optimizer tuning the parameter sets β, γ . The updated parameters are used in the next iteration's QMOO circuit, and the quantum-classical loop is completed.

QAOA, which target reducing the quantum state to one optimal solution, i.e., one basis state.

The total number of variational parameters of the proposed algorithm is $N_{\text{params}} = 3LK$, where L is the number of layers and K is the number of objectives. For the case of qubits ($d = 2$) without a local squeezing operator, the number of parameters reduces to $N_{\text{params}} = 2LK$. In both cases, the number of parameters is increased by a factor of K compared to the standard QAOA.

VI. BENCHMARK FUNCTIONS AND EXPERIMENTAL SETUP

A. Benchmark Functions

The performance of the proposed multi-objective quantum optimization algorithm is evaluated on several benchmark functions. We combine linear and quadratic cost functions into two-, three-, and five-objective problems where all objectives are subject to minimization. To define a useful multi-objective benchmark, the objectives should be in a trade-off relation, where improving on one objective typically leads to performance degradation in other objectives. The benchmark functions are available as Python functions from a [github](#) repository [54].

The first class of benchmark problems is defined by a linear two-objective function of the form

$$\vec{C}_I(\mathbf{x}) = \begin{pmatrix} \mathbf{c}_1 \cdot \mathbf{x} \\ \mathbf{c}_2 \cdot \mathbf{x} \end{pmatrix}. \quad (16)$$

Here, $\mathbf{x} \in \{0, \dots, d-1\}^N$ and the cost coefficients \mathbf{c}_1 are sampled from $[-1, 1]^N$ uniformly at random, i.e., $\mathbf{c}_1 \sim \mathcal{U}([-1, 1]^N)$. The trade-off between the objectives is generated by choosing the second cost coefficients to have substantial anti-correlations with the first coefficients, i.e., $\mathbf{c}_2 \sim -\frac{1}{2}\mathbf{c}_1 + \frac{1}{2}\mathcal{U}([-1, 1]^N)$.

We also composed two-, three-, and five-objective benchmark problem classes using randomized quadratic cost functions of the form

$$C(\mathbf{x}) = \mathbf{x}^\top \mathbf{J} \mathbf{x} + \mathbf{h}^\top \mathbf{x}, \quad (17)$$

where $\mathbf{J} = \mathbf{J}^\top \in \mathbb{R}^{N \times N}$ are symmetric randomized coupling matrices and $\mathbf{h} \in \mathbb{R}^N$ are randomized local fields. To generate trade-offs between the objectives, we are guided by physical spin models that realize such cost Hamiltonians.

For one two-objective problem class ($K = 2$), we chose one objective to favor ferromagnetic (FM) alignment of the variables and the other to favor anti-ferromagnetic (AFM) alignment. Consequently, the symmetric coupling matrix elements are chosen from uniform distributions, i.e., $J_{ij} = J_{ji} \sim \mathcal{U}(a, b)$ where for the FM case all matrix elements are negative, i.e., $a < b < 0$, and for the AFM case all are positive, i.e., $b > a > 0$. For the cost function to truly represent spin models, the local fields are chosen as $\mathbf{h} = \mathbf{g} - d\mathbf{1}^\top \cdot \mathbf{J}$ where \mathbf{g} is a random vector with $g_i \sim \mathcal{U}(-1, 1)$, d the local dimension of the qudits and $\mathbf{1}$ the N -dimensional vector of ones. We refer to this as problem class II and denote the objective function vector by \vec{C}_{II} .

The third benchmark problem class also has two quadratic objectives. The first is implemented with AFM couplings described above, and the second is designed to minimize the distance of the randomly scaled search variables to a random point $\mathbf{x}_0 \sim \mathcal{U}([0, d-1])^N$. Consequently the cost function matrix is diagonal with randomized positive entries, i.e., $J_{ij} = \delta_{ij} J_r$ where $J_r \sim \mathcal{U}(a, b)$ with $0 < a < b$, and the local fields are determined by the scaled random point $\mathbf{h} = -2\mathbf{x}_0^\top \cdot \mathbf{J}$. We denote this as problem class III with objective vectors \vec{C}_{III} .

The fourth problem class denoted as \vec{C}_{IV} are three-objective problems ($K = 3$). The first and second objectives are AFM and FM-coupled spin models as in problem II, respectively, and the third objective is given by

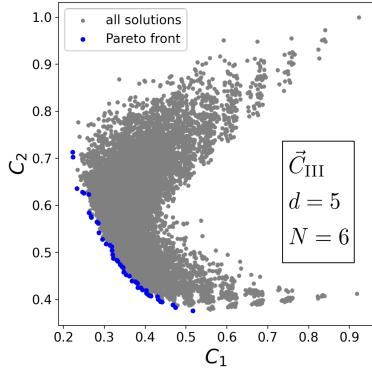


FIG. 2. Objective space of one example problem instance with $N = 6$ qudits ($d = 5$) of the benchmark problem type \vec{C}_{III} . The gray dots indicate all possible solutions, while the blue dots indicate the Pareto-optimal front.

the quadratic distance to a given random vector \mathbf{x}_0 as described for the second objective of problem class III.

The last class of benchmark problem has five objectives and is denoted by \vec{C}_V . The first two objectives are chosen to be the AFM and FM spin chains with nearest neighbor coupling, i.e., $J_{ij} = J_{ji} \sim \delta_{j,i+1} \mathcal{U}(a, b)$ with $b > a > 0$ and $a < b < 0$ for the AFM and the FM case, respectively. The third objective is the distance to a given random vector $\mathbf{x}_0 \sim \mathcal{U}([0, d-1]^N)$, i.e., $J_{ij} = \delta_{ij}$ and $\mathbf{h} = -2\mathbf{x}_0$. The fourth (fifth) objective is also realized as a nearest neighbor spin-chain, but where the first half of the variables, x_q with $q \leq N/2$, have FM (AFM) couplings while the latter half, x_q with $q > N/2$, have opposite, i.e. AFM (FM), couplings.

Figure 2 shows the objective space of one specific instance of the bi-objective problem class \vec{C}_{III} . The gray dots represent all possible solutions while the blue dots indicate the Pareto-optimal front. Note that we normalized each objective cost function such that the values are between $C(\mathbf{x}) \in [0, 1]$ for all functions, numbers of qudits N , and dimensionality of the qudits d . This allows us to always take the reference vector for the hypervolume calculation as the all-ones vector $\vec{r} = (1, \dots, 1)^\top$. More example instances of benchmark problems are found in Figure 8 in Appendix A.

B. Experimental Setup

The proposed quantum multi-objective optimization algorithm is evaluated on several instances of the previously introduced benchmark functions. For each specific randomly generated instance of a benchmark problem, the algorithm is run 50 times with different random initializations of the parameter values of the quantum circuits and the classical optimizers. For each run, all parameters are initialized randomly in the interval $[-\pi, \pi]$. In each iteration of the classical optimization algorithm, N_{shots} measurement shots are performed on the quan-

tum state prepared by the quantum circuit. From the measurement results the $N_S = 20$ most frequent classical solutions are selected. We vary N_{shots} between 128 and ∞ to investigate the dependency of the algorithm on measurement shot noise, which is a necessary step toward bringing the algorithm to real hardware.

We tested several classical optimization algorithms to determine the optimized parameters of the quantum circuit. We used COBYLA, L-BFGS-B, and Powell from the `scipy` library [55], covariance matrix adaptation evolutionary strategies (CMA-ES) [56] taken from [57], and in-house implementations of particle swarm optimization (PSO) [58], and differential evolution (DE) [59]. The overall performance of all tested optimization algorithms was roughly comparable to each other over all settings and benchmarks. An exception was the L-BFGS-B algorithm, which yields consistently lower hypervolume values, most likely because it is a quasi-Newton method with a numerical evaluation of the gradients. The population-based stochastic optimizers like PSO, CMA-ES, and DE gave slightly better results than Powell and COBYLA in most instances. However, given their large resource requirements, which are long algorithm runtimes and a large number of circuit evaluations, we focus on Powell algorithm.

For comparison, we also run several state-of-the-art classical multi-objective evolutionary optimization algorithms (MOEAs) from the `Platypus` [60] package. Specifically, these are NSGA-II, IBEA, and MOEA/D. We set the population size of the MOEAs to 20, comparable to $N_S = 20$ extracted solutions from the quantum states in QMOO. We execute each MOEA with 40 independent runs on each benchmark problem and limit the number of iterations to 200.

VII. RESULTS

Figure 3 shows the evolution of the hypervolume as a function of the classical optimization iterations for 50 runs of one specific instance of a five-objective benchmark problem \vec{C}_V . The algorithm increases the hypervolume over the iterations for all considered numbers of measurement shots N_{shots} . Even for as low as $N_{\text{shots}} = 128$ shots, it increases, and the algorithm produces increasingly better sets of Pareto efficient solutions. For lower N_{shots} the progress of the hypervolume is more noisy. In the context of optimization, this is not necessarily a drawback. More noise during the optimization implies more exploration in the search space, which can be beneficial or detrimental depending on the classical optimization algorithm.

The increase in the hypervolume indicates the algorithm's efficiency in generating increasingly better approximations of Pareto-efficient solution sets. To explicitly show this, we show an example of progress for one specific optimization run for a two-objective quadratic problem of class \vec{C}_{II} in Fig. 4. The two-objective problem was chosen for visualization purposes. The random

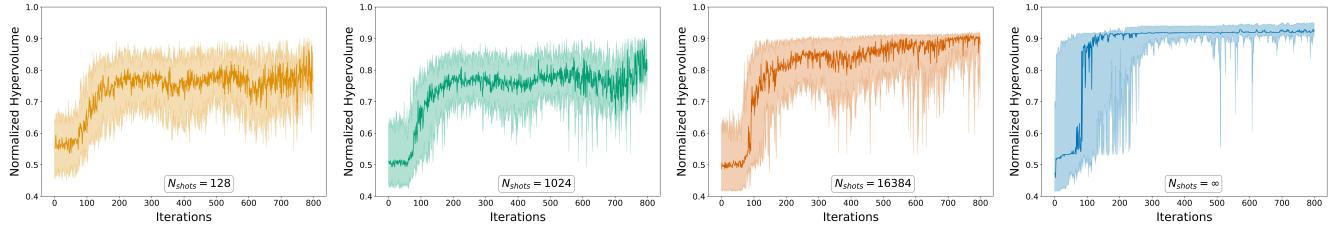


FIG. 3. Normalized hypervolume, i.e., hypervolume relative to the maximal achievable value of the Pareto front, as a function of the iterations of the QMOO algorithm for one particular instance of the five objective optimization problem using $N = 6$ ququints ($d = 5$). The thick line represents the median taken over 50 runs with different random initialization of the QMOO parameters, and the shaded region denotes the 20th and 80th quantiles. The classical Powell optimizer is used to optimize the quantum circuit consisting of $L = 1$ layer. The results are for increasing the number of measurement shots N_{shots} during the QMOO iterations (from left to right): $N_{\text{shots}} = 128$, $N_{\text{shots}} = 1024$, $N_{\text{shots}} = 16384$, and $N_{\text{shots}} = \infty$ (full state tomography). For all the measurement shots, the performance increases with several iterations. The lower number of shots N_{shots} leads to more noise, but already rather small numbers of shots, such as 128, give satisfactory results.

initialization leads to initial solutions in the top right part of the objective space. With increasing iterations, better parameters of the QMOO circuit are found, and the solution set represented in the final quantum state progresses toward the Pareto front.

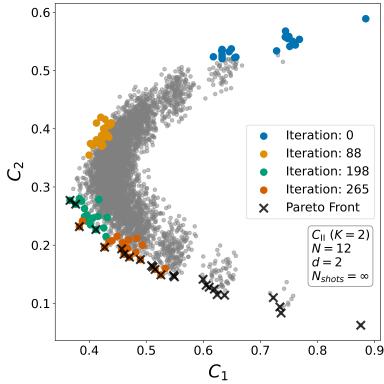


FIG. 4. Progression of the solution set of one run for a $L = 1$ layer QMOO circuit of the $K = 2$ objective quadratic problem of class C_{II} for $N = 12$ qubits. Gray dots show all possible solutions, while the colored dots show the solutions extracted from the QMOO at the indicated iterations. The Pareto front is shown with black crosses. Over the iteration, the solutions generated by the QMOO approach move toward the Pareto front.

When targeting to run the algorithm on actual hardware, it is mandatory to explore its behavior when changing the number of measurement shots N_{shots} performed in each iteration of QMOO to extract the N_S solutions. The results presented in Fig. 5 show that the dependency on N_{shots} is overall rather weak. The algorithm's performance is almost unchanged when increasing the shots from as low as 128 to as high as 16384. Even performing exact state tomography ($N_{\text{shots}} = \infty$) gives similar results. For some problem classes, the performance seems to slightly increase with an increasing number of shots, while all results are within the statistical variation of each other. However, for problem class C_{II} with two objectives, it seems that exact full-state tomography gives more statistical variations and even slightly worse results

than a finite number of shots. This hints at the fact that for this specific problem class, the noise induced by the measurement shots leads to a more explorative search in the classical optimization of the circuit, which seems to be beneficial for the optimization procedure.

In Fig. 5, we compare the performance of three classical multi-objective optimizers. For the two-objective problems, the proposed QMOO approach reaches up to but is still mostly inferior in performance to the classical approaches. At the same time, it is comparable to the three and five-objective problem classes. Also, it should be noted that the variation between the classical optimizers is also rather large for the three and five objective problems. This is a sign of the well-known fact that the performance of optimization algorithm can be very problem dependent and there is not one globally best algorithm [61, 62].

The effect of the choice of the classical optimizer and the influence of number of layers L in the quantum circuit were also investigated. Representative result are shown in Fig 6 for the problem class \tilde{C}_{IV} . We observed qualitatively similar results for all other problems and parameter choices.

The performance of all tested optimization algorithms can be clearly ranked. The CMA-ES gives the best results, while Powell and COBYLA are following closely. The L-BFGS-B approach gives significantly worse results with very large statistical variations. This is most likely a consequence of using numerically evaluated gradients, which are rather unstable for noisy cost functions. While achieving the best performance, the CMA-ES has the major drawback that it needs an order of magnitude more circuit evaluations since it operates with a population of solutions candidates (in our case 10), which need to be evaluated in each optimization iteration. Therefore, for our main results we used the Powell optimizer as it represents a very good trade-off between achievable results quality and required computational effort.

Figure 6 also shows that the QMOO results are comparable to the classical optimization algorithms for these problem instances. The performance increases as a func-

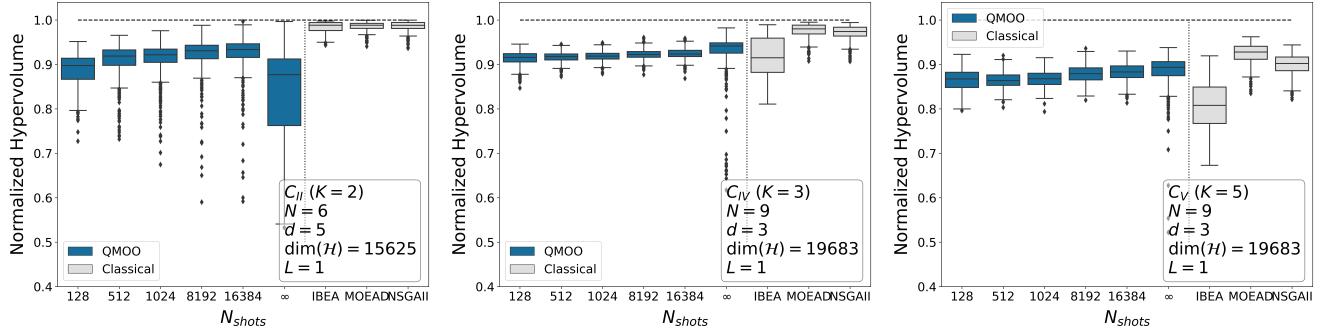


FIG. 5. Normalized hypervolume for $L = 1$ layer QMOO circuit as a function of shots for three problem classes. Left: $N = 6$ ququints ($d = 5$) on bi-objective problem class \tilde{C}_{II} . Middle: $N = 9$ qutrits ($d = 3$) for the three-objective problem class \tilde{C}_{IV} . Right: $N = 9$ qutrits ($d = 3$) for the five-objective problem class \tilde{C}_{V} . The results are from 11 problem instances and 50 randomly initialized runs for each problem instance. In each panel, results from three classical multi-objective algorithms are shown for comparison on the right in grey. The dependence of the QMOO results on the number of measurement shots is rather weak, and good results are already achieved for a low number of shots, much smaller than the Hilbert space dimension, which is of the order of 10^4 for these problems. The QMOO performance reaches up to the classical optimizers, having a slightly lower median for bi-objective problems (left) but comparable for three and five objectives (middle and right).

tion of the layers in the quantum circuit, as expected. However, this increase is only minor. This was the motivation to focus only on single-layer ($L = 1$) circuits in this proof-of-concept study.

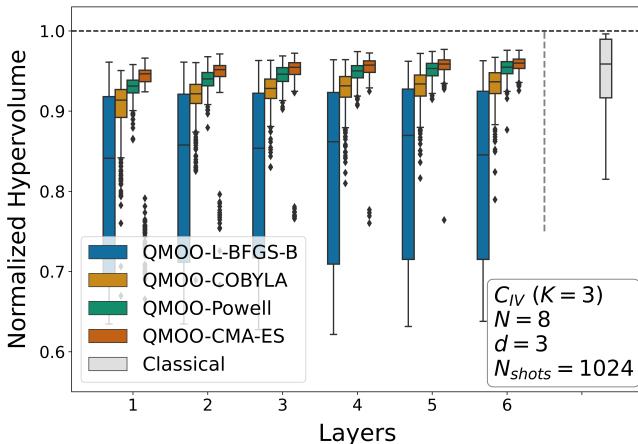


FIG. 6. Normalized hypervolume as a function of QMOO circuit layers for different classical optimizers for the problem class C_{IV} with $K = 3$ objectives for $N = 8$ qutrits ($d = 3$) using $N_{\text{shots}} = 1024$ shots. The Hilbert space dimension for these problems is $\dim(\mathcal{H}) = 6561$. The statistics are collected from 11 problem instances and 50 runs for each instance. The results from the purely classical optimization are shown on the right. The classical optimization algorithms can be clearly ranked from L-BFGS-B over COBYLA and Powell to CMA-ES (worst to best). The hypervolume increases with more layers for all cases, and the values are comparable to the classical approaches for this example.

We summarize the overall results of the proposed QMOO algorithm with $L = 1$ layer in Fig. 7. We show the performance for all five benchmark problem classes for a setup with $N = 13$ qubits ($d = 2$), $N = 9$ qutrits ($d = 3$), and $N = 6$ ququints ($d = 5$). We include results from QMOO using only 128 measurement shots

and when using full-state tomography ($N_{\text{shots}} = \infty$). We also compare the statistics of the achieved hypervolume values to the ones obtained with purely classical algorithms. The first observation is that the conclusions drawn from the previous analysis regarding the dependency on the number of shots hold for all studied problem setups. The performance with $N_{\text{shots}} = 128$ is similar to that obtained using exact state tomography. For problem class C_{III} , it even seems to be slightly larger with a lower number of shots, indicating enhanced performance due to more explorative search as a result of increased shot noise. Compared to the classical approaches, the QMOO approach performs slightly lower for problems with $K = 2$ objectives but is comparable for the three and five-objective problem classes. It should be noted that classical multi-objective algorithms are the result of considerable research efforts and have improved substantially over time by implementing best practices and adaptation strategies. In contrast, no such tuning or improvement has yet been made to the QMOO approach. Therefore, we may expect the QMOO performance to improve in the future by hyper-parameter tuning [63, 64] or warm-starting [12, 65–70], which is known have a potentially large impact.

VIII. DISCUSSION AND CONCLUSION

The proposed QMOO algorithm utilizes quantum superpositions as a resource to represent a whole set of classical solutions in one quantum state. We showed that this allows us to solve multi-objective optimization problems by optimizing a parametrized quantum circuit to maximize the hypervolume indicator for the extracted classical solutions. Unlike the standard single objective QAOA, the algorithm does not target the quantum state generated by the variational circuit to reduce to a single computational basis state (i.e., one classical solution).

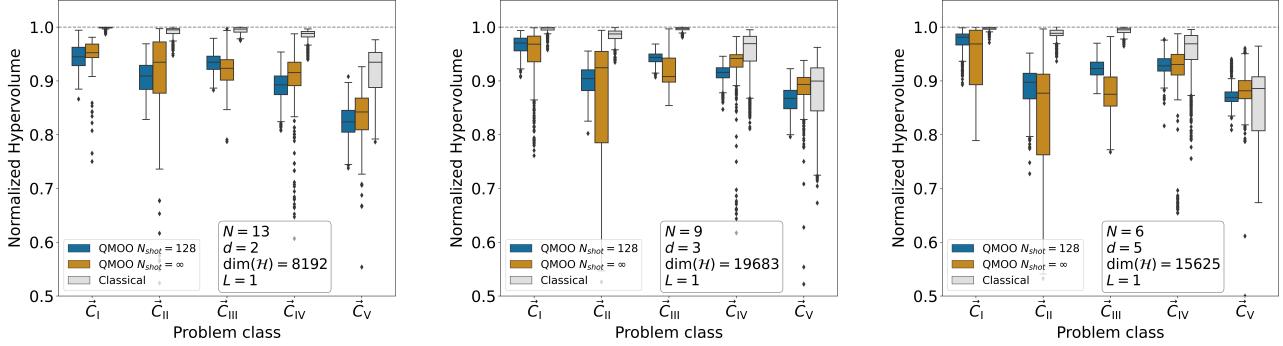


FIG. 7. Normalized hypervolume of the QMOO approach using $N_{\text{shots}} = 128$ measurement shots (blue) and exact state tomography, i.e. $N_{\text{shots}} = \infty$ (orange) on all benchmark problem classes for different setups compared to the results from purely classical multi-objective solvers (grey). Left: $N = 13$ qubits ($d = 2$). Middle: $N = 9$ qutrits ($d = 3$). Right: $N = 6$ ququints ($d = 5$). The quantum circuits only consisted of $L = 1$ layer and were optimized with the Powell optimizer. The results are for 11 different problem instances for each problem class and 50 randomized runs for each specific problem instance. The QMOO approach gives satisfactory results already with low number of shots and even reaches comparable performance to classical optimizers for the three and five objective problem classes, \vec{C}_{IV} and \vec{C}_V respectively.

Rather, the goal of the QMOO is to obtain a quantum state consisting of a superposition of non-dominated solutions, ideally belonging to the Pareto front of the multi-objective optimization problem.

We showed the effectiveness of the QMOO approach by evaluating its performance over a variety of benchmark problem classes. These classes are built from randomized linear and quadratic cost functions, which can be formulated with an arbitrary number of qudit variables. Throughout the variational optimization iterations, the hypervolume indicator increases, indicating a quality improvement of the approximation to the Pareto-front. The proposed QMOO approach was able to find solutions of similar quality to the purely classical methods for many cases. But overall, the performance is not yet fully competitive with state-of-the-art classical multi-objective algorithms.

The results suggest that the QMOO algorithm is inherently resilient to shot noise and does not require exponential scaling of the number of measurement shots with Hilbert space dimension. We evaluated this for Hilbert space dimension up to $\dim(\mathcal{H}) \approx 2 \cdot 10^4$ and could not detect a systematic degradation in performance with lower number of shots. As few as 128 measurement shots per iteration already gave consistently good results. This is reassuring in the light of running this algorithm on quantum hardware in the near future.

We are aware that our analysis merely constitutes a first step, and many issues need to be investigated in the future. In the context of qudit quantum circuits, a potential performance increase can be realized by including more generators of the $SU(d)$ group into the mixing operators [27]. Similarly, allowing each qudit to have an individual mixing parameter to increase the flexibility of the quantum circuit was shown to be beneficial in some applications for QAOA [71–73]. Including more than one

layer, as we focused on in this work, is also expected to increase performance as indicated by our result shown in Fig. 6. However, these measures come at the cost of greatly increasing the number of variational parameters and leading to a more complicated classical optimization problem. Finally, random initialization and direct optimization of all parameters of the circuit as we currently do it is not very efficient and much more favorable schemes exist [12, 65–70].

A very important topic to consider is trainability and the so-called barren plateau phenomenon [74]. On the one hand, the structure of each layer in the proposed QMOO algorithm is more complex than the standard QAOA, as there is one mixing and one phase operator for each objective. As circuit complexity is one route to barren plateaus, the QMOO circuit could be suspected to be prone to trainability issues for deep circuits with many layers. On the other hand, the QMOO approach does not need to calculate an expectation value with the output quantum state of the circuit. Rather, individual basis states are sampled from the quantum state and processed, which is also the reason for the resilience of the QMOO algorithm to a low number of measurement shots. Thus, one prominent source of trainability issues and barren plateaus is eliminated. The investigation of these issues is left for future research.

In summary, the proposed QMOO algorithm is a very promising novel approach to solving multi-objective optimization problems natively on quantum hardware. The approach requires few measurement shots and is, as such, suitable for running on actual quantum hardware. We are confident that this approach will widen the application domain for quantum-optimization approaches and hopefully allow for successful solutions to real-world problems of practical relevance. We consider this work a first step toward a truly quantum algorithm for solving multi-

objective optimization problems, which utilizes quantum resources for computing Pareto-optimal sets.

IX. ACKNOWLEDGEMENTS

We acknowledge fruitful discussions with T. Bäck, X. Bonet-Monroig, A. Bottarelli, V. Dunjko, P. Hauke and B. Sendhoff.

LE and SS acknowledge funding by the European Union under the Horizon Europe Programme. Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European

Commission. Neither the European Union nor the granting authority can be held responsible for them. Grant Agreement 101080086 — NeQST.

Appendix A: Visualization of exemplary benchmark problems

Figure 8 shows schematic examples of the linear and quadratic objective functions for two- and three-objective benchmarks for qubits ($d = 2$), qutrits ($d = 3$) and ququints ($d = 5$). The gray dots indicate all possible solutions while the blue dots denote Pareto-optimal solutions.

-
- [1] Google Quantum AI, *Nature* **614**, 676–681 (2023).
 - [2] Y. Kim, A. Eddins, S. Anand, K. Wei, E. Berg, S. Rosenblatt, H. Nayfeh, Y. Wu, M. Zaletel, K. Temme, and A. Kandala, *Nature* **618**, 500 (2023).
 - [3] D. Bluvstein, S. J. Evered, A. A. Geim, S. H. Li, H. Zhou, T. Manovitz, S. Ebadi, M. Cain, M. Kalinowski, D. Hangleiter, J. P. B. Ataides, N. Maskara, I. Cong, X. Gao, P. S. Rodriguez, T. Karolyshyn, G. Semeghini, M. J. Gullans, M. Greiner, V. Vuletić, and M. D. Lukin, *Nature* **626**, 58–65 (2023).
 - [4] M. P. da Silva, C. Ryan-Anderson, J. M. Bello-Rivas, A. Chernoguzov, J. M. Dreiling, C. Foltz, F. Frachon, J. P. Gaebler, T. M. Gatterman, L. Grans-Samuelsson, D. Hayes, N. Hewitt, J. Johansen, D. Lucchetti, M. Mills, S. A. Moses, B. Neyenhuis, A. Paz, J. Pino, P. Siegfried, J. Strabley, A. Sundaram, D. Tom, S. J. Wernli, M. Zanner, R. P. Stutz, and K. M. Svore, arXiv:2404.02280 10.48550/arXiv.2404.02280 (2024).
 - [5] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W.-K. Mok, S. Sim, L.-C. Kwek, and A. Aspuru-Guzik, *Rev. Mod. Phys.* **94**, 015004 (2022).
 - [6] Y. Deller, S. Schmitt, M. Lewenstein, S. Lenk, M. Federer, F. Jendrzejewski, P. Hauke, and V. Kasper, *Physical Review A* **107**, 062410 (2023).
 - [7] S. Yarkoni, E. Raponi, T. Bäck, and S. Schmitt, *Reports on Progress in Physics* **85**, 104001 (2022).
 - [8] A. Abbas, A. Ambainis, B. Augustino, A. Bärtschi, H. Buhrman, C. Coffrin, G. Cortiana, V. Dunjko, D. J. Egger, B. G. Elmegreen, N. Franco, F. Fratini, B. Fuller, J. Gacon, C. Gonciulea, S. Gribling, S. Gupta, S. Hadfield, R. Heese, G. Kircher, T. Kleinert, T. Koch, G. Korpas, S. Lenk, J. Marecek, V. Markov, G. Mazzola, S. Mensa, N. Mohseni, G. Nannicini, C. O’Meara, E. P. Tapia, S. Pokutta, M. Proissl, P. Rebentrost, E. Sahin, B. C. B. Symons, S. Tornow, V. Valls, S. Woerner, M. L. Wolf-Bauwens, J. Yard, S. Yarkoni, D. Zechiel, S. Zhuk, and C. Zoufal, arXiv:2312.02279 10.48550/arXiv.2312.02279 (2023).
 - [9] M. Veshchezerova, M. Somov, D. Bertsche, S. Limmer, S. Schmitt, M. Perelshtein, and A. Joshi Tripathi, in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Vol. 01 (2023) pp. 636–641.
 - [10] Dalyac, Constantin, Henriet, Loïc, Jeandel, Emmanuel, Lechner, Wolfgang, Perdrix, Simon, Porcheron, Marc, and Veshchezerova, Margarita, *EPJ Quantum Technol.* **8**, 12 (2021).
 - [11] E. Farhi, J. Goldstone, and S. Gutmann, arXiv:1411.4028 10.48550/arXiv.1411.4028 (2014).
 - [12] K. Blekos, D. Brand, A. Ceschini, C.-H. Chou, R.-H. Li, K. Pandya, and A. Summer, *Physics Reports* **1068**, 1 (2024), a review on Quantum Approximate Optimization Algorithm and its variants.
 - [13] E. Farhi, J. Goldstone, and S. Gutmann, arXiv:1412.6062 10.48550/arXiv.1412.6062 (2015).
 - [14] E. Farhi and A. W. Harrow, arXiv:1602.07674 10.48550/arXiv.1602.07674 (2019).
 - [15] S. Sharma and V. A. Kumar, *Arch Computat Methods Eng* **29**, 5605 (2022).
 - [16] J. L. J. Pereira, G. A. Oliver, M. B. Francisco, S. S. C. Jr, and G. F. Gomes, *Arch Computat Methods Eng* **29**, 2285 (2022).
 - [17] N. Chancellor, *Quantum Science and Technology* **4**, 045004 (2019).
 - [18] N. P. Sawaya, A. T. Schmitz, and S. Hadfield, *Quantum* **7**, 1111 (2023).
 - [19] Y. Wang, Z. Hu, B. C. Sanders, and S. Kais, *Frontiers in Physics* **8**, 589504 (2020).
 - [20] B. P. Lanyon, M. Barbieri, M. P. Almeida, T. Jennewein, T. C. Ralph, K. J. Resch, G. J. Pryde, J. L. O’Brien, A. Gilchrist, and A. G. White, *Nature Physics* **5**, 134–140 (2009).
 - [21] X. Gao, P. Appel, N. Friis, M. Ringbauer, and M. Huber, *Quantum* **7**, 1141 (2023).
 - [22] G. Duclos-Cianci and D. Poulin, *Phys. Rev. A* **87**, 062338 (2013).
 - [23] F. H. E. Watson, H. Anwar, and D. E. Browne, *Phys. Rev. A* **92**, 032309 (2015).
 - [24] Z. Gedik, I. Silva, B. Çakmak, G. Karpat, E. L. G. Viñedo, D. O. Soares-Pinto, E. R. deAzevedo, and F. F. Fanchini, *Sci Rep* **5**, 14671 (2015).
 - [25] G. Bottrill, M. Pandey, and O. Di Matteo, in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Vol. 01 (2023) pp. 177–183.
 - [26] S. Bravyi, A. Kliesch, R. Koenig, and E. Tang, *Quantum* **6**, 678 (2022).

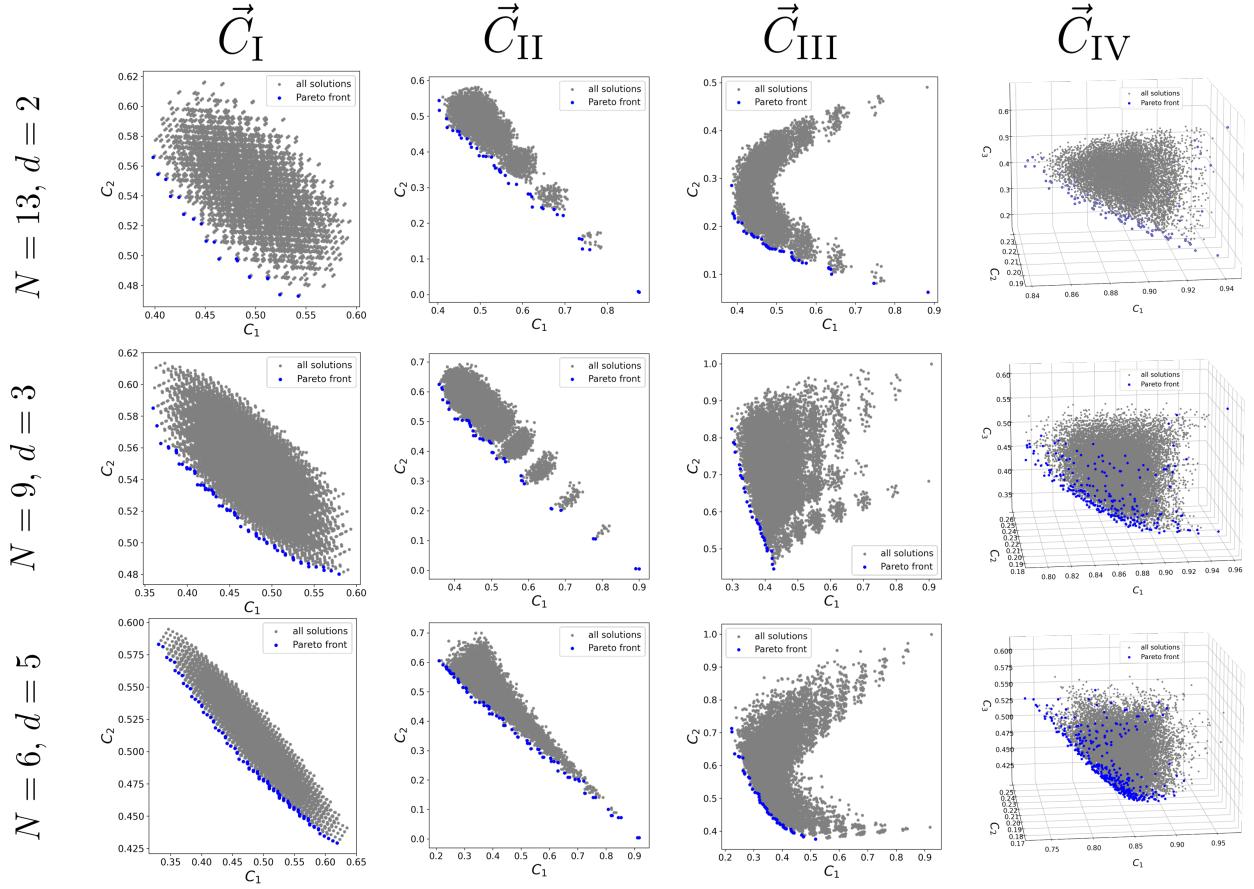


FIG. 8. Objective space of example instances of several benchmark problems. The problems are, from left to right: linear $K = 2$ objective C_I , two quadratic $K = 2$ problems C_{II} and C_{III} and a quadratic $K = 3$ objective problem C_{IV} . The top row is for $N = 13$ qubits ($d = 2$), the middle row is $N = 9$ qutrits ($d = 3$), and the lowest row is $N = 6$ ququints ($d = 5$). The gray dots indicate all possible solutions, while the blue dots indicate the true Pareto-optimal front.

- [27] N. L. Wach, M. S. Rudolph, F. Jendrzejewski, and S. Schmitt, *Quantum Mach. Intell.* **5**, 36 (2023).
- [28] S. Roca-Jerat, J. Román-Roche, and D. Zueco, arXiv:2308.16230 10.48550/arXiv.2308.16230 (2023).
- [29] D. H. Useche, A. Giraldo-Carvajal, H. M. Zuluaga-Bucheli, J. A. Jaramillo-Villegas, and F. A. González, *Quantum Inf Process* **21**, 12 (2022).
- [30] M. García de Andoin, A. Bottarelli, S. Schmitt, I. Oregi, P. Hauke, and M. Sanz, arXiv:2306.04414 10.48550/arXiv.2306.04414 (2023).
- [31] V. Vargas-Calderón, N. Parra-A., H. Vinck-Posada, and F. A. González, *Journal of the Physical Society of Japan* **90**, 114002 (2021).
- [32] A. Bottarelli, S. Schmitt, and P. Hauke, arXiv:2410.07674 10.48550/arXiv.2410.07674 (2024).
- [33] M. Ringbauer, M. Meth, L. Postler, R. Stricker, R. Blatt, P. Schindler, and T. Monz, *Nat. Phys.* **18**, 1053–1057 (2022).
- [34] P. Hrmo, B. Wilhelm, L. Gerster, M. W. van Mourik, M. Huber, R. Blatt, P. Schindler, T. Monz, and M. Ringbauer, *Nature Communications* **14**, 2242 (2023).
- [35] Y. Chi, J. Huang, Z. Zhang, J. Mao, Z. Zhou, X. Chen, C. Zhai, J. Bao, T. Dai, H. Yuan, M. Zhang, D. Dai, B. Tang, Y. Yang, Z. Li, Y. Ding, L. K. Oxenløwe, M. G. Thompson, J. L. O'Brien, Y. Li, Q. Gong, and J. Wang, *Nature Communications* **13**, 1166 (2022).
- [36] M. S. Blok, V. V. Ramasesh, T. Schuster, K. O'Brien, J. M. Kreikebaum, D. Dahmen, A. Morvan, B. Yoshida, N. Y. Yao, and I. Siddiqi, *Phys. Rev. X* **11**, 021010 (2021).
- [37] S.-H. Chiew, K. Poirier, R. Mishra, U. Bornheimer, E. Munro, S. H. Foon, C. W. Chen, W. S. Lim, and C. W. Nga, *IEEE Transactions on Quantum Engineering* **5**, 1 (2024).
- [38] P. Díez-Valle, J. Luis-Hita, S. Hernández-Santana, F. Martínez-García, Álvaro Díaz-Fernández, E. Andrés, J. J. García-Ripoll, E. Sánchez-Martínez, and D. Porras, *Quantum Science and Technology* **8**, 045009 (2023).
- [39] D. Chivilikhin, A. Samarin, V. Ulyantsev, I. Iorsh, A. Oganov, and O. Kyriienko, arXiv:2007.04424 10.48550/arXiv.2007.04424 (2020).
- [40] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca, *IEEE Trans. Evol. Comput.* **7**, 117 (2003).
- [41] C. Coello, G. Lamont, and D. van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Genetic and Evolutionary Computation (Springer US, 2007).
- [42] N. Beume, C. M. Fonseca, M. López-Ibáñez, L. Paquete, and J. Vahrenhold, *IEEE Trans. Evol. Comput.* **13**, 1075

- (2009).
- [43] N. Beume, *Evol. Comput.* **17**, 477 (2009).
- [44] A. P. Guerreiro, C. M. Fonseca, and L. Paquete, *ACM Comput. Surv.* **54**, 119:1 (2022).
- [45] E. Zitzler, D. Brockhoff, and L. Thiele, in *Evolutionary Multi-Criterion Optimization, 4th International Conference, EMO 2007, Matsushima, Japan, March 5-8, 2007, Proceedings*, Lecture Notes in Computer Science, Vol. 4403, edited by S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata (Springer, 2007) pp. 862–876.
- [46] J. G. Falcón-Cardona, S. Z. Martínez, and A. García-Nájera, in *GECCO '21: Genetic and Evolutionary Computation Conference, Lille, France, July 10-14, 2021*, edited by F. Chicano and K. Krawiec (ACM, 2021) pp. 395–402.
- [47] J. G. Falcón-Cardona, M. T. M. Emmerich, and C. A. C. Coello, *Evol. Comput.* **30**, 381 (2022).
- [48] A. Auger, J. Bader, D. Brockhoff, and E. Zitzler, in *Proceedings of the Tenth ACM SIGEVO Workshop on Foundations of Genetic Algorithms*, FOGA '09 (Association for Computing Machinery, New York, NY, USA, 2009) p. 87–102.
- [49] V. Kasper, D. González-Cuadra, A. Hegde, A. Xia, A. Dauphin, F. Huber, E. Tiemann, M. Lewenstein, F. Jendrzejewski, and P. Hauke, *Quantum Science and Technology* **7**, 015008 (2022).
- [50] P. Giorda, P. Zanardi, and S. Lloyd, *Physical Review A* **68**, 062320 (2003).
- [51] S. M. Barnett and S. Croke, *Advances in Optics and Photonics* **1**, 238 (2009).
- [52] M. K. Kurmapu, V. Tiunova, E. Tiunov, M. Ringbauer, C. Maier, R. Blatt, T. Monz, A. K. Fedorov, and A. Lvovsky, *PRX Quantum* **4**, 040345 (2023).
- [53] M. Paris and J. Řeháček, eds., *Quantum State Estimation*, Lecture Notes in Physics (Springer Berlin, Heidelberg, 2004).
- [54] S. Schmitt and L. Ekström, qmoo benchmark functions, https://github.com/HRI-EU/qmoo_benchmark_functions (2024).
- [55] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, *Nature Methods* **17**, 261 (2020).
- [56] N. Hansen, in *Towards a New Evolutionary Computation: Advances in the Estimation of Distribution Algorithms*, edited by J. A. Lozano, P. Larrañaga, I. Inza, and E. Benítez (Springer Berlin Heidelberg, Berlin, Heidelberg, 2006) pp. 75–102.
- [57] M. Shibata, CMA-ES, <https://github.com/CyberAgentAILab/cmaes> (2022).
- [58] J. Kennedy and R. Eberhart, in *Proceedings of ICNN'95 - International Conference on Neural Networks*, Vol. 4 (1995) pp. 1942–1948 vol.4.
- [59] R. Storn and K. Price, *Journal of Global Optimization* **11**, 341–359 (1997).
- [60] D. Hadka, Platypus - Multiobjective Optimization in Python, v1.1.0, <https://github.com/Project-Platypus/Platypus> (2022).
- [61] D. H. Wolpert, *Neural Computation* **8**, 1341 (1996), <https://direct.mit.edu/neco/article-pdf/8/7/1341/813495/neco.1996.8.7.1341.pdf>.
- [62] D. Wolpert and W. Macready, *IEEE Transactions on Evolutionary Computation* **1**, 67 (1997).
- [63] S. van Rijn, H. Wang, B. van Stein, and T. Bäck, in *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO '17* (Association for Computing Machinery, New York, NY, USA, 2017) p. 737–744.
- [64] M. López-Ibáñez, J. Dubois-Lacoste, L. Pérez Cáceres, M. Birattari, and T. Stützle, *Operations Research Perspectives* **3**, 43 (2016).
- [65] D. J. Egger, J. Mareček, and S. Woerner, *Quantum* **5**, 479 (2021).
- [66] V. Akshay, D. Rabinovich, E. Campos, and J. Biamonte, *Phys. Rev. A* **104**, L010401 (2021).
- [67] S. H. Sack, R. A. Medina, R. Kueng, and M. Serbyn, *Phys. Rev. A* **107**, 062404 (2023).
- [68] N. Jain, B. Coyle, E. Kashefi, and N. Kumar, *Quantum* **6**, 861 (2022).
- [69] S. Bravyi, A. Kliesch, R. Koenig, and E. Tang, *Phys. Rev. Lett.* **125**, 260505 (2020).
- [70] G. Scriva, N. Astrakhantsev, S. Pilati, and G. Mazzola, *Phys. Rev. A* **109**, 032408 (2024).
- [71] R. Herrman, P. Lotshaw, J. Ostrowski, T. Humble, and G. Siopsis, *Sci Rep* **12**, 6781 (2022).
- [72] Z. H. Saleem, T. Tomesh, B. Tariq, and M. Suchara, arXiv:2010.06660 (2021).
- [73] K. Shi, R. Herrman, R. Shaydulin, S. Chakrabarti, M. Pistoia, and J. Larson, in *2022 IEEE/ACM 7th Symposium on Edge Computing (SEC)* (2022) pp. 414–419.
- [74] M. Larocca, S. Thanasisip, S. Wang, K. Sharma, J. Biamonte, P. J. Coles, L. Cincio, J. R. McClean, Z. Holmes, and M. Cerezo, arXiv:2405.00781 10.48550/arXiv.2405.00781 (2024).