Out of the Loop: Structural Approximation of Optimisation Landscapes and non-Iterative Quantum Optimisation

Tom Krüger[1] and Wolfgang Mauerer[1, 2]

[1] *Technical University of Applied Sciences Regensburg, Regensburg, Germany*[*]
[2] *Siemens AG, Technology, Munich, Germany*[†]

The Quantum Approximate Optimisation Algorithm (QAOA) is a widely studied quantum-classical iterative heuristic for combinatorial optimisation. While QAOA targets problems in complexity class **NP**, the classical optimisation procedure required in every iteration is itself known to be **NP**-hard. Still, advantage over classical approaches is suspected for certain scenarios, but nature and origin of its computational power are not yet satisfactorily understood.

By introducing means of efficiently and accurately approximating the QAOA optimisation landscape from solution space structures, we derive a new algorithmic variant: Instead of performing an iterative quantum-classical computation for each input instance, our non-iterative method is based on a quantum circuit that is instance-independent, but problem-specific. It matches or outperforms unit-depth QAOA for key combinatorial problems, despite reduced computational effort.

Our approach is based on proving a long-standing conjecture regarding instance-independent structures in QAOA. By ensuring generality, we link existing empirical observations on QAOA parameter clustering to established approaches in theoretical computer science, and provide a sound foundation for understanding the link between structural properties of solution spaces and quantum optimisation.

I. INTRODUCTION

With the advent of early commercially available quantum computers, interest in the field has spilled from academia to early industrial adopters, and the hope for possible advantage or even supremacy is manifest [1–3]. However, challenges arise not only from deficiencies of noisy, intermediate-scale quantum (NISQ) hardware [4–7], but quantum algorithmic theory in general lags behind the classical case. While fundamental complexity-theoretic boundaries have long been established [8–10], a more precise understanding of concrete algorithmic building blocks and how to construct them is required [11–13].

How to systematically construct quantum algorithms is a multi-faceted, highly non-trivial task that remains essentially unsolved. Instead, heuristics like variational quantum circuits (VQC) [14] have emerged as popular alternatives. Originally intended to extract useful computational power from NISQ devices despite their limitations, they may also prove relevant as resource-efficient primitives in the post-NISQ era. They are centred around an iterative quantum-classical process that learns a parameterised form of a quantum circuit such that sampling a produced quantum state obtains a valid solution with high probability. Depending on the variational ansatz, this approach defers considerable aspects (*i.e.*, finding an appropriate quantum circuit that eventually implements the optimisation routine) of the task to classical components [15–17].

Structured forms of VQCs, in particular the quantum approximate optimisation algorithm (QAOA), enjoy popularity in prototypical applications [18]. QAOA is a specific

VQC for combinatorial optimisation. Similar to quantum annealing, to which QAOA is closely related [4, 19], it provides a strict framework on the quantum side. Given a specific problem, users only need to find a fitting problem representation, choose a suitable circuit depth, and pick a classical optimisation method. Devising such problem representations is not foreign to traditional computer science: In particular, techniques to reduce computational problems specified using an apt formalism into representations that are suitable for QAOA [20–22] are curricular knowledge [23].

The most essential ingredient of combinatorial optimisation is, of course, the optimisation landscape. This paper is centred around a new theorem, detailed in Section IV, that allows us to approximate the expected QAOA optimisation landscape of a problem from existing solution space structures. Informally speaking, QAOA is concerned with multiple objects: A problem (*e.g.*, can a graph be partitioned into two halves such that only a certain amount of edges must be cut?), an instance (*e.g.*, a specific graph), a solution space (*e.g.*, lists of edges to cut), and parameters that define an appropriate quantum circuit specific for each *instance* to obtain solutions from sampling the circuit output.

Hamming distances between solutions offer an intuitive handle to discover and describe local structures in solution space. Our approach continues a line of research [3, 24–26] using such distance information, particularly the connection to state amplitudes and their interference, to obtain insights about the inner workings of QAOA. By using stochastic methods, we establish novel means of reasoning about QAOA landscapes on an *instance independent*, but *problem specific* level, and show that instance-specific quantum circuits can be replaced by one single problem-generic alternative, while obeying a strictly bounded maximal difference between approximated and exact ingredients of the overall combinatorial optimisation process.

---

[*] tom.krueger@oth-regensburg.de
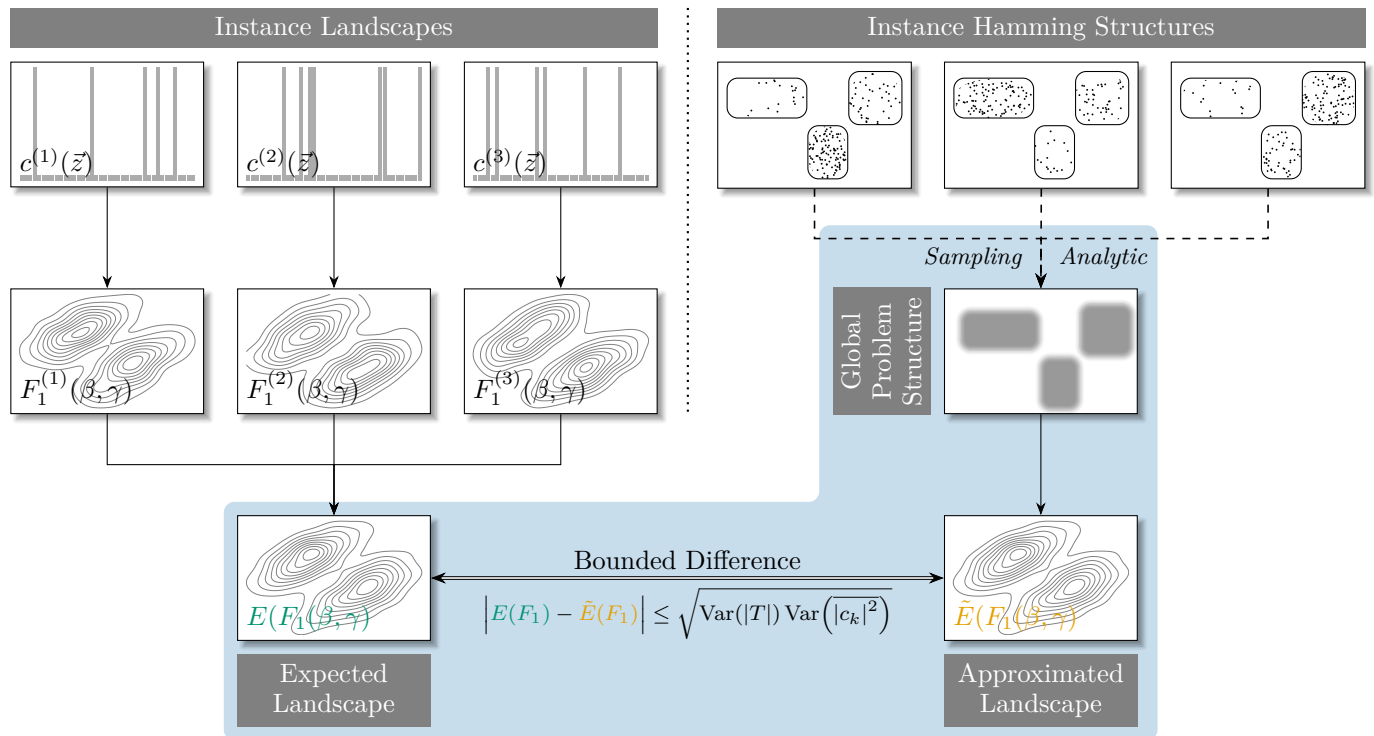[†] wolfgang.mauerer@othr.de

FIG. 1. Overview of our main foundational contributions. *Left:* Previously conjectured parameter clustering in QAOA optimisation landscapes $F_1^{(i)}(\beta, \gamma)$ (derived from a combinatorial optimisation objective function $c^{(i)}(\vec{z})$ of a decision problem instance) suggests that shared macroscopic similarities exist between instances i. Averaging over these reveals that the expected landscape $E(F_1(\beta, \gamma))$ is a common structure at the problem-global level. *Right:* Shared macroscopic features exist across all instances, based on structural properties manifest in the solution spaces. Aggregation (which may be possible analytically, but can always be performed using empirical sampling) leads to a macroscopic description of a problem-global solution space, from which the expected optimisation landscape can be efficiently approximated as $\tilde{E}(F_1(\beta, \gamma))$. Most importantly, we prove that the approximation has a bounded difference to the underlying exact quantity. The approximation approach (ingredients indicated by a blue background) and its consequences are subject of this paper.

Our result contributes to both, foundational understanding of QAOA and practical implementations.

As for the fundamental aspects, we provide a rigorous mathematical proof of a long-standing empirically motivated conjecture [27–31] that instance independent structures of a problem form the global structure of the QAOA optimisation landscape. We combine methods from physics and computer science to ascertain this structural insight, and can use either efficient sampling techniques or an analytical approach to obtain an accurate approximation of the QAOA optimisation landscape. In particular, we can separate the instance sampling from specific QAOA parameters. These aspects of our contribution are visually summarised in Fig. 1.

In terms of practical impact, we replace unit-depth QAOA as an iterative heuristic comprising an interplay of quantum and classical components (that, notably, requires solving an **NP**-hard parameter optimisation problem [32] individually for *every* instance) with a two-phase, non-iterative algorithm that first approximates the instance-independent, but problem-specific expected landscape followed by sampling a fixed quantum circuit, as illustrated in Fig. 2. While omitting the outer loop has, in

various ways, been suggested in prior work based on the aforementioned empirical observations [27, 28], our approach places the idea on a sound theoretical basis. It also introduces a new method of devising classical optimal parameters, and equips the method with a quantitative quality criterion. We demonstrate that our simplified variant is identical or better to standard QAOA for several seminal subject problems in terms of successfully finding solutions to combinatorial optimisation problems.

By showing that the required structures exist for all problems in **NP**, we establish generality of our results. Our approach therefore opens up the discussion on parameter clustering and optimisation landscape similarities to the vast body of established results in theoretical computer science on structural properties of problem solutions.

The rest of this paper is organised as follows: Section II first presents related work, supporting and motivating our ideas, and then establishes precise terminology for QAOA fundamentals, particularly given the many subtly different variants in current use. Section III covers the QAOA optimisation landscape and prepares the necessary groundwork for Section IV, where the main approxima-
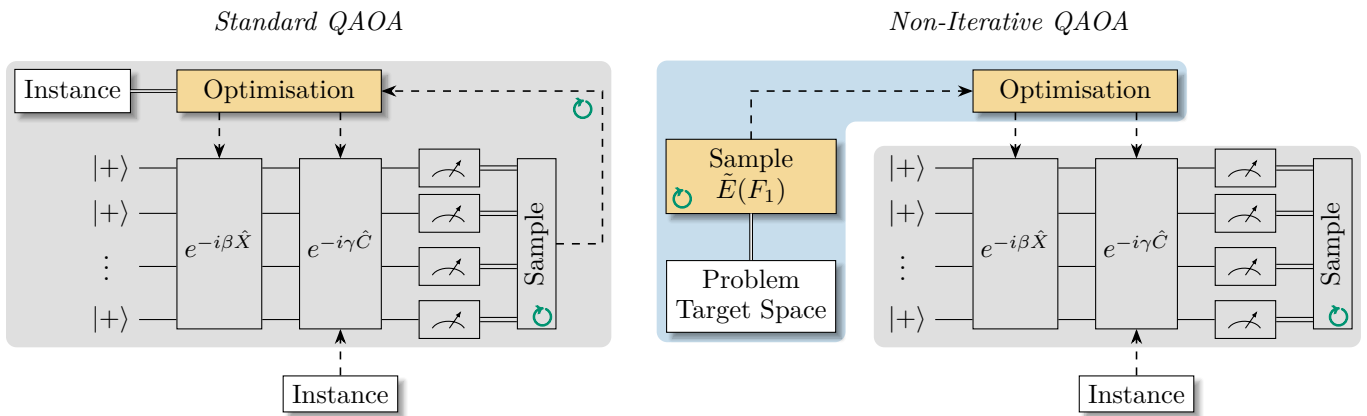
Standard QAOA *Non-Iterative QAOA*

FIG. 2. Overview of our main practical contributions (blue background ▨ indicates *problem-global*, grey background ▨ *instance-specific* components; yellow boxes ▨ mark classical computation. Dashed lines - - - indicate transfer of classical data, and double lines ══ denote querying a resource). *Left:* Standard QAOA that iteratively (symbolised by ↻) determines optimal parameters (β, γ) for every instance by repeatedly sampling a structured quantum circuit. *Right:* Two-phase QAOA approach introduced in this paper that first determines optimal parameters (β, γ) by sampling $\tilde{E}(F_1(\beta, \gamma))$ from the problem-global target space, and then uses the derived instance-independent optimal parameters (β, γ) to obtain instance-specific solutions by sampling from a quantum circuit that is constant for each instance $\hat{C}$. As we show, $\tilde{E}(F_1)$ has a strictly bounded difference to the expectation value of F_1, the core quantity of interest in QAOA.

tion theorem is presented and proved. With the new theorem at hand, we discuss a list of five examples building on each other in Section V. This illustrates how our theorem can be used to analyse synthetic and realistic computational problems. The practical application of our insights is subject of Section VI, where we introduce a novel non-iterative variant of QAOA based on our insights that matches or exceeds the performance of the standard heuristic for a set of subject problems, yet at substantially reduced computational cost. After setting our contributions in context with the current state of the art, and discussing potential further uses and future improvements in Section VII, we conclude in Section VIII.

II. FOUNDATIONS

A. Context and Related Work

QAOA and variants of the algorithm have been subject of intensive research [6, 15, 16, 20, 26, 29–31, 33–67], as recently reviewed by Blekos *et al.* [7] or Zhou *et al.* [17]. Apart from improving the understanding of the heuristic construction, modifications to the structure of the quantum circuit itself (*e.g.*, Refs. [17, 26, 44, 61, 68]) aim at improving performance especially in NISQ scenarios. Likewise, changes to the classical optimisation procedure (*e.g.*, Refs. [3, 28, 31, 54, 55, 60, 62, 63]) have been proposed. Given that limitations of NISQ hardware restrict programs to shallow circuits, many analytical and practical studies focus on unit-depth QAOA [7, 26, 29, 30, 33, 34, 36, 54, 55, 69, 70]; our considerations are also based on this commonly employed scenario. Perhaps fuelled by the possibility of empirically

investigating early-stage deployments, QAOA has been applied to a considerable variety of practical problems; see Bayerstadler *et al.* [18] for a review.

Our work is particularly motivated by efforts that observe concentrations of optimal QAOA parameters across instances, which has received substantial consideration in the literature. In 2020, Streif and Leib [28] observed a clustering of optimal parameters when training QAOA circuits for random Max-Cut instances, which inspired them to propose training QAOA without quantum hardware access. Wybo and Leib, recently used these methods to analyse fixed parameter QAOA landscapes for Ising formulations of the maximum cut and the maximum independent set problems [71]. Here, they also observed a promising performance of QAOA with fixed parameters. Sack and Serbyn [29] reported similar parameter concentrations in 2021 also for Max-Cut instances. They provided a theoretical definition of the effect based on the closeness of optimal sets in parameter space. Their highlighting the importance of picking good initialisation values for QAOA parameters evolved into *warm start QAOA*: Here, the focus lies on finding optimal initialisation parameters [54]; this is obviously tightly linked to optimal parameters concentrating in certain regions. Predicting the location of those clusters would benefit warm start efforts. Following that, Akshay *et al.* published a thorough analytic analysis, providing a new definition and concrete analytic insights [35]. They, for instance, showed that QAOA circuit parameters concentrate as an inverse polynomial in problem size. Galda *et al.* [30] demonstrated the transferability of QAOA parameters between different Max-Cut instances in 2023, and explained their findings with local graph properties. They also investigated optimisation landscapes of certain sub-graphs and observed stark similarities between differ-

ent sub-graphs. This overlaps with earlier observations by Brandao *et al.* in 2018 [27] that fixed parameters of the optimisation landscape concentrate around certain values for different instances. All these findings suggest the existence of higher level problem structures shared between instances that significantly influence the shape of the underlying optimisation landscapes.

Classical theoretical computer science has studied structural properties of problems for decades. Among the most fundamental structural observations are phase transitions in constraint satisfaction problems: At a certain point, problem instances transition from under-constrained to over-constrained. The parameter correlated with this effect depends on the problem: For the seminal problem of Boolean satisfiability (SAT), it is known to be the ratio of numbers of clauses to the number of variables in an instance [72]. For graph colouring, the connectivity of the underlying graph is the relevant determinant. Finding a solution is especially hard at the phase transition. While the phase transitions provide a high level description of problem hardness, more complex structural properties are known. Hogg argued in 1996 [73] that real-world applications model interactions between physical or social entities. Thus, interaction likelihood strongly depends on the distance between entities based on some appropriate distance measure, leading to recurring local structures. In 2004, Pari *et al.* [74] showed that even trivially sampled random SAT instances do possess structured solution spaces. The probability of a potential solution satisfying the SAT formula depends on the Hamming distance to other solutions. A few years later in 2011, Achlioptas *et al.* [75] discovered a clustering of the solution space of SAT instances. They further proved that under-constrained SAT formulas have exponentially many small clusters in their solution space. The presence of community structures in the solution space of industrial SAT instances was demonstrated in 2012 by Ansótegui *et al.* [76].

B. Quantum Approximate Optimisation Algorithm

More often than not, the *Quantum Approximate Optimisation Algorithm* (QAOA) is associated with the *quadratic unconstrained binary optimisation* (QUBO) problem, which is NP-complete in its decision form and relatively straight forward to solve with QAOA. A QUBO problem is defined by Boolean quadratic formula of the form $\sum_{i \neq j} a_{i,j} x_i x_j + \sum_i a_i x_i$, where $a_{i,j}, a_i \in \mathbb{R}$ are real valued weights of the Boolean variables $x_i \in \mathbb{F}_2$, with $\mathbb{F}_2 := \{0, 1\}$. The goal is to find a variable assignment to maximise this QUBO formula. This can be easily formulated as a ground state problem of an Ising Hamiltonian $\sum_{i \neq j} -J_{i,j} \sigma_i^z \sigma_j^z - \sum_i h_j \sigma_i^z$, with $J_{i,j}, h_j \in \mathbb{R}$ and σ_i^z being the Pauli-Z operator on the i-th qubit. Therefore, it seems reasonable to view QAOA as a dedicated QUBO solver. This approach is analogous to how SAT solvers are employed in classical systems. There is a rich community of SAT experts working on newer and better solvers, while

the users on the other side can rely on the interface of abstract SAT formulas to solve their concrete use cases without them needing to dive deep into the intricacies of Boolean satisfiability. Similarly, if we look at QAOA as just another QUBO solver, this takes the majority of *quantum* out of quantum computing as the QUBO formalism serves as a classical interface. This is arguably a major reason why QUBOs have been a welcoming entry point to quantum computing, particularly for researchers who do not feel the need to understand details of the computational process [22, 77]. Accompanying that, promising methods of solving QUBO problems with means other than quantum computing have enjoyed a certain amount of attention [78–82].

While our results apply to QUBO problems, our considerations are not restricted to this scenario, but consider QAOA as proposed by Farhi *et al.* to solve general combinatorial optimisation problems [69], of which QUBOs only form a restricted subset.

Definition 1. *Let $z \in \mathbb{F}_2^n$ be a n-bit binary string. Further $\{c_\alpha\}_{\alpha=1}^m$ shall be a set of Boolean clauses with $c_\alpha(z) = 1$ iff z satisfies c_α and $c_\alpha(z) = 0$ otherwise. Maximising*

$$c(z) = \sum_{\alpha=1}^m c_\alpha(z) \qquad (1)$$

is known as the Boolean constraint optimisation problem.

We can bring Definition 1 to the quantum world by defining a corresponding basis state vector $|z\rangle$ for each bit string $z \in \mathbb{F}_2^n$. The obvious choice for $|z\rangle$ is the computational basis vector $|z\rangle \in \{|0\rangle, |1\rangle\}^{\otimes n}$ encoded by z. Then we map the clause values $c_\alpha(z)$ to eigenvalues of quantum operators C_α representing the clauses. For a one to one mapping, we get a projector C_α per clause c_α that projects onto the subspace spanned by all states representing satisfying assignments of c_α. With Hermitian operators being closed under addition, we have that $C = \sum_{\alpha=1}^m C_\alpha$ is itself a Hermitian operator. This allows us to define the Hamiltonian time evolution operator $e^{-i\gamma C}$. Hamiltonian C describes the energy of a quantum system, and energy levels are eigenvalues of C. The solution z of Definition 1 maximises Eq. (1) and thus the corresponding eigenstate $|z\rangle$ of C has eigenvalue $\lambda_{\max} = \max \sigma(C)$. Here, $\sigma(C)$ is the set of eigenvalues of C. From the Hamiltonian point of view, solving the combinatorial optimisation problem is equivalent to finding a state with maximal energy of a system described by the Hamiltonian C. Farhi *et al.* came up with QAOA by trotterising an interpolated Hamiltonian time evolution from an easy to prepare maximal energy state of a system described by a simple Hamiltonian to the state of maximal energy of the system described by the constraint Hamiltonian C, in which the problem structure of Definition 1 is encoded.

Definition 2 (QAOA circuit as in Ref. [69])**.** *We consider a constraint Hamiltonian C implementing the constraint*

cost function Eq. (1) for $\boldsymbol{z} \in \mathbb{F}_2^n$ and a mixer Hamiltonian $X = \sum_{j=1}^n \sigma_j^x$, with $\sigma_i^x = \mathbb{1}^{\otimes i-1} \otimes \sigma^x \otimes \mathbb{1}^{n-i}$. Then, the QAOA circuit

$$U_p(\boldsymbol{\beta}, \boldsymbol{\gamma}) = e^{-i\beta_p X} e^{-i\gamma_p C} \cdots e^{-i\beta_1 X} e^{-i\gamma_1 C} \quad (2)$$

produces the state

$$|\boldsymbol{\beta}, \boldsymbol{\gamma}\rangle_p = U_p(\boldsymbol{\beta}, \boldsymbol{\gamma})|+\rangle^{\otimes n} \quad (3)$$

with real angles $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbb{R}^p$, $p \in \mathbb{N}$.

Definition 2 actually defines a parameterised family of circuits $\{U_p(\boldsymbol{\beta}, \boldsymbol{\gamma})\}_{\boldsymbol{\beta}, \boldsymbol{\gamma}, p}$. For the Hamiltonian implementation of Eq. (1), the evaluation $c(\boldsymbol{z})$ can be performed by calculating the expectation value $\langle \boldsymbol{z}|C|\boldsymbol{z}\rangle$. Farhi et al. showed that $\lim_{p \to \infty} \langle \boldsymbol{\beta}, \boldsymbol{\gamma}|_p C|\boldsymbol{\beta}, \boldsymbol{\gamma}\rangle_p = \max_{\boldsymbol{z}} c(\boldsymbol{z})$. This lets one define an algorithm to approximate the combinatorial optimisation problem with the circuit defined in Eq. (2).

Definition 3 (QAOA). *Consider a combinatorial optimisation problem with constraint cost function $c(\boldsymbol{z})$ and $\boldsymbol{z} \in \mathbb{F}_2^n$. For a fixed $p \in \mathbb{N}$, choose a set of angles $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbb{R}^p$ that maximises the QAOA cost function*

$$F_p(\boldsymbol{\beta}, \boldsymbol{\gamma}) = \langle \boldsymbol{\beta}, \boldsymbol{\gamma}|_p C|\boldsymbol{\beta}, \boldsymbol{\gamma}\rangle_p \quad (4)$$

Then construct the circuit $U_p(\boldsymbol{\beta}, \boldsymbol{\gamma})$, with which the state $|\boldsymbol{\beta}, \boldsymbol{\gamma}\rangle_p$ will be prepared and measured in the computational basis to produce a binary string $\boldsymbol{z} \in \mathbb{F}_2^n$. Repeat this sampling step m times with the same circuit to get a binary string that is close to $\max_{\boldsymbol{z}} c(\boldsymbol{z})$ with high probability.

Strictly speaking, Definition 3 defines a heuristic and not an algorithm. The process of finding optimal angles $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbb{R}^p$ is not further specified and open to interpretation. As shown by Farhi et al. [69], F_p can be simplified for specific problems—they consider Max-Cut on 3-regular graphs—which allows for finding an efficient classical evaluation of F_p. It would also be feasible to evaluate F_p on quantum hardware. The possible parameter optimisation methods are plentiful [7, 15, 16], and their impact is subject to ongoing research.

Even for $p = 1$, the parameter optimisation problem of $F_1(\beta, \gamma)$ is **NP**-hard [32]. We restrict our considerations to a single layer, as is common practice [7, 26, 29, 30, 33, 34, 36, 54, 55, 69, 70]. For the sake of simplicity, we also focus on constraint Hamiltonians with two-level eigenspectra. This avoids some effort in notation for the following theorems, but still allows us to solve problems in **NP**. Note that the structural properties we prove below also exist for constraint Hamiltonians C with more than two distinct eigenvalues. Also, two-level constraint Hamiltonians include all decision problems with classical proofs $\boldsymbol{z} \in \mathbb{F}_2^n$, most notably the complete class of **NP**. Despite the restriction to $|\sigma(C)| = 2$ and $p = 1$, our setting therefore covers a large body of non-trivial, interesting computational problems.

Before we proceed with our analysis, let us fix terminology regarding QAOA. The unitary gates $e^{-i\gamma_i C}$ defined by the Hamiltonian time evolution of C are usually called phase separation gates or just phase separators. As becomes clear in Eq. 7 below, it separates the solution space from the search space by a complex phase, and hence earns its name. The term $e^{-i\beta_i X}$ is usually called mixer, and X is the mixer Hamiltonian. Together, a phase separator and mixer pair forms a layer $e^{-i\beta_i X} e^{-i\gamma_i C}$. In this case we speak of the i-th layer of a p-layer QAOA circuit $U_p(\boldsymbol{\beta}, \boldsymbol{\gamma})$.

III. THE OPTIMISATION LANDSCAPE

Now we want to take a look at the optimisation landscape induced by the decision problem derived from Definition 1, which asks whether there exists an assignment $\boldsymbol{z} \in \mathbb{F}_2^n$ satisfying all clauses—or, equivalently, if $c(\boldsymbol{z}) = \prod_\alpha c_\alpha(\boldsymbol{z}) = 1$. This directly translates to the Hamiltonian implementation $C = \prod_\alpha C_\alpha$ which, as a product of projectors, remains a projector itself. Thus, the eigenspectrum of C is $\sigma(C) = \{0, 1\}$. Since C is Hermitian, there exists a unitary diagonalisation $C = U(\begin{smallmatrix} \mathbb{1} & \\ & \mathbf{0} \end{smallmatrix})U^\dagger$. Note that we can collect all eigenvalues $\lambda = 1$ in the upper left block of the diagonal matrix by simply applying a permutation operator, which we can subsume into the unitaries U and $U^\dagger$. Using the power series expansion of the exponential function, we see that the exponentiation can be passed through the diagonalisation. Let H be a diagonalisable operator with $H = UDU^\dagger$ with $D = \text{diag}(d_1, \ldots, d_n)$ and unitary U. Then using the power series expansion of the exponential function, we get $e^H = \sum_{k=0}^\infty \frac{1}{k!}(UDU^\dagger)^k$. Since U is a unitary operator, the inner contributions $U^\dagger U = \mathbb{1}$ vanish in the product, and $(UDU^\dagger)^k = UD^k U^\dagger$. It follows that

$$e^H = \sum_{k=0}^\infty \frac{1}{k!} U D^k U = U\left(\sum_{k=0}^\infty \frac{1}{k!} D^k\right)U^\dagger. \quad (5)$$

Since D is diagonal, we find that

$$\sum_{k=0}^\infty D^k = \text{diag}\left(\sum_{k=0}^\infty d_1^k, \ldots, \sum_{k=0}^\infty d_n^k\right) = \text{diag}(e^{d_1}, \ldots, e^{d_n}). \quad (6)$$

Consequently, we can express the phase separator by $e^{-i\gamma C} = U(\begin{smallmatrix} e^{-i\gamma}\mathbb{1} & \\ & \mathbb{1} \end{smallmatrix})U^\dagger$. Let $|z\rangle$ be a computational basis state. Then $(\begin{smallmatrix} e^{-i\gamma}\mathbb{1} & \\ & \mathbb{1} \end{smallmatrix})U^\dagger|z\rangle = e^{-i\gamma}U^\dagger|z\rangle = U^\dagger e^{-i\gamma}|z\rangle$ iff $|z\rangle$ has eigenvalue 1 under C, and $(\begin{smallmatrix} e^{-i\gamma}\mathbb{1} & \\ & \mathbb{1} \end{smallmatrix})U^\dagger|z\rangle = U^\dagger|z\rangle$ otherwise. If we apply the full phase separator gate on the state, we obtain $e^{-i\gamma C}|z\rangle = U(\begin{smallmatrix} e^{-i\gamma}\mathbb{1} & \\ & \mathbb{1} \end{smallmatrix})U^\dagger|z\rangle = UU^\dagger e^{i\gamma}|\phi\rangle = e^{i\gamma}|z\rangle$ iff $|z\rangle$ has eigenvalue 1 under C, and $UU^\dagger|z\rangle = |z\rangle$ otherwise. We conclude that $e^{-i\gamma C}$ adds a global phase to a computational basis state $|z\rangle$ if and only if $|z\rangle$ has the eigenvalue 1 under C, and leaves the state invariant otherwise. Applying $e^{-i\gamma C}$ to an arbitrary superposition of computational basis states $|\psi\rangle = \sum_{z=0}^{2^n-1} \omega_z|z\rangle$ can be characterised by

$$e^{-i\gamma C}|\psi\rangle = e^{-i\gamma} \sum_{|z\rangle \in T} \omega_z |z\rangle + \sum_{|z\rangle \notin T} \omega_z |z\rangle. \qquad (7)$$

Here, $T = \{|z\rangle \mid c(z) = 1\}$ is the target space of C, which directly maps to the solution space of Eq. (1). At this point, the amplitude symmetry is broken in the QAOA circuit, which allows for establishing interference effects whose pattern can be controlled by γ and β. This, eventually, allows us to benefit from quantum effects in the computational process.

The angle parameters β also have an interesting effect on state amplitudes. As we will shortly show in detail, solely the Hamming distances between states and the target/non-target partition of the state space suffice to describe the inner workings of QAOA circuits for decision problems. However, we need to analyse the effect of mixer layers before we can commence to proving this statement.

Lemma 1 (Projector version of [24]). *Let $X = \sum_{j=1}^{n} \sigma_j^x$ be the n-qubit mixer Hamiltonian. The effect of $e^{-i\beta X}$ on an arbitrary basis state $|z\rangle \in \mathbb{F}_2^n$ can be characterised by*

$$e^{-i\beta X}|z\rangle = \sum_{k \in \mathbb{F}_2^n} f(\beta, z, k)|k\rangle \qquad (8)$$

where f is defined as

$$f(\beta, z, k) := (\cos \beta)^{n - d_H(z,k)} (-i \sin \beta)^{d_H(z,k)}, \qquad (9)$$

and $d_H(z, k)$ is the Hamming distance of z and k.

Proof. Given that X is a 1-local Hamiltonian, we can analyse its time evolution by only considering its effect on single qubits. Let us look at $n = 1$ first: In this case, $X = \sigma^x$, and $e^{-i\beta X}|z\rangle = \cos \beta |z\rangle - i \sin \beta |z \oplus 1\rangle$ for $z \in \mathbb{F}_2$, which performs a bit flip with probability $(\sin \beta)^2$. For arbitrary $n \in \mathbb{N}$, this generalises to

$$e^{-i\beta X}|z\rangle = \bigotimes_{l=1}^{n} (\cos \beta |z_l\rangle - i \sin \beta |z_l \oplus 1\rangle). \qquad (10)$$

We would now like to express this state as a superposition of computational basis states, which can be achieved by reconstructing each of its components $|k \in \mathbb{F}_2^n\rangle$. From Eq. (10), we see that the amplitude of $|k\rangle$ acquires a multiplicative pre-factor of either $-i \sin \beta$ for each flipped, or $\cos \beta$ for each preserved qubit, respectively. In other words, the eventual amplitude of state $|k\rangle$ is given by $f(\beta, z, k) = (\cos \beta)^{n - d_H(z,k)} \cdot (-i \sin \beta)^{d_H(z,k)}$. $\square$

Now that we have characterised the mixer and phase separation layers individually, we have the tools to further our analysis. Next in line is the optimisation landscape, which is a central point of interest when analysing QAOA circuits.

Lemma 2. *A QAOA circuit with $p = 1$ constructed for general decision problems induces the optimisation landscape*

$$F_1(\beta, \gamma) = \frac{1}{2^n} \sum_{k \in T} |c_k(\beta, \gamma)|^2 \qquad (11)$$

with

$$c_k(\beta, \gamma) := \sum_{d=0}^{n} \left(\#_d(k) (e^{-i\gamma} - 1) + \binom{n}{d} \right) f_n(\beta, d) \quad (12)$$

where $\#_d(k) = |\{z \in T \mid d_H(z, k) = d\}|$ and $f_n(\beta, d) = (\cos \beta)^{n-d}(-i \sin \beta)^d$. Here, $d_H(z, k)$ is the Hamming distance between the bit strings of the binary representations of z and k.

Proof. By fixing $p = 1$ in Eq. (3), we obtain the state $|\beta, \gamma\rangle_1 = e^{-i\beta X} e^{-i\gamma C} |+\rangle^{\otimes n}$. After applying the phase separating gates to $|+\rangle^{\otimes n}$ and linearly pulling the mixer operators into the sums, we arrive at $\frac{1}{\sqrt{2^n}} \left(e^{-i\gamma} \sum_{z \in T} e^{-i\beta X} |z\rangle + \sum_{z \notin T} e^{-i\beta X} |z\rangle \right)$. From Lemma 1, it follows that

$$|\beta, \gamma\rangle_1 = \frac{1}{\sqrt{2^n}} \left(e^{-i\gamma} \sum_{z \in T} \sum_{k \in \mathbb{F}_2^n} f(\beta, z, k)|k\rangle + \right.$$
$$\left. \sum_{z \notin T} \sum_{k \in \mathbb{F}_2^n} f(\beta, z, k)|k\rangle \right) \qquad (13)$$

By reordering the sums and factoring out $|k\rangle$ in Eq. (13), we can express the state $|\beta, \gamma\rangle_1$ as $\frac{1}{\sqrt{2^n}} \sum_{k \in \mathbb{F}_2^n} c_k(\beta, \gamma)|k\rangle$ with $c_k(\beta, \gamma) = e^{-i\gamma} \sum_{z \in T} f(\beta, z, k) + \sum_{z \notin T} f(\beta, z, k)$ (note that we omit parameters (β, γ) on c_k and other quantities below when the dependency is clear from the context). If we apply C to $|\beta, \gamma\rangle$ we basically filter out all $|k\rangle \notin T$ by linearly pulling C into the sum, therefore obtaining $C|\beta, \gamma\rangle = \frac{1}{\sqrt{2^n}} \sum_{k \in T} c_k(\beta, \gamma)|k\rangle$. We conclude that

$$F_1(\beta, \gamma) = \langle \beta, \gamma|_1 C |\beta, \gamma\rangle_1$$
$$= \frac{1}{2^n} \left(\sum_{k \in \mathbb{F}_2^n} \overline{c_k(\beta, \gamma)} \langle k| \right) \left(\sum_{k \in T} c_k(\beta, \gamma)|k\rangle \right) \qquad (14)$$
$$= \frac{1}{2^n} \sum_{k \in T} \overline{c_k(\beta, \gamma)} c_k(\beta, \gamma) = \frac{1}{2^n} \sum_{k \in T} |c_k(\beta, \gamma)|^2. \qquad (15)$$

The step from Eq. (14) to Eq. (15) follows since all pairs in $\{|k\rangle \mid k \in \mathbb{F}_2^n\} \times \{|k\rangle \mid k \in T\}$ comprise orthogonal states. Note that $c_k(\beta, \gamma)$ essentially contains a sum over all $z \in \mathbb{F}_2^n$ with an optional phase factor of $e^{-i\gamma}$ if $z \in T$. Thus, we find that

$$c_k(\beta, \gamma) = \sum_{z \in \mathbb{F}_2^n} \chi_T(z) e^{-i\gamma} f(\beta, z, k) + (1 - \chi_T(z)) f(\beta, z, k).$$
$$(16)$$

Here $\chi_T(z) = 1$ iff $z \in T$ and otherwise $\chi_T(z) = 0$. By recalling the definition of $f(\beta, z, k)$ in Eq. (9), we recognise $(\cos\beta)^{n-d}(-i\sin\beta)^d$ as the basic shape of f, where d is the Hamming distance between two concrete states $k, z \in \mathbb{F}_2^n$. This means that for a fixed angle β_c, $f(\beta_c, k, z)$ is actually a function of the Hamming distance between two states. Therefore, while the sum in Eq. (16) iterates over *all* 2^n states z, there are effectively only $n+1$ *different* basic terms in the sum—one for each possible distance d. Thus, if we count the number of occurrences $\#_d(k) = |\{z \in T \mid d_H(z, k) = d\}|$ of each distance d between k and other states in the target set, we can reorder the sum to obtain:

$$c_k(\beta, \gamma) = \sum_{d=0}^{n} \#_d(k) e^{-i\gamma} f_n(\beta, d) +$$
$$\left(\binom{n}{d} - \#_d(k) \right) f_n(\beta, d)$$

with

$$f_n(\beta, d) := (\cos\beta)^{n-d}(-i\sin\beta)^d. \tag{17}$$

Now it is simply a matter of factoring out f_n and $\#_d$ to arrive at Eq. (12). □

IV. THE QAOA APPROXIMATION THEOREM

We now state and prove our main result.

A. Intuition

A straightforward description of an optimisation landscape might be directly derived by considering all point to point interactions between all states in superposition. Obviously, there are exponentially many state such interactions to consider. In Section III, we captured those effects in $f(\beta, k, z)$ (see Eq. (9)) and showed that there are only $n+1$ different outcomes that can occur based on the Hamming distance between two states. Now we use this reduction in complexity of the effect domain to derive an approximation theorem for the *expected optimisation landscape* $E(F_1)$ of a specific problem.

B. Formalisation

In Section III, we presented a closed form of the optimisation landscape based on the Hamming distances between states. The components $|c_k(\beta, \gamma)|^2$ in Eq. (12) play a crucial role in $F_1(\beta, \gamma)$ (see Eq. (11)). As they are functions of the optimisation angles, we find it prudent to first obtain an intuitive visual understanding of their interrelationship. Each contribution $|c_k(\beta, \gamma)|^2$ can be seen as a landscape for a specific value of k itself. Consider Fig. 3 that depicts, as an illustration, vertical slices of

three $|c_k(\beta, \gamma)|^2$ for $k \in T = \{10, 13, 14\}$ and dimension $n = 5$. The macroscopic similarities between the three are visually obvious, which motivate the idea to take the mean over all $|c_k(\beta, \gamma)|^2$. To further quantify this idea, we look at the overlaid cross sections for all three components compared with their mean in Fig. 4. Here, we evaluate $|c_k(\beta, \gamma)|^2$ for all $k \in T = \{10, 13, 14\}$ across $0 \le \beta \le \pi$ at a fixed $\gamma_c = 1.2$ and calculate the mean of all $|c_k(\beta, \gamma_c)|^2$ at each point. The chosen constant 1.2 does not have any particular computational or physical significance, but results in a "typical" two-dimensional sub-space of the optimisation landscape. We again observe how the mean value nicely captures the higher level structures shared between all individual components. In fact, this intuition can be expressed as a precise mathematical relationship, as F_1 is nothing more than this mean value scaled by a factor of $\frac{|T|}{2^n}$, since

$$\frac{|T|}{2^n}\overline{|c_k(\beta, \gamma)|^2} = \frac{|T|}{2^n}\frac{1}{|T|}\sum_{k \in T}|c_k|^2 = \frac{1}{2^n}\sum_{k \in T}|c_k|^2 = F_1(\beta, \gamma).$$
$$\tag{18}$$

As remarked above, the global structure of F_1 is very similar across different instances of a given problem. This renders the *expected value* of an optimisation landscape for a problem an interesting object of study, as it captures the salient properties *across* instances. In particular, we argue that the ability to efficiently approximate is well suited to improve the understanding of QAOA, as we show in Section V, and also leads to practical implications that can help optimise the use of QAOA, as we detail in Section VI.

Before that, and based on these observations, let us however first state and prove our main theorem for efficiently approximating the expected value of $F_1(\beta, \gamma)$ across instances of a computational problem:

Theorem 1. *Let $\#_d(k)$ be defined as in Lemma 2, then we can approximate the expected value of $F_1(\beta, \gamma)$ for a random instance of a problem by*

$$\tilde{E}(F_1(\beta, \gamma)) = \frac{E(|T|)}{2^n}E\left(\overline{|c_k(\beta, \gamma)|^2}\right)$$

with

$$E\left(\overline{|c_k(\beta, \gamma)|^2}\right) = \sum_{d_1, d_2=0}^{n} w_{d_1, d_2}(\gamma) f_n(\beta, d_1) f_n(\beta, d_2)^*,$$
$$\tag{19}$$

where $w_{d_1, d_2}(\gamma)$ is

$$w_{d_1, d_2}(\gamma) = E\left(\overline{\#_{d_1}(k)\#_{d_2}(k)}\right)\left(e^{-i\gamma} - 1\right)\left(e^{i\gamma} - 1\right) +$$
$$E\left(\overline{\#_{d_1}(k)}\right)\left(e^{-i\gamma} - 1\right)\binom{n}{d_2} +$$
$$E\left(\overline{\#_{d_2}(k)}\right)\left(e^{i\gamma} - 1\right)\binom{n}{d_1} + \binom{n}{d_1}\binom{n}{d_2}.$$

We calculate mean values over all k in the instance specific target set T. The absolute approximation error is bounded by $\left|E(F_1(\beta, \gamma)) - \tilde{E}(F_1(\beta, \gamma))\right| \le \sqrt{\text{Var}(|T|)\,\text{Var}\left(\overline{|c_k|^2}\right)}.$
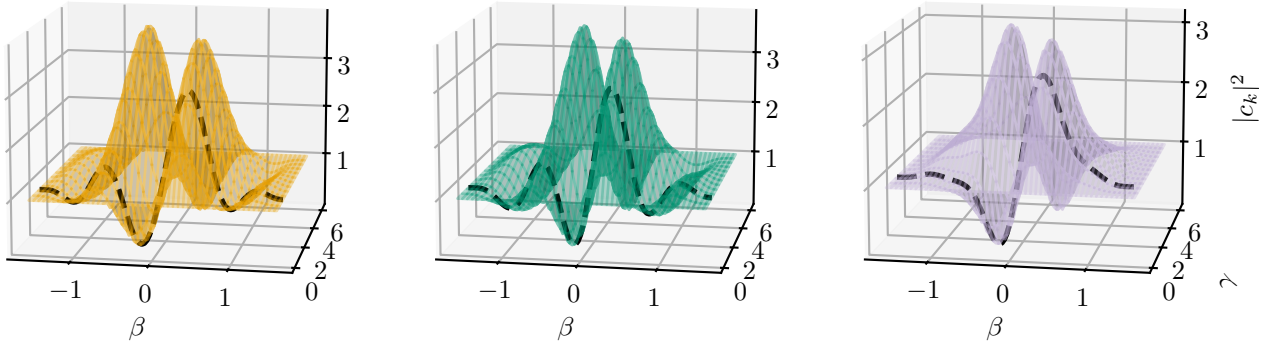
FIG. 3. Three landscape components $|c_k(\beta,\gamma)|^2$ for $k \in \{10,13,14\} = T$ and a state space of dimension $n = 5$ are depicted in this order from left to right. The landscapes are represented by an array of vertical cross sections along β.
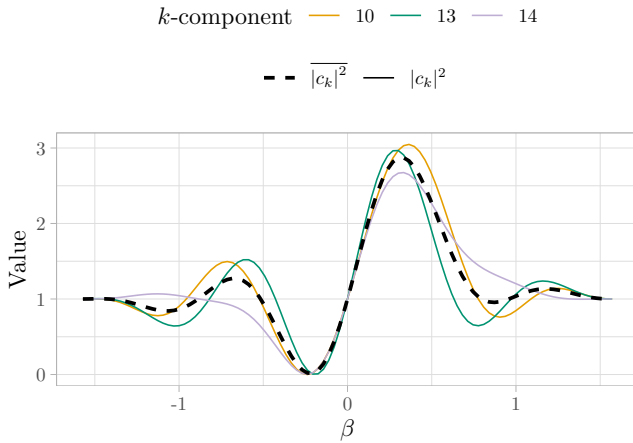

FIG. 4. Cross section of the optimisation landscape components $|c_k(\beta,\gamma_c)|^2$ depicted in Fig. 3 at $\gamma_c = 1.2$ for $k \in \{10,13,14\} \subset T$ (note matching colours). The dashed line shows $E(|c_k(\beta,\gamma_c)|^2)$ for these three components, and illustrates how the mean value captures the globally relevant features of instance-specific information.

Proof. We start by reformulating $|c_k(\beta,\gamma)|^2$. Recall that $|c_k(\beta,\gamma)|^2 = c_k(\beta,\gamma)c_k(\beta,\gamma)^*$ and $c_k(\beta,\gamma) = \sum_{d=0}^{n}\left(\#_d(k)(e^{-i\gamma}-1) + \binom{n}{d}\right)f_n(\beta,d)$. Then we use the distributivity of the complex conjugate over addition and multiplication to pull it into the sum:

$$c_k(\beta,\gamma)^* = \sum_{d=0}^{n}\left(\#_d(k)\left((e^{-i\gamma})^* - 1\right) + \binom{n}{d}\right)f_n(\beta,d)^*$$
$$= \sum_{d=0}^{n}\left(\#_d(k)(e^{i\gamma} - 1) + \binom{n}{d}\right)f_n(\beta,d)^*$$

where the second equality follows from $(e^z)^* = e^{z^*}$. From this we see, that

$$|c_k(\beta,\gamma)|^2 = \left(\sum_{d_1=0}^{n}\left(\#_{d_1}(k)(e^{-i\gamma}-1) + \binom{n}{d_1}\right)f_n(\beta,d_1)\right) \cdot$$
$$\left(\sum_{d_2=0}^{n}\left(\#_{d_2}(k)(e^{i\gamma}-1) + \binom{n}{d_2}\right)f_n(\beta,d_2)^*\right)$$

and further

$$|c_k(\beta,\gamma)|^2 = \sum_{d_1,d_2=0}^{n}\left(\#_{d_1}(k)\#_{d_2}(k)(e^{-i\gamma}-1)(e^{i\gamma}-1)+\right.$$
$$\#_{d_1}(k)(e^{-i\gamma}-1)\binom{n}{d_2}+$$
$$\left.\#_{d_2}(k)(e^{i\gamma}-1)\binom{n}{d_1} + \binom{n}{d_1}\binom{n}{d_2}\right).$$

Now from $\overline{|c_k(\beta,\gamma)|^2} = \frac{1}{|T|}\sum_{k\in T}|c_k(\beta,\gamma)|^2$ it follows by reordering the sum that

$$|c_k(\beta,\gamma)|^2 = \sum_{d_1,d_2=0}^{n}\left(\overline{\#_{d_1}(k)\#_{d_2}(k)}(e^{-i\gamma}-1)(e^{i\gamma}-1)+\right.$$
$$\overline{\#_{d_1}(k)}(e^{-i\gamma}-1)\binom{n}{d_2}+$$
$$\left.\overline{\#_{d_2}(k)}(e^{i\gamma}-1)\binom{n}{d_1} + \binom{n}{d_1}\binom{n}{d_2}\right)$$

where $\overline{\#_d(k)} = \frac{1}{|T|}\sum_{k\in T}\#_d(k)$. Finally, Eq. (19) follows from the linearity of the expected value. From Eq. (18) we deduce that $E(F_1) = E\left(\frac{|T|}{2^n}\overline{|c_k|^2}\right) = \frac{E(|T|)}{2^n}E\left(\overline{|c_k|^2}\right) + \text{Cov}\left(|T|,\overline{|c_k|^2}\right)$. Therefore we conclude that $\left|\tilde{E}(F_1) - E(F_1)\right| = \left|\text{Cov}\left(|T|,\overline{|c_k|^2}\right)\right| \leq \sqrt{\text{Var}(|T|)\,\text{Var}\left(\overline{|c_k|^2}\right)}$. $\square$

C. Generality

The approximation theorem only operates on the Hamming distance structure of problem target spaces. Recall, however, that a target space is isomorphic to the solution space of a problem by an encoding function. A more complex constraint Hamiltonian might also need to operate on ancilla qubits that are not part of the target space. We now show that for every problem in **NP**, it is possible to efficiently construct an ancilla register independent constraint Hamiltonian. This guarantees a wide applicability of Theorem 1.

Theorem 2. *For every problem in* **NP** *there exists a family of constraint Hamiltonians* $\{C_n\}_{n\in\mathbb{N}}$ *that can be efficiently constructed such that*

$$C_n|z\rangle|\mathbf{0}\rangle = \begin{cases} |z\rangle|\mathbf{0}\rangle & z \text{ proves the decision property} \\ 0 & otherwise \end{cases}$$
(20)

for proofs z of length n.

Proof. By definition a problem is in **NP** if and only if there exist an efficient verifier v that returns $v(z) = 1$ on all valid proofs z of the decision property and $v(z) = 0$ on all other inputs. Thus, there also exits an efficiently constructable family of quantum circuits $\{V_n\}_{n\in\mathbb{N}}$: $V_n|z\rangle|\mathbf{0}\rangle = |z\rangle|\mathbf{z}\rangle$ with $\mathbf{z} = \big(v(z), v_2, \ldots, v_{p(n)}\big)$, where $|\mathbf{0}\rangle = |0\rangle^{\otimes p(n)}$ is an ancilla register of size $p(n)$ for some polynomial p. Now, the construction of C_n looks as follows: $C_n := V_n^\dagger(\mathbb{1}^{\otimes n} \otimes |1\rangle\langle 1| \otimes \mathbb{1}^{p(n)-1})V_n$. $\square$

Remark 3. *Theorem 2 ensures us that if a problem is in* **NP***, we only have to consider its target space T of problems when applying Theorem 1. Remember that T is isomorphic to the solution or proof space of a problem. In essence, Theorem 1 addresses structural properties of the expected solution space of a problem. So, the combination of Theorem 1 and Theorem 2 shows the existence of macroscopic structures on a problem level that significantly influence the optimisation landscapes induced by* QAOA *circuits for at least all* **NP** *problems.*

V. APPLICATION

After having set the foundations for a methodology to understand structural properties of optimisation problems across instances, let us now commence with applying the framework to several concrete examples. We consider five different subject problems respectively scenarios (uniform random sampling, clustered sampling, Boolean satisfiability, k-clique, and one-way functions in the form of qr factoring) that build on each other to best introduce the application of the QAOA approximation theorem on real problems. Overall, the selection of examples is carefully curated to highlight different aspects of using our approximation theorem: We first compare the two possible approaches of either analytically or empirically analysing the target space structure of a problem at hand. Then we demonstrate with a purposefully constructed sampling method that the existence of a stochastic dependence of two states with a certain Hamming distance in a random instance significantly influences he optimisation landscape. Following that, we use SAT as a first straight forward real world example with an easy to construct constraint Hamiltonian. After that, we show that for all problems in **NP**, even with more complicated constraints, there is a construction for an appropriate constraint Hamiltonian that satisfies the preconditions of our approximation theorem. As a final example, we highlight the case of integer factoring to argue that problems based on one-way functions are interesting subjects for our methods as their target space can be relatively easy characterised.

A. Uniform Random Sampling

Central in our theory is the target space T of the constraint projector C. Given a concrete interpretation, every state in T corresponds to a problem solution.[1] Thus, our notion of target spaces T is an abstraction of concrete problem solution spaces. Sampling a random target space T equals sampling a random problem minus the abstraction of a target state interpretation. Furthermore, the description of a sampling procedure for T defines an abstract random problem, where the problem itself becomes a random variable in the probabilistic point of view. To provide a smooth onramp to more complex examples further below, we start by simply sampling T by randomly drawing states from the state space with uniform probability.

From Theorem 1, we see that $E(F_1)$ can be efficiently approximated if the quantities (a) $|T|$, (b) $E\left(\overline{\#_d(k)}\right)$ for all $0 \le d \le n$, and (c) $\text{Cov}\left(\overline{\#_{d_1}(k)}, \overline{\#_{d_2}(k)}\right)$ for all $0 \le d_1, d_2 \le n$ are provided. Recall that the expected value is calculated over all instances of a specific problem, thus $\overline{\#_d(k)}$ are random variables describing the mean value $\frac{1}{|T|}\sum_{k\in T} \#_d(k)$ over the target set T of a random problem instance. There are, in general, two approaches: We can, if possible, determine the distribution of the random variables $\overline{\#_d(k)}$ by analytical means, or by an empirical numerical approach. For uniform sampling as considered in this motivating example, it is relatively straightforward to execute the necessary analytic calculations.

1. Analytic Approach

Definition 4. *We consider an urn model with balls of $m \in \mathbb{N}$ different colours. The population of all balls in*

------

[1] Note that given different interpretations, a state in T can encode different solutions of different problems, too.

the urn is described by $\boldsymbol{X} = (X_1, X_2, \ldots, X_m)$, where X_i is the number of balls with colour i. Then, if n balls are drawn without replacement, the probability of having x_i balls of colour i in the sample is given by the multivariate hypergeometric distribution $P(\boldsymbol{x}) = \mathrm{HypG}(\boldsymbol{X}, \boldsymbol{x})$ with $\boldsymbol{x} = (x_1, x_2, \ldots, x_m)$. Recall that, by textbook knowledge, the hypergeometric distribution is defined by $P(\boldsymbol{x}) = \prod_{i=1}^{m} \binom{X_i}{x_i} / \binom{N}{n}$, with $N := \sum_{i=1}^{m} X_i$.

The size of T is trivially given, as we always sample a fixed number of states. In the following discussion we will encounter expected values over different sample spaces, we will differentiate this by $E_T(\cdot)$ and $E_{\mathcal{I}}(\cdot)$ being defined to be the mean over all states in the target space T and over all instances in the set of problem instances $\mathcal{I}$. No subscript is used if the sample space is clear from context. Further note that $\overline{\#_d(k)}$ is the sample mean of T where T is sampled from the sample space

of all problem instances $\mathcal{I}$. Therefore, $E_{\mathcal{I}}\left(\overline{\#_d(k)}\right) = E_T(\#_d(k))$ as the sample mean $\overline{\#_d(k)}$ is an unbiased estimator of the expected value $E_T(\#_d(k))$. The same also holds for $\overline{\#_{d_1}(k)\#_{d_1}(k)}$. To calculate $E_T(\#_d(k))$, we need to know the distributions of $\{\#_d(k)\}_{d=0}^{n}$. To compute $E_{\mathcal{I}}\left(\overline{\#_{d_1}(k)\#_{d_2}(k)}\right) = E_T(\#_{d_1}(k)\#_{d_2}(k))$ we use that $E_T(\#_{d_1}(k)\#_{d_2}(k)) = E_T(\#_{d_1}(k))E_T(\#_{d_2}(k)) + \mathrm{Cov}(\#_{d_1}(k), \#_{d_2}(k))$. For $\mathrm{Cov}(\#_{d_1}(k), \#_{d_2}(k))$, the joint probability distribution of $\{(\#_{d_1}(k), \#_{d_2}(k))\}_{d_1,d_2=0}^{n}$ is required.

As T is uniformly sampled from the complete state space, this basically leads to an urn model without replacement. Therefore, except for some edge cases, the random variables $\#_d(k)$ can be described by a Hypergeometric distribution, see Lemma 3.

Lemma 3. *In case of uniform target sampling the probabilities $P(\#_d(k) = x)$ and $P(\#_{d_1}(k) = x_1, \#_{d_2}(k) = x_2)$ are defined as follows:*

$$P(\#_d(k) = x) = \begin{cases} 1 & d = 0 \wedge x = 1 \\ 0 & d = 0 \wedge x \neq 1 \\ \mathrm{HypG}(\boldsymbol{X}, \boldsymbol{x}) & \textit{otherwise} \end{cases} \tag{21}$$

with $\boldsymbol{X} = \left(\binom{n}{d}, 2^n - 1 - \binom{n}{d}\right)$ and $\boldsymbol{x} = (x, |T| - 1 - x)$. Furthermore,

$$P(\#_{d_1}(k) = x_1, \#_{d_2}(k) = x_2) = \begin{cases} P(\#_{d_1}(k) = x_1)P(\#_{d_2}(k) = x_2) & d_1 = 0 \vee d_2 = 0 \\ P(\#_{d_1}(k) = x_1) & d_1 = d_2 \wedge x_1 = x_2 \\ 0 & d_1 = d_2 \wedge x_1 \neq x_2 \\ \mathrm{HypG}(\boldsymbol{X}, \boldsymbol{x}) & \textit{otherwise} \end{cases} \tag{22}$$

with $\boldsymbol{X} = \left(\binom{n}{d_1}, \binom{n}{d_2}, 2^n - 1 - \binom{n}{d_1} - \binom{n}{d_2}\right)$ and $\boldsymbol{x} = (x_1, x_1, |T| - 1 - x_1 - x_2)$.

Proof. This proof is structured in two parts. We start by showing the correctness of Eq. (21), which in turn necessitates distinguishing between two cases. As $\#_d(k)$ describes the distance relationship of a random state in T to all other states in T (including itself), $\#_0(k)$ is always 1, from which the first two cases of Eq. (21) follow. For $d \geq 0$, we need to consider the sampling process of T. Assume we start with just one random state in T, and then sample the rest. This is described by an urn model with $X_1 = \binom{n}{d}$ balls of colour d, and $X_2 = 2^n - 1 - \binom{n}{d}$ other balls. Note that by picking a random reference state in T, there are only $2^n - 1$ balls left in the urn to draw the $|T| - 1$ states left to fill the target space. Therefore, the probability of sampling x states of Hamming distance d from the random initial state is given by

$$P(\#_d(k) = x) = \mathrm{HypG}\big((X_1, X_2), (k, |T| - 1 - x)\big),$$

which concludes the proof of Eq. (21).

We need to consider three cases for Eq. (22). If

$d_1 = 0$, the only possible samples (x_1, x_2) must have $x_1 = 1$, with the same reasoning as above. Thus, $P(\#_{d_1}(k) = x_1, \#_{d_2}(k) = x_2) = \delta_{1,x_1}P(\#_{d_2}(k) = x_2)$, where $\delta_{i,j}$ is the Kronecker delta function. Further note that $\delta_{1,x_1} = P(\#_{d_1}(k) = x_1)$. The case of a vanishing second argument, $d_2 = 0$, is symmetric to $d_1 = 0$. Therefore, if follows that $P(\#_{d_1}(k) = x_1, \#_{d_2}(k) = x_2) = P(\#_{d_1}(k) = x_1)P(\#_{d_2}(k) = x_2)$ iff $d_1 = 0$ or $d_2 = 0$. If $d_1 = d_2$, then obviously $x_1 = x_2$ needs to be satisfied, in which case $P(\#_{d_1}(k) = x_1, \#_{d_2}(k) = x_2)$ reduces to $P(\#_{d_1}(k) = x_1) = P(\#_{d_2}(k) = x_2)$. For all other d_1 and d_2, re-consider the sampling process from an urn without replacement: This time, we are interested in two colours d_1 and d_2. Consequently, the urn contains $\binom{n}{d_1}$ balls of colour d_1, $\binom{n}{d_2}$ balls of colour d_2, and $2^n - 1 - \binom{n}{d_1} - \binom{n}{d_2}$ other balls. The probability of obtaining k_1 states of Hamming distance d_1 and k_2 states of Hamming distance

d_2 in a sample of size $|T|$ is therefore given by

$$P(\#_{d_1}(k) = x_1, \#_{d_2}(k) = x_2) = \text{HypG}(\boldsymbol{X}, \boldsymbol{x})$$

with $\boldsymbol{X} = \left(\binom{n}{d_1}, \binom{n}{d_2}, 2^n - 1 - \binom{n}{d_1} - \binom{n}{d_2}\right)$ and $\boldsymbol{x} = (x_1, x_2, |T| - 1 - x_1 - x_2)$. □

Now that we know the distribution of our random variables $\#_d(k)$, we can deduce properties like expected values and covariances. In Lemma 4, we use Lemma 3 to determine $E_T(\#_d(k))$ and $\text{Cov}(\#_{d_1}(k), \#_{d_2}(k))$.

Lemma 4.

$$E_T(\#_d(k)) = \begin{cases} 1 & d = 0 \\ |T| \frac{\binom{n}{d}}{2^n} & \textit{otherwise} \end{cases} \tag{23}$$

$$\text{Cov}(\#_{d_1}(k), \#_{d_2}(k)) =$$
$$\begin{cases} 0 & d_1 = 0 \vee d_2 = 0 \\ |T| \frac{2^n - |T|}{2^n - 1} \frac{\binom{n}{d}}{2^n}\left(1 - \frac{\binom{n}{d}}{2^n}\right) & d_1 = d_2 = d > 0 \\ -|T| \frac{2^n - |T|}{2^n - 1} \frac{\binom{n}{d_1}\binom{n}{d_2}}{2^{2n}} & \textit{otherwise} \end{cases} \tag{24}$$

Proof. In large, this proof follows the structure of the proof of Lemma 3. We start by show the correctness of Eq. (23). If $d = 0$, we trivially have $E(\#_0) = 0P(\#_0 = 0) + 1P(\#_0 = 1) = 1$. For all $d > 0$, $P(\#_d(k) = x)$ is described by the Hypergeometric distribution as in the second case of Eq. (21), with an expected value of $|T|\binom{n}{d}2^{-n}$ as required.

We now prove Eq. (24). Assume that $d_1 = 0$, then with $\#_{d_1 d_2}(k) := \#_{d_1}(k)\#_{d_2}(k)$ we have

$$E(\#_{d_1 d_2}(k)) = \sum_k 0 \cdot \#_{d_2}(k) P(\#_{d_1}(k) = 0, \#_{d_2}(k) = x) +$$
$$\sum_k 1 \cdot \#_{d_2}(k) P(\#_{d_1}(k) = 1, \#_{d_2}(k) = x)$$
$$= \sum_k \#_{d_2}(k) P(\#_{d_1}(k) = 1, \#_{d_2}(k) = x)$$

and of course $P(\#_{d_1}(k) = 1) = 1$ for $d_1 = 0$ and thus $E(\#_{d_1}(k)\#_{d_2}(k)) = \sum_k \#_{d_2}(k)P(\#_{d_2}(k) = k) = E(\#_{d_2}(k))$. Together with Eq. (23), it follows that

$$\text{Cov}(\#_{d_1}(k), \#_{d_2}(k)) = E(\#_{d_1}(k)\#_{d_2}(k)) - E(\#_{d_1}(k))E(\#_{d_2}(k))$$
$$= E(\#_{d_2}(k)) - E(\#_{d_2}(k)) = 0.$$

The case for $d_2 = 0$ is symmetric. We conclude that $\text{Cov}(\#_{d_1}(k), \#_{d_2}(k)) = 0$ iff $d_1 = 0$ or $d_2 = 0$. Note that $\text{Cov}(\#_d(k), \#_d(k)) = \text{Var}(\#_d(k))$, which is given by

$$\text{Var}(\#_d(k)) = |T| \frac{2^n - |T|}{2^n - 1} \frac{\binom{n}{d}}{2^n}\left(1 - \frac{\binom{n}{d}}{2^n}\right)$$

since the sampling process is described by a multivariate hypergeometric distribution as shown in Lemma 3. The third and second case also directly follow from the covariance of the multivariate hypergeometric distribution from Lemma 3. □

With the two lemmas above, we can efficiently calculate $\tilde{E}(F_1)$. This demonstrates a use case that benefits from our approximation theorem: Given a problem for which the target space structure with respect to Hamming distances between targets can be modelled analytically, the expected optimisation landscape can be approximately derived solely from this model. While instance specific analytic formulations of F_1 are known for the Ising model problem [67] from a physics-centric point of view, our approach accesses the problem from a previously unexplored angle that benefits from insights from theoretical computer science. We also conclude that an efficient description of such models allows for an efficient approximation of a QAOA optimisation landscape. Further investigations into the complexity theoretical implications of this could provide insights into the link between the structure target spaces, the complexity of problems and the computational power of QAOA, albeit we need to leave these questions to future research, as our primary goal in this paper is to establish the framework and derive direct practical utility.

2. Sampling Approach

Approximating $E(F_1)$ with an efficient theoretical model of the target space is obviously the preferred way to address a given problem, if feasible analytically. However, it is also possible to determine $\left\{E\left(\overline{\#_d(k)}\right)\right\}_{d=0}^n$ and $\left\{E\left(\overline{\#_{d_1}(k)\#_{d_2}(k)}\right)\right\}_{d_1, d_2 = 0}^n$ empirically by sampling from a set of problem instances. Above, we described the problem instances of the uniform sampling example by giving a sampling routine for a random target space. Recall that this defines problem instances as random variables. If we now want to determine the expected values and covariance matrix empirically, we need a concrete sample of instances. Which in this example will be a set of realisations of problem random variables.

In our experiments, we sampled 500 random instances for each state space dimension from $8 \leq n \leq 11$. The upper bound of $n = 11$ keeps computational cost of explicitly evaluating F_1 at a reasonable level, whereas the lower bound $n = 8$ ensures a sufficiently large state space. For the scope of this paper, this range is sufficient to show the effects of increasing the state space dimension. Every instance has a target set of 2^{n-1} states, making it scale with the size of the whole state space. As defined above, $\overline{\#_d(k)} = \frac{1}{|T|}\sum_{k \in T} \#_d(k)$. Let $\mathcal{T}$ be the set of target sets of all sampled instances. Then, empirically determining $E\left(\overline{\#_d(k)}\right)$ means calculating the mean of $\overline{\#_d(k)}$ over the set $\mathcal{T}$ of all sampled target sets. To clarify over which set a mean is calculated, we introduce the

notation $\overline{f(x)}^A = \frac{1}{|A|}\sum_{x \in A} f(x)$. Then,

$$\overline{\overline{\#_d(k)}^T}^{\mathcal{T}} = \frac{1}{|\mathcal{T}|}\sum_{T \in \mathcal{T}}\frac{1}{|T|}\sum_{k \in T}\#_d(k).$$

The same holds for $E\left(\overline{\#_{d_1}(k)\#_{d_2}(k)}\right)$, with

$$\overline{\overline{\#_{d_1}(k)\#_{d_2}(k)}^T}^{\mathcal{T}} = \frac{1}{|\mathcal{T}|}\sum_{T \in \mathcal{T}}\frac{1}{|T|}\sum_{k \in T}\#_{d_1}(k)\#_{d_2}(k).$$

3. Comparison

As we can see in Fig. 5, both approaches lead to approximations that fit the expected value of F_1 exceptionally well. In all our experiments we used the sample mean (over all instances) as an estimator for the expected landscape $E(F_1)$. We calculated the mean of F_1 by evaluating $F_1(\gamma, \beta)$ for every instance at 100 sample points for $0 \leq \beta \leq \pi$ at the cross section for $\gamma = 1.2$. This shows the versatility of our approximation: Theoretical models can be used to get excellent results. This is not always feasible for more complex problems where the inherent structural properties of the target space are not as straightforward. However, we also showed that an empirical approach leads to equally excellent results in such cases. We therefore argue that our approximation theorem can be a useful tool that is equally well suited for theoretical considerations and empirical analyses.

B. Clustered Sampling

The uniform sampling example was chosen deliberately trivial to showcase the analytical approach. As a consequence, one important aspect of more realistic problems remains unaddressed: Owing to the uniform sampling process, the states in T are independent of each other. This is, in general, not the case for real-world problems. SAT instances, for example, often have clustered solution spaces. Industrial SAT instances exhibit community-like structures in their solution spaces [76]. But contrary to intuition, even uniformly sampled SAT instances possess solution structures [74]. Inspired by this observation, we conduct further experiments with a random clustered sampling process: Three cluster seeds are sampled uniform at random from the complete state space. Then, per seed, 30 new states are added to the target set. Each state is reached by a random walk starting from its corresponding seed by flipping a random bit each step, with a step probability of ½. Note that reaching a basis state $|k\rangle$ from another basis state $|z\rangle$, with $k, z \in \mathbb{F}_2^n$ means that qubit l was flipped if and only if $k_l \neq z_l$. The process of empirically determining $\left\{E\left(\overline{\#_d(k)}\right)\right\}_{d=0}^n$ and

$\left\{E\left(\overline{\#_{d_1}(k)\#_{d_2}(k)}\right)\right\}_{d_1,d_2=0}^n$ stays the same as in the uniform example above.

Figure 6 shows the empirical estimate of $E(F_1)$ compared to the actual mean value $\overline{F_1}$ and the introduced absolute error. As can be seen, we clearly introduce some error by approximating. But although the variance of F_1 drops significantly with higher dimensions, the absolute error never surpasses the standard deviation of F_1. Also note how the whole co-domain of F_1 is compressed. This is a direct consequence of the scaling coefficient $|T|/2^n$ in Eq. (11) as we chose to generate fixed dimension independent sized target sets. Recall that $E(F_1)$ describes the result probability of sampling a solution state from $|\gamma, \beta\rangle_1$, which unveils an interesting relation between the size of the solution space and the solution sample probability.

C. Boolean Satisfiability

After two artificially constructed sampling examples, it is time to consider our first real problem. While SAT is one of the cornerstones of **NP**-complete problems in theoretical computer science, it also has considerable practical impact [83], and has also been intensively studied in the quantum domain [84–88].

Most importantly, it is known to often exhibit structured solution spaces [75], and is a natural combinatorial decision problem, which makes it perfectly suited for our framework.

Note that in both above sampling examples, we only discussed the target set and completely disregarded any actual realisation of the QAOA circuit in our analysis. However, if the complete target set were known, it would be trivial to construct a constraint Hamiltonian $C = \prod_{k \in T}|k\rangle\langle k|$ by simply projecting onto every target state. This is obviously not an adequate approach for hard computational problems, as full knowledge about the solution space cannot be expected. Unfortunately, explicitly projecting onto every state in T violates this requirement for difficult problems. We therefore show now that we still can apply our approximation theorem even if this requirement is added to the picture.

Let $\mathcal{C} = \{c_i\}_{i=1}^m$ a Boolean formula in conjunctive normal form, with c_i being the i-th clause. The SAT problem asks whether there exists a Boolean variable assignment such that all clauses are satisfied. Each clause is a disjunction of literals. A literal is a possibly negated Boolean variable. Let's consider the clause $c = x_1 \vee x_3 \vee \overline{x_8}$. Then c is satisfied by all possible assignments except for the characteristic unsatisfying assignment $[x_1 \mapsto 0, x_3 \mapsto 0, x_8 \mapsto 1]$. Let $\overline{C_i}$ be a projector onto this characteristic unsatisfying assignment of the clause c_i. Given $\{\overline{C_i}\}_{i=0}^m$ we can construct a constraint Hamiltonian $C = \prod_{i=0}^m(\mathbb{1} - \overline{C_i})$ that projects onto all satisfying assignments of $\mathcal{C}$. This allows us to just focus on the solution space, and set the stage for using the approximation theorem.
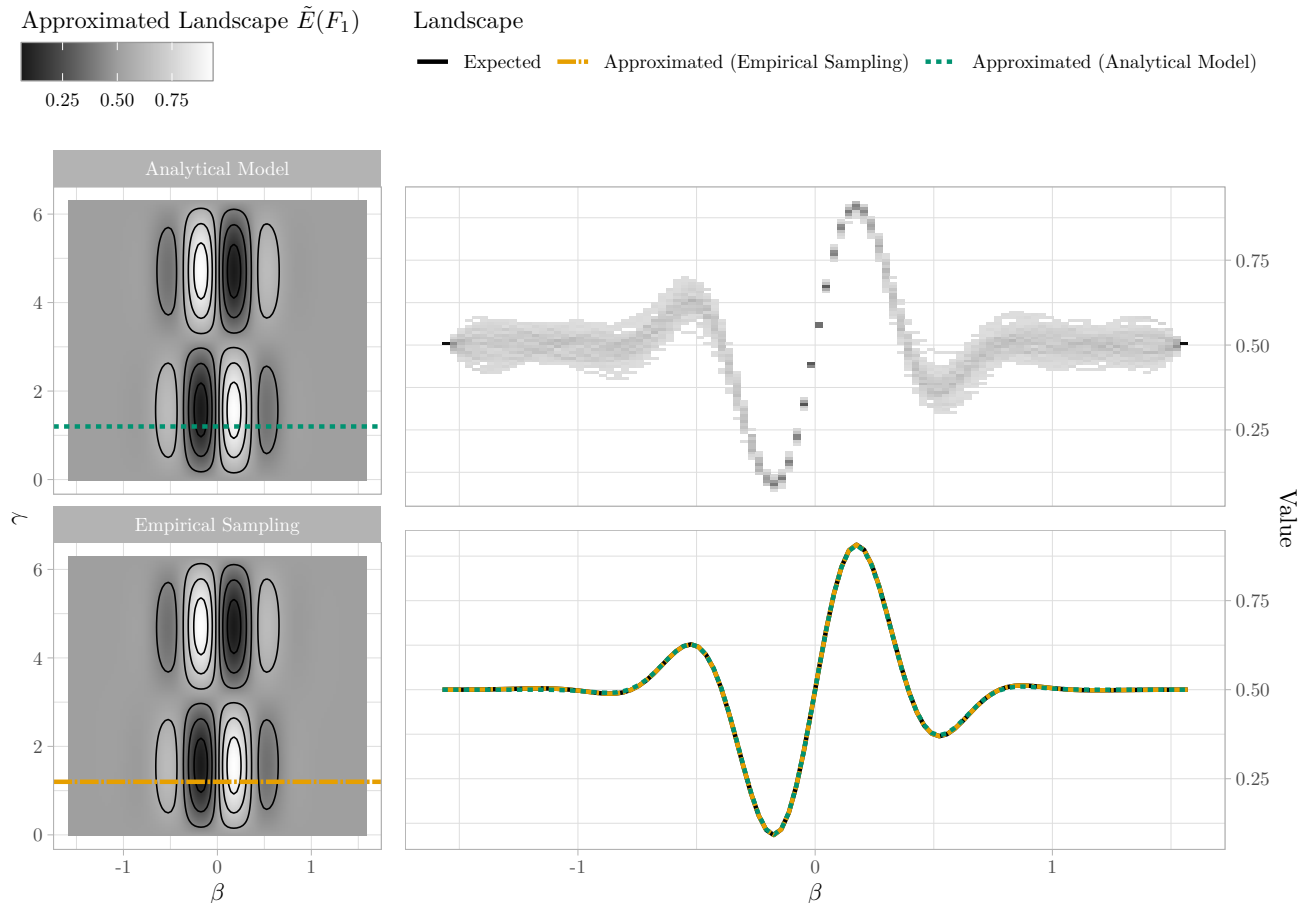
For our experiments, we generated 500 random satisfi-

FIG. 5. Comparison of analytic and sampling-based landscape approximation for the uniform sampling experiment. *Left:* Landscapes for sampling methods. The difference between the analytically modelled and the empirically sampled landscape approximations is negligibly small. *Top Right:* A two-dimensional heat-map ▪ (grey) that illustrates the vertical distribution of $F_1(\beta, \gamma_0 = 1.2)$ values of 500 random instances sampled as described in Section V A 2. The heat-map is calculated over a grid of 100×100 bins, where each column collects all F_1 values in the corresponding β interval and sums up to 1. *Bottom Right:* Cross-section through the above landscape at $\gamma_c = 1.2$. The expected landscape estimated by the mean value of the data points displayed in the heat map above ▬ (black line) and two approximations of F_1, where $\left\{ \overline{\#_d(k)} \right\}_{d=0}^{n}$ and $\left\{ (\overline{\#_{d_1}(k)}, \overline{\#_{d_2}(k)}) \right\}_{d_1, d_2 = 0}^{n}$ are determined empirically ▬ (orange line) and modelled theoretically ▬ (green line) are shown. Again, exact and approximate results are in excellent agreement, with little variation.

able SAT instances for each $n \in \{8, 9, 10, 11\}$, where the number of variables directly translates to the dimension ($|V| = n$) and the number of clauses is always $|C| = 4n$. Each clause has three literals that are negated with probability ½. Glucose 4.2 [89] was used to enumerate all satisfying assignments for each instance. In contrast to the previous sampling examples, where we always sampled a target set of fixed size $|T|$, the size of T depends on the instance and its number of satisfying assignments in this scenario. Therefore, we also have to determine $|T|$ empirically.

In Fig. 7 we see once again how well our approximation fits the real mean values for F_1. Note that compared to the examples above, there seems to be a unusual amount of variance on the y-axis. This is a result of different

target set sizes $|T|$ across instances, which causes global vertical shifts of function F_1. If one is just interested in the overall structure of F_1 one could consider ignoring the scaling factor induced by $|T|$ altogether.

D. k-CLIQUE

An efficient in-place projection like the SAT constraint Hamiltonian can not always be expected. So what if ancilla bits are needed? In this example, we demonstrate an approach following Theorem 2 by showing how to construct an ancilla register independent of the constraint Hamiltonian for k-CLIQUE.

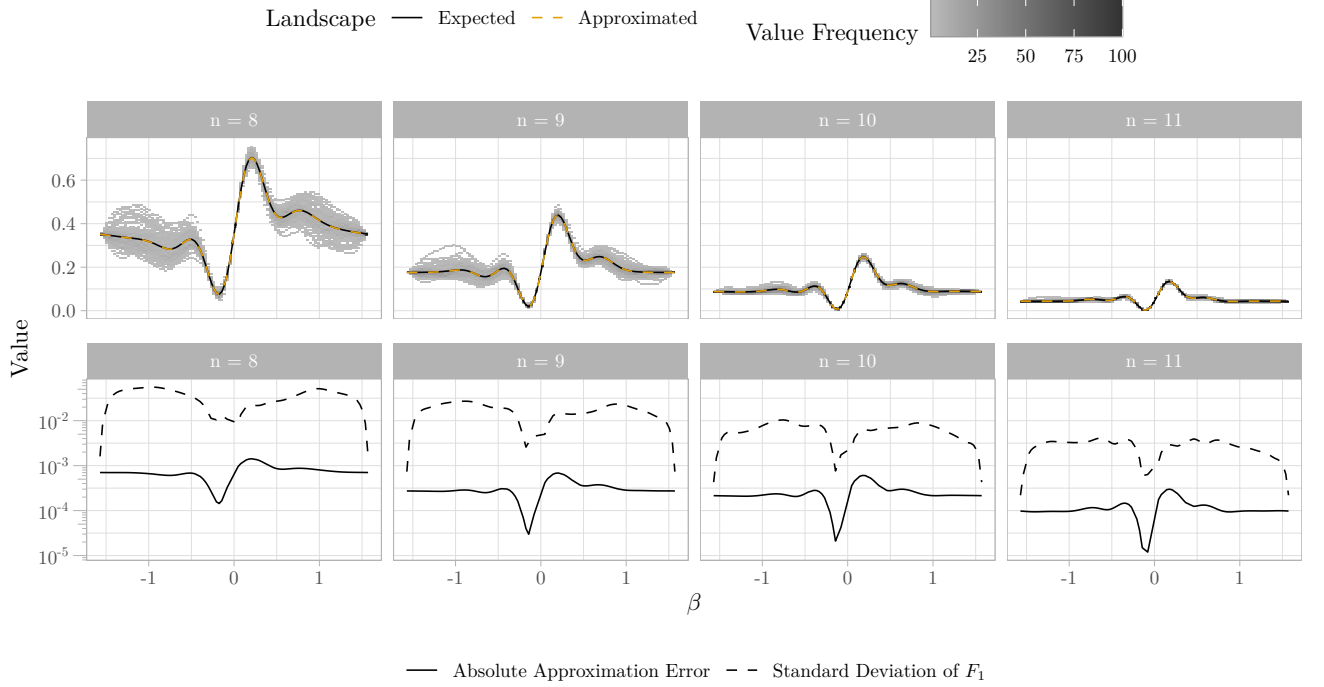**Definition 5.** *Given a graph G and $k \in \mathbb{N}$, then the*

FIG. 6. Landscape cross-sections and approximation quality for clustered sampling for different state space dimensions in columns. *Top:* Two-dimensional heat-map of sampled values for $F_1(\beta, \gamma_c)$ at $\gamma_c = 1.2$ and across $0 \leq \beta \leq \pi$, overlaid with the expected landscape estimated by the mean value of the sampled data points $\overline{F_1}$ ▬ (black line) and the approximated expectation $\tilde{E}(F_1)$ ▬ (orange line). *Bottom:* Absolute approximation error (solid line) and standard deviation of F_1 (dashed line). The data demonstrate a small approximation error that lies strictly and considerably below the standard deviation of F_1.

*decision problem whether G has a clique (*i.e., *a fully connected sub-graph) of size k is called the k-Clique Problem.*

To solve k-Clique with QAOA, we first need to define the state space. Let $G = (V, E)$ be a graph with n vertices, then we have a n-dimensional state space as we map each vertex to a specific qubit. A basis state $|z\rangle$ with $z \in \mathbb{F}_2^n$ marks vertices such that a vertex v_i is marked iff $z_i = 1$. Then, $|z\rangle$ represents a valid solution to k-Clique iff $K = \{v_i \in V \mid z_i = 1\}$ is a clique in G and $|K| = k$.

Let $\overline{G}$ be the complement of G and (i,j) an edge in $\overline{G}$. Then,

$$C_{\text{cliques}} = \prod_{(i,j) \in \overline{G}} (\mathbb{1} - P_{ij})$$

with $P_{ij} = \mathbb{1}_{0,i-1} \otimes |1\rangle\langle 1| \otimes \mathbb{1}_{i+1,j-1} \otimes |1\rangle\langle 1| \otimes {}_{j+1,n}$ (25)

projects onto all cliques in G. However, this also includes cliques of sizes different from k, such as trivial cliques as single vertices and edges. A second step is needed to filter out the k-cliques. For this we have to define an unitary operator D_H that calculates the Hamming weight of $|z\rangle$. This is done by writing $d_H(z)$ on an ancilla register but note that $|z\rangle|y\rangle \mapsto |z\rangle|d_H(z)\rangle$ is only invertible for a fixed y (*e.g.*, $y = 0$), which conflicts with D_H being unitary. Thus, we define $D_H : \mathcal{B}^{\otimes n} \otimes \mathcal{B}^{\otimes \lceil \text{ld}(n) \rceil} \rightarrow$

$\mathcal{B}^{\otimes n} \otimes \mathcal{B}^{\otimes \lceil \text{ld}(n) \rceil}$ as $D_H : |z\rangle|y\rangle \mapsto |z\rangle|y + d_H(z) \mod 2^m\rangle$ with $m := \lceil \text{ld}(n) \rceil + 1$. Since $\mathbb{N}/m\mathbb{N}$ forms a group under addition, there exists a unique inverse element for each $d_H(z)$ which ensures D_H to be invertible. Our construction is illustrated in Fig. 8.

With D_H we can construct a second constraint Hamiltonian $C_{d_H=k} = D_H^\dagger (\mathbb{1}^{\otimes n} \otimes |k\rangle\langle k|) D_H$ that projects onto the space spanned by $\{|z\rangle|y\rangle \mid y + d_H(z) \mod 2^m = k\}$. Therefore, if the ancilla register is initialised to $|0\rangle$ the application of $C_{d_H=k}$ projects onto states with Hamming distance k. Applying both projectors results in

$$C := C_{d_H=k}\Big(C_{\text{cliques}} \otimes \mathbb{1}^{\otimes \lceil \text{ld}(n) \rceil}\Big) \quad \text{with}$$

$$C|z\rangle|\mathbf{0}\rangle = \begin{cases} |z\rangle|\mathbf{0}\rangle & z \text{ is k-clique in } G \\ 0 & \text{otherwise} \end{cases} \quad (26)$$

This results in a QAOA state $|\beta, \gamma\rangle_1 = e^{-i\beta X^{\otimes n}} e^{-i\gamma C} \frac{1}{\sqrt{2^n}} |+\rangle^{\otimes n} |\mathbf{0}\rangle$ after the phase separation introduced by $e^{-i\gamma C}$ the state evolves to $\frac{1}{\sqrt{2^n}} e^{-i\beta X^{\otimes n}} \big(e^{-i\gamma} \sum_{z \in K} |z\rangle|\mathbf{0}\rangle + \sum_{z \notin K} |z\rangle|\mathbf{0}\rangle \big)$ finally, after factoring out the ancilla register and applying the
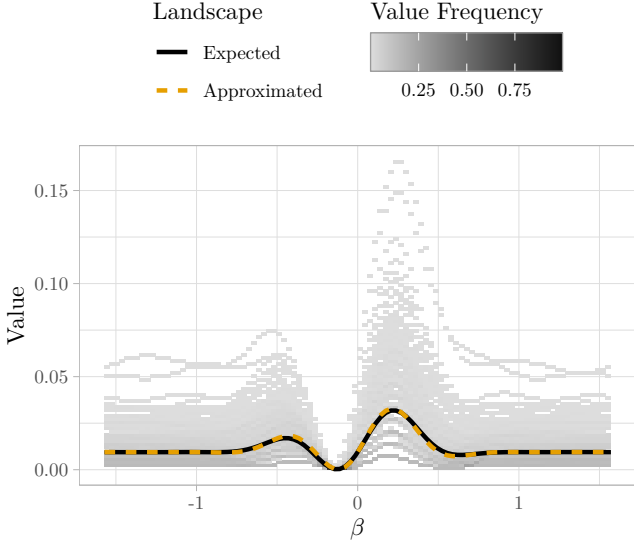
FIG. 7. Approximate expected value $\tilde{E}(F_1)$ of $F_1(\beta, \gamma)$ for a random SAT instance perfectly fits the actual expected landscape $E(F_1)$ over all sampled instances, even though SAT induces a higher variance in the values of F_1. This is illustrated by a cross section at $\gamma = \gamma_0 = 1.2$. The two-dimensional heatmap ▪ (gray) illustrates the vertical distribution of $F_1(\beta, \gamma_0)$ over a grid of 100×100 bins, where each column collects all F_1 values in the corresponding β interval and sums up to 1. This histogram is overlaid by the expected Landscape $E(F_1(\beta, \gamma_0))$ ▬ (black, solid) and the approximated mean value $\tilde{E}(F_1(\beta, \gamma_0))$ ▬ (orange, dashed) obtained from the approximation theorem. All three quantities are in very good agreement.

mixer we end up with

$$\frac{1}{\sqrt{2^n}} \left(e^{-i\gamma} \sum_{z \in K} \sum_{x \in \mathbb{F}_2^n} f(\beta, z, x)|x\rangle + \right.$$

$$\left. \sum_{z \notin K} \sum_{x \in \mathbb{F}_2^n} f(\beta, z, x)|x\rangle \right) \otimes |\mathbf{0}\rangle$$

Note that this QAOA circuit is invariant on the ancilla register, which allows us to trace the ancilla qubits out. Then, $|\beta, \gamma\rangle_1$ is identical to Eq. (13). The ancilla qubits have no influence on the distribution of Hamming weights. Therefore, they can be ignored and once again we only have to reason about the problem specific solution space.

E. One-Way Functions

Theorem 1 is especially useful if we either have a theoretical understanding of the solution space of a problem or if the solution space can be efficiently sampled. This is the case for one-way functions: Instances can be easily generated by first picking a state from solution space and then generating the original input instance by applying

the inverse of the one-way function, as it is easy to compute by definition. We will showcase this scenario with qr-FACTORING.

Definition 6. *Given a value $x = qr$ with $q, r \in \mathbb{P}$, qr-FACTORING is the problem of computing q and r given x.*

$$f_{qr} : \{qr \mid q, r \in \mathbb{P}\} \to \mathbb{P} \times \mathbb{P}, \quad f_{qr} : qr \mapsto (q, r) \quad (27)$$

Obviously the inverse of Eq. (27) is simply the integer multiplication $f_{qr}^{-1}(q, r) = q \cdot r$. So, given a pre-computed set of primes $\mathcal{P} \subset \mathbb{P}$, we can efficiently sample a pool of qr-FACTORING instances by $f_{qr}^{-1}(\{(q, r)_i\}_{i=1}^m)$ with $(q, r)_i$ being sampled from $\mathcal{P} \times \mathcal{P}$ at uniform random for all $0 \le i \le m$. In fact, as we argued in Section V D, the target space T without ancillary qubits fully suffices for our approximate analysis. For qr-FACTORING, a solution (q, r) can be represented in T as follows: Let $x_{(2)}$ be the binary representation of $x \in \mathbb{N}$, and let $z \in \mathbb{F}_2^k$ with $m \le n$. Then $p_n(z)$ is a padding of z with $n - k$ leading zeros. Now, $\{(q, r)_i\}_{i=1}^m$ is mapped to T by $(q, r) \mapsto p_n(q_{(2)}) \circ p_n(r_{(2)})$ with $n = \max\{\lceil \mathrm{ld}(q) \rceil, \lceil \mathrm{ld}(r) \rceil\}$, where $\circ$ denotes concatenation of two bit strings.

With this mapping we performed a series of experiments for different target space dimensions $n \in \{12, 14, 16, 18\}$, where we sampled $\{(q, r)_i\}_{i=1}^{100}$ from $\mathcal{P}_{\frac{n}{2}} \times \mathcal{P}_{\frac{n}{2}}$, with $\mathcal{P}_{\frac{n}{2}} := \{q \in \mathbb{P} \mid q < 2^{\frac{n}{2}}\}$. Again, $\left\{E(\overline{\#_d(k)})\right\}_{d=0}^n$ and $\left\{E\left(\overline{\#_{d_1}(k)\#_{d_2}(k)}\right)\right\}_{d_1, d_2 = 0}^n$ were empirically determined as described above. The cardinality of the target set is $|T| = 2$, as $f_{qr}^{-1}(q, r) = f_{qr}^{-1}(r, q)$ and thus for each $x = qr$, solution and target space each contain exactly two elements. Figure 9 shows the approximated and actual mean of F_1 as well as the absolute approximation error for $n = 18$. Although the margin for error apparently is extremely tight for qr-FACTORING, our approximation nicely lies on top of the actual mean. Again, the absolute error remains below the standard deviation. As is likewise visible in Fig. 9, qr-FACTORING exhibits only small deviations between instances in their optimisation landscapes. But even with this small margin of error, the approximation tightly fits the real distribution, and the error is bounded by the standard deviation of F_1.

VI. PRACTICAL UTILITY

It is well known that the classical task of optimising parameters in QAOA requires substantial amounts of computational effort: This aspect of the heuristic is **NP**-hard [32] even for classically tractable systems; likewise, every polynomial time algorithm is susceptible to instances for which the relative resulting error can be arbitrarily large, thus rendering approximate approaches likewise troublesome. The QAOA quantum circuit, in addition, usually needs to be evaluated for many different sets of angles to gain the necessary information on the
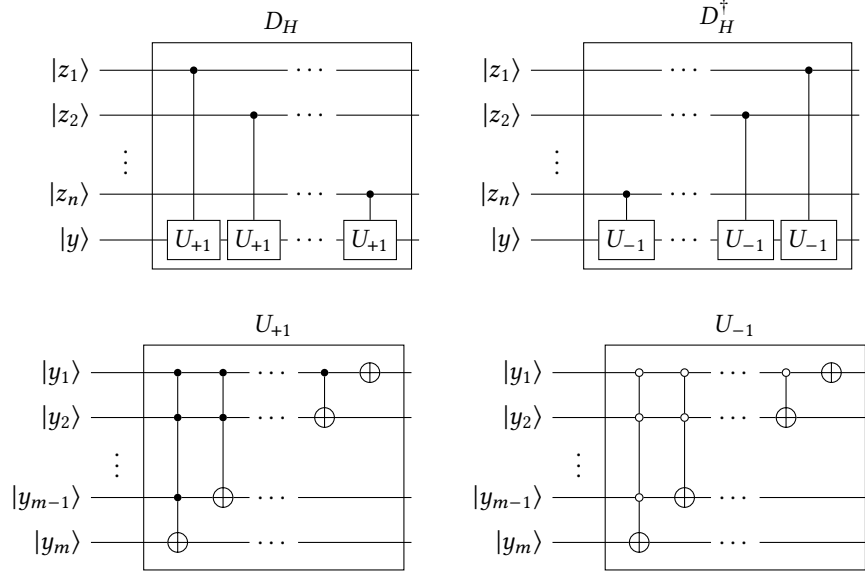
FIG. 8. Circuits for Hamming weight computation. D_H is used to construct the constraint Hamiltonian for the k-Clique problem. It is used to check whether a potential clique is of size k or not. For this D_H adds the Hamming weight of register $|\boldsymbol{z}\rangle$ to another register $|\boldsymbol{y}\rangle$, while $D_H^\dagger$ subtracts it from $|\boldsymbol{y}\rangle$. State $|\boldsymbol{y}\rangle$ is initialised with $\boldsymbol{y} = 0$ and holds the Hamming weight of $\boldsymbol{z}$ encoded as a bit string, after the application of D_H. The D_H gate adds the Hamming weight to $|\boldsymbol{y}\rangle$ by conditionally applying U_{+1} to it for each qubit in $|\boldsymbol{z}>$, with $U_{+1}|\boldsymbol{y}\rangle = |\boldsymbol{y} + 1\rangle$. The inverse operation $D_H^\dagger$ subtracts the Hamming weight of $\boldsymbol{z}$ from $|\boldsymbol{y}\rangle$, by conditionally applying U_{-1} for each qubit, with $U_{-1}|\boldsymbol{y}\rangle = |\boldsymbol{y} - 1\rangle$.

optimisation landscape that is required as input for the classical parameter optimisation routine, albeit efforts differ depending on the concrete choice [6, 64, 90]

Our approximation approach allows us to separate instance sampling from optimisation landscape sampling. Even at a fixed point (β, γ), directly sampling $F_1(\beta, \gamma)$ is intractable for exact computation, as it contains a sum over exponentially many terms. Likewise, determining an expectation value $E(F_1)$ requires to consider a substantial amount of points (β, γ) and instances. However, given structural information about the target space in form of $\left\{ E\left(\overline{\#_d(k)}\right) \right\}_{d=0}^n$ and $\left\{ E\left(\overline{\#_{d_1}(k)\#_{d_2}(k)}\right) \right\}_{d_1,d_2=0}^n$, only a sum over linear many terms has to be evaluated to approximate $E(F_1)$ efficiently at a point (β, γ). The inputs $\left\{ E\left(\overline{\#_d(k)}\right) \right\}_{d=0}^n$ and $\left\{ E\left(\overline{\#_{d_1}(k)\#_{d_2}(k)}\right) \right\}_{d_1,d_2=0}^n$ of our approximation method can reasonably be obtained empirically by statistical methods, or even theoretically modelled as demonstrated in Section V A.

This separation enables a different approach to QAOA that does not require an unbounded iteration involving the quantum circuit. It starts by sampling (partial) target spaces of random instances, from which the structural metrics needed as input for Theorem 1 are gathered. Then, a classical optimiser is utilised to determine the optimal QAOA angles with regard to the approximate optimisation landscape, resulting from the previously sampled target spaces. Because we used the expected landscape of a random instance of the problem at hand, the found parameters apply to the complete problem. Thus, param-

eter optimisation only needs to be performed once per problem to find one single set of parameters for all instances. Finally, the resulting angles are used to initialise a QAOA circuit from which potential problem solutions will be sampled on quantum hardware. Following this approach, only the final sampling step needs to be performed on real quantum hardware. The standard QAOA procedure is in fact more of a heuristic than a quantum algorithm, where we have to optimise a quantum circuit for each instance. In our approach to QAOA we optimise the parameters on a error bounded approximation of the expected landscape. Thus, we end up with one circuit for all problem instances. This mathematically sound splitting of computation into a problem-global and instance-specific phase significantly moves QAOA towards the realm of true quantum algorithms, in stark contrast to empirically motivated heuristics. Recall Fig. 2, where we illustrated the difference between both approaches.

We explored this approach empirically (using numerical simulations; see Appendix A for details about our reproduction package that allows researchers to directly employ our approach or inspect our implementation) for the examples introduced above in Section V. For every example, we randomly generated 50 instances to be solved with both QAOA methods: standard QAOA and non-iterative QAOA. The approximate optimisation landscape for the non-iterative QAOA was calculated based on the dataset generated earlier in Section V. After optimisation, we visually verified each set of parameters for non-iterative QAOA resides on one of the two main extrema of the land-
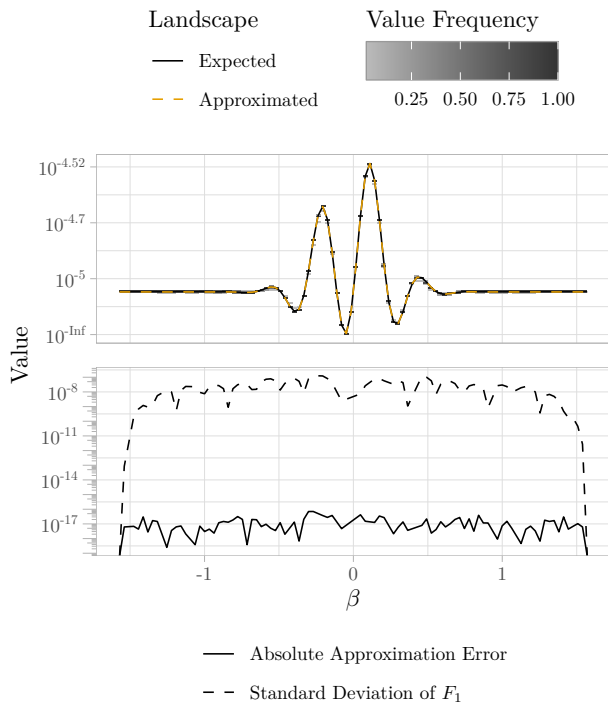
FIG. 9. *Top:* 2D histogram of the F_1 values for all instances for $n = 18$ from the qr-Factoring example, overlaid by the approximated landscape $\tilde{E}(F_1)$ ▬ (orange) and the expected landscape $E(F_1)$ ▬ (black). *Bottom:* Absolute approximation error (solid line) compared to the standard deviation of F_1 (dashed line). The variance in F_1 is negligible, which makes the margin of error for approximations extremely tight. Our approximation provides excellent results even under such challenging conditions.

scape. After that, the parameters were used to create a QAOA circuit for each problem instance. This circuit then was compared with a QAOA circuit trained on the actual instance. From both circuits, we sampled 50 potential solutions per instance (100 states were sampled per instance for qr-Factoring to more accurately capture the extremely low maximal possible success probability). Fig. 13 shows the resulting approximate landscapes.

As for practical performance, we can see from Fig. 10 that parameters obtained by standard QAOA are mostly located around extrema of the approximate optimisation landscapes. However, the optimisation apparently also often ends in local maxima, especially for qr-Factoring. This suggests matching success rates for standard and non-iterative QAOA. Indeed, we can see in Fig. 11 that states sampled from non-iterative QAOA are valid solutions with identical (or slightly better) probability than states sampled from standard QAOA. Hard combinatorial constraint satisfaction problems are known to have phase transitions between trivially under- and over-constrained instances, and the actually hard instances reside in the parameter region of this very phase transition. For SAT, the level of under- or over-constrainedness is measured by the coefficient of variables to clauses $\alpha = |C|/|V|$, where

$|C|$ and $|V|$ denote the number of clauses and variables, respectively. The phase transition of SAT happens at $\alpha \approx 4$. For our comparison, we generated 50 instances for each value of $\alpha = 2$, $\alpha = 4$ and $\alpha = 6$. As before, the non-iterative QAOA approach is on par with standard QAOA for under- ($\alpha = 2$) and over-constrained ($\alpha = 6$) SAT instances, as well as for hard SAT instances right in the phase transition at $\alpha = 4$. This underlines the utility of our approach not only for "average", but also hard instances. Fig. 12 visualises the outcome distribution in detail.

Judging from the above evaluation, our non-iterative QAOA algorithm at least matches standard QAOA success probabilities, and occasionally shows slight advantage. However, QAOA involves classical parameter optimisation for every instance, while our approach is more resource efficient: Only a constant number of quantum circuit evaluations instead of sampling the circuit in every iteration of the classical optimisation loop is required, and we avoid instance-specific classical optimisation altogether: One single up-front classical optimisation process is required to infer optimal parameters from the approximated landscape per problem. Costly circuit evaluations in the quantum-classical iteration of QAOA are replaced by classical instance sampling in non-iterative QAOA: The quantum resource is only used for instance optimisation, not for preparatory work.

VII. DISCUSSION

We have considered QAOA from a mixed perspective comprising computer science and physics, and could not only establish a new approximation theorem for the optimisation landscape, but have also suggested an algorithm that translates these insights directly into useful advantages. The evaluation of both aspects in the preceding sections has shown that either improves upon the state of the art in various ways.

However, our results also raise new questions. Obvious issues include how to extend the approach to QAOA depths larger than one; possible gains in post-NISQ systems; a more comprehensive evaluation on specific industrial problems and larger instance sizes; and the robustness to noise. Likewise, the question of how computational power that eventually arises from our algorithm is distributed between classical and quantum resources, and what impact this distribution has on the properties like runtime or solution quality, remains open. We need to leave addressing these questions to future research.

For all subject problems analysed above in Section V our approximation matches the expected landscape extremely well, with negligibly small errors. Given that we could show the absolute approximation error to be bounded by $\sqrt{\mathrm{Var}(|T|)\,\mathrm{Var}\left(\overline{|c_k|^2}\right)}$, it follows that for all problems with constant solution space sizes (*i.e.*, $\mathrm{Var}(|T|) = 0$), our approximation actually delivers exact results. It should
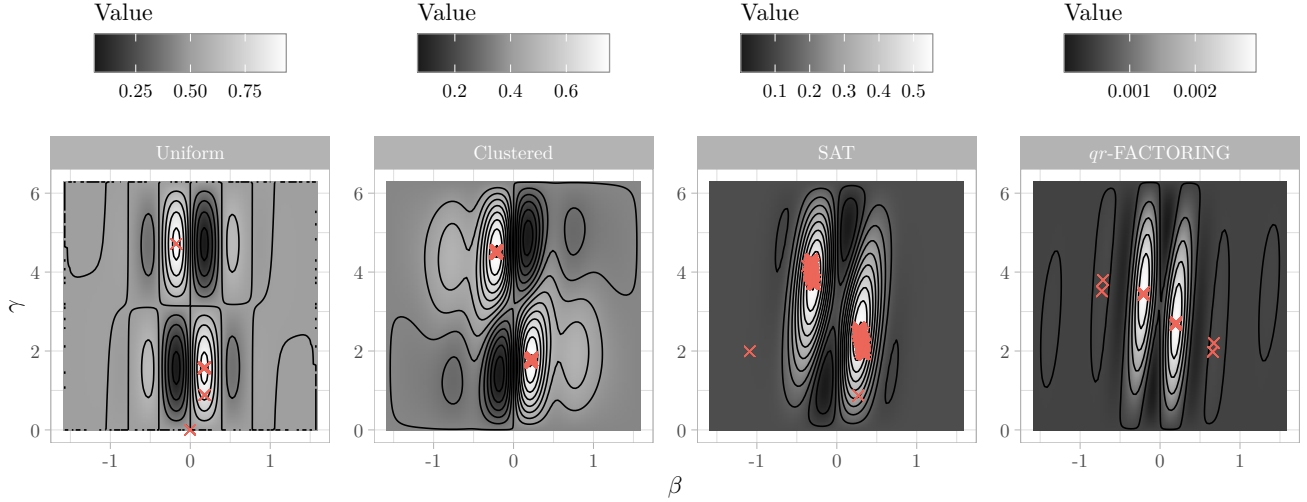
FIG. 10. Each red cross marks a combination (β, γ) obtained from the standard QAOA algorithm for one single instance of the subject problem. The parameter pairs are plotted on top the expected optimisation landscapes. As is visually apparent, optimal parameters for different instances follow the *problem-specific*, but *instance-independent* patterns expected from our considerations. Solutions for local optima of the instance-averaged landscape, as they arise for some QAOA executions, are typical artefacts expected from numerical optimisation.
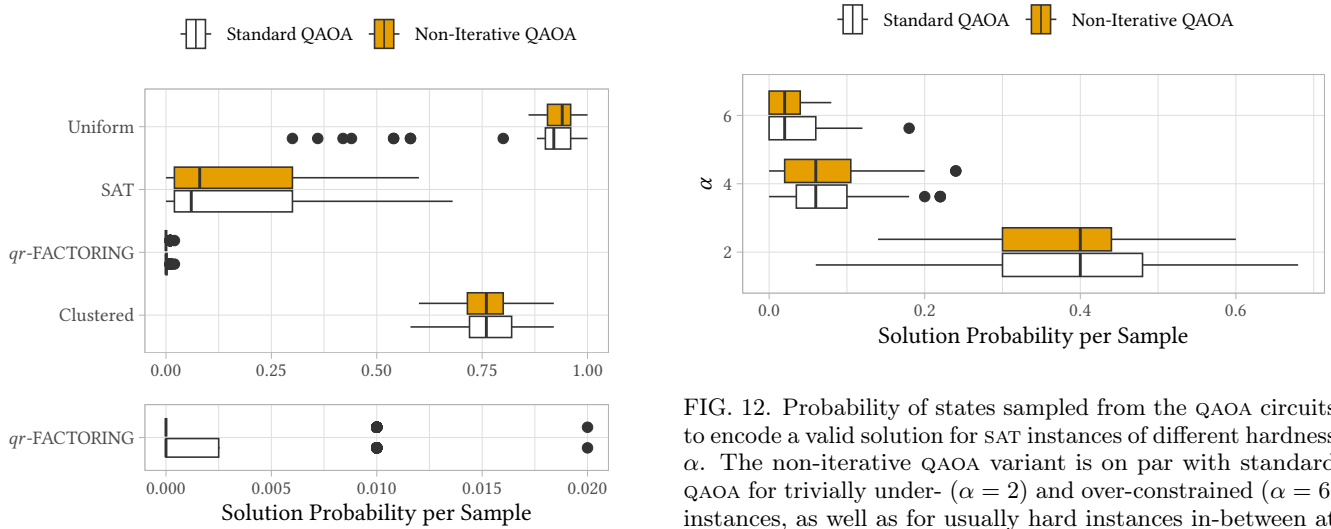
FIG. 11. Probability of states sampled from the QAOA circuits to encode valid solution for the various subject problems considered in this paper with a standard QAOA approach (individual parameter optimisation for each instance), and with our non-iterative schema using a-priori parameters obtained from the approximated landscape for all instances.

FIG. 12. Probability of states sampled from the QAOA circuits to encode a valid solution for SAT instances of different hardness α. The non-iterative QAOA variant is on par with standard QAOA for trivially under- ($\alpha = 2$) and over-constrained ($\alpha = 6$) instances, as well as for usually hard instances in-between at $\alpha = 4$ (a comparison between standard and non-iterative QAOA over all SAT instances is included in Fig. 11).

be noted that this requires knowledge of exact values for $\left\{ E\left(\overline{\#_d(k)}\right) \right\}_{d=0}^n$ and $\left\{ E\left(\overline{\#_{d_1}(k)\#_{d_2}(k)}\right) \right\}_{d_1, d_2 = 0}^n$, which is usually not practically achievable, especially for empirical sampling. This explains the approximation errors observed for all subject problems with fixed target sizes above—except for SAT. This observation naturally

raises questions about the relationship between the estimation accuracy of the target space structures and the resulting approximation error. Here, a trade-off between sample size and approximation quality is to be expected and should be analysed in future work. An answer to this question also could give interesting insight into the variance of optimisation landscapes between individual problem instances.

Our optimisation landscape approximation is solely based on structural information about a problem. This allowed us to prove the existence of conjectured instance
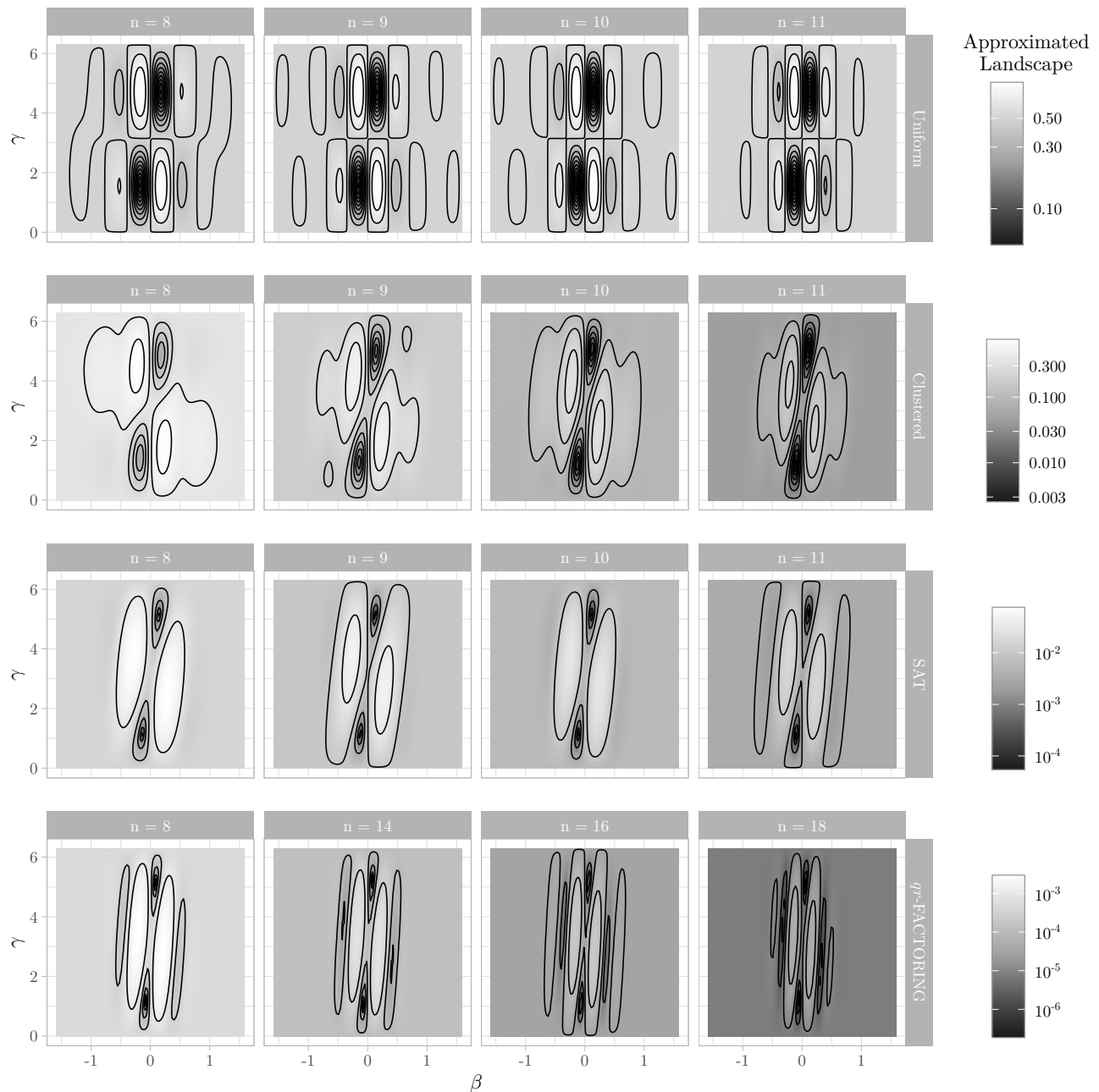
FIG. 13. Overview of all approximated landscapes for the considered subject problems (100 random instances per problem and dimension). Each panel shows the approximated optimisation landscape $\tilde{E}(F_1(\beta, \gamma))$ obtained by Theorem 1 as surrogate for the expected value of $F_1(\beta, \gamma)$. Macroscopic similarities do not only arise between instances of different dimensions (left to right), but also to a certain extent across problems (top to bottom).

invariants that stem from observed effects like parameter clustering [27, 28]. The approximation allows us to obtain a smooth representation of the complete optimisation landscape. In Fig. 13, we collect such landscapes for the subject problems discussed in Section V for different dimensions of the target space T. This provides us with some interesting observations: On the one hand, it

can be seen that macroscopic similarities exist between all problems. They share very similar high level features (with some problem specific differentiation). These macroscopic features are also present for structure-less target sets like in the case of the uniform sampling example. We thus conjecture that these features are truly problem independent and could be explained by either just

the approximation theorem presented in this work, or in combination with straight forward and minimalistic examples like uniform sampled target sets. Additional structure causes some displacement and skewing. This structure can be plausibly explained by local effects like membership of a state to the target set depending on the membership of other neighbouring states. As we can additionally observe from Fig. 13, simply increasing the state space dimension (*i.e.*, going from left to right on the panels for a specific problem) seems to compress the landscape along $\beta = 0$ along the β-axis. This effect is independent of the subject problem (*i.e.*, does not change when traversing the panels from top to bottom in the plot). As it can also be observed in the uniform sampling example, it must be independent of structural properties. While further examining these effects could provide valuable insight into the behaviour of QAOA optimisation landscapes, we need to leave the efforts required for such an investigation to future work, as they go beyond the scope of this paper. The approximation theorem devised in this paper will likely serve as an analytical basis for such considerations.

Focusing on two-level Hamiltonians and QAOA circuits keeps the approximation framework simple and straight-forward to use. We thus provide a well defined base framework to analyse the essential connection of problem structure and QAOA landscapes. At the same time, it is extendable to accommodate, for instance, Hamiltonians with more complex eigenspectra or QAOA circuits with more layers, albeit we also need to leave such efforts to future work. However, it seems pertinent to note that the landscapes for the subject problems considered in this work match empirical results with higher level eigenspectra and deeper QAOA circuits in their macroscopic structure [19, 28]. The same holds true for Ising model QAOA landscapes based on an instance-specific analytical derivation [67].

We believe our work lays a foundation to map significant insights from classical theoretical computer science to quantum approaches along the lines of QAOA. In the classical case, the analysis of solution space structures is well established [73–76]. With the approximation theorem (and the techniques introduced to derive it), we present a handle to apply this knowledge to quantum optimisation. This provides an opportunity for future progress in understanding and utilising the class of algorithms initiated by QAOA.

VIII. CONCLUSION

Our new perspective on QAOA from a mixed physics and theoretical computer science point of view allowed us to accurately approximate the QAOA optimisation landscape based on the inherent structural properties of a problem, and prove a long-standing set of hypotheses about relationships between optimal QAOA parameters for different instances of a problem. By directly linking the solution

space structure of problems to their QAOA optimisation landscapes, we could devise an approximation theorem that does not only provide structural insights, but also impacts the practical use of QAOA.

Based on our results, we have constructed a non-iterative QAOA algorithm that is more resource-efficient than standard QAOA for various measures. Our evaluation on five different scenarios and subject problems has shown excellent agreement from the theoretical and practical point of view, and provides concrete advantages over standard QAOA. Our new perspective on QAOA opens various possibilities for future research to understand unsolved issues about quantum optimisation.

Appendix A: Reproduction Package

To make our experiments reproducible by other researchers [91], we provide the complete source code for the calculations performed in the paper, in form of a long-term stable [92] reproduction package (link in PDF; a DOI-safe version[2] is also available). We ascertain that the package is fully self-contained, and does not rely on resources that may eventually vanish from public access. In particular, we provide code that is easily extensible to more subject problems than considered in the paper, and include routines to (a) perform target space sampling; (b) implement non-iterative QAOA; (c) perform numerical simulations to explore performance and compare with standard QAOA. Any raw data obtained from our simulations, together with a complete pipeline to perform the evaluation and visualisation presented in the paper, are included.

---

[2] Will be made available for the camera-ready version.

[1] N. Pirnay, V. Ulitzsch, F. Wilde, J. Eisert, and J.-P. Seifert, An in-principle super-polynomial quantum advantage for approximating combinatorial optimization problems via computational learning theory, Science Advances **10**, 10.1126/sciadv.adj5170 (2024).

[2] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. S. L. Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi, B. Foxen, A. Fowler, C. Gidney, M. Giustina, R. Graff, K. Guerin, S. Habegger, M. P. Harrigan, M. J. Hartmann, A. Ho, M. Hoffmann, T. Huang, T. S. Humble, S. V. Isakov, E. Jeffrey, Z. Jiang, D. Kafri, K. Kechedzhi, J. Kelly, P. V. Klimov, S. Knysh, A. Korotkov, F. Kostritsa, D. Landhuis, M. Lindmark, E. Lucero, D. Lyakh, S. Mandrà, J. R. McClean, M. McEwen, A. Megrant, X. Mi, K. Michielsen, M. Mohseni, J. Mutus, O. Naaman, M. Neeley, C. Neill, M. Y. Niu, E. Ostby, A. Petukhov, J. C. Platt, C. Quintana, E. G. Rieffel, P. Roushan, N. C. Rubin, D. Sank, K. J. Satzinger, V. Smelyanskiy, K. J. Sung, M. D. Trevithick, A. Vainsencher, B. Villalonga, T. White, Z. J. Yao, P. Yeh, A. Zalcman, H. Neven, and J. M. Martinis, Quantum supremacy using a programmable superconducting processor, Nature **574**, 505–510 (2019).

[3] J. Montanez-Barrera and K. Michielsen, Towards a universal qaoa protocol: Evidence of quantum advantage in solving combinatorial optimization problems, arXiv preprint arXiv:2405.09169 (2024).

[4] 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, Noisy intermediate-scale quantum algorithms, Rev. Mod. Phys. **94**, 015004 (2022).

[5] F. Greiwe, T. Krüger, and W. Mauerer, Effects of imperfections on quantum algorithms: A software engineering perspective, in *IEEE International Conference on Quantum Software* (2023) pp. 31–42.

[6] S. Thelen, H. Safi, and W. Mauerer, Approximating under the influence of quantum noise and compute power, in *Proceedings of the IEEE Conference on Quantum Computing and Engineering*, HPCQC@QCE'24 (2024).

[7] K. Blekos, D. Brand, A. Ceschini, C.-H. Chou, R.-H. Li, K. Pandya, and A. Summer, A review on Quantum Approximate Optimization Algorithm and its variants, Physics Reports **1068**, 1 (2024).

[8] L. M. Adleman, J. DeMarrais, and M.-D. A. Huang, Quantum computability, SIAM Journal on Computing **26**, 1524–1540 (1997).

[9] E. Bernstein and U. Vazirani, Quantum complexity theory, SIAM Journal on Computing **26**, 1411–1473 (1997).

[10] L. Fortnow and J. Rogers, Complexity limitations on quantum computation, Journal of Computer and System Sciences **59**, 240–252 (1999).

[11] S. Aaronson, Read the fine print, Nature Physics **11**, 291–293 (2015).

[12] A. Montanaro, Quantum algorithms: an overview, npj Quantum Information **2**, 10.1038/npjqi.2015.23 (2016).

[13] C. Shao, Y. Li, and H. Li, Quantum algorithm design: Techniques and applications, Journal of Systems Science and Complexity **32**, 375–452 (2019).

[14] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, Variational quantum algorithms, Nature Reviews Physics **3**, 625–644 (2021).

[15] A. Pellow-Jarman, S. McFarthing, I. Sinayskiy, D. K. Park, A. Pillay, and F. Petruccione, The effect of classical optimizers and ansatz depth on qaoa performance in noisy devices, Scientific Reports **14**, 10.1038/s41598-024-66625-6 (2024).

[16] M. Fernández-Pendás, E. F. Combarro, S. Vallecorsa, J. Ranilla, and I. F. Rúa, A study of the performance of classical minimizers in the quantum approximate optimization algorithm, Journal of Computational and Applied Mathematics **404**, 113388 (2022).

[17] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices, Physical Review X **10**, 021067 (2020).

[18] A. Bayerstadler, G. Becquin, J. Binder, T. Botter, H. Ehm, T. Ehmer, M. Erdmann, N. Gaus, P. Harbach, M. Hess, J. Klepsch, M. Leib, S. Luber, A. Luckow, M. Mansky, W. Mauerer, F. Neukart, C. Niedermeier, L. Palackal, R. Pfeiffer, C. Polenz, J. Sepulveda, T. Sievers, B. Standen, M. Streif, T. Strohm, C. Utschig-Utschig, D. Volz, H. Weiss, and F. Winter, Industry quantum computing applications, EPJ Quantum Technology **8**, 10.1140/epjqt/s40507-021-00114-x (2021).

[19] E. Pelofske, A. Bärtschi, and S. Eidenbenz, Short-depth qaoa circuits and quantum annealing on higher-order ising models, npj Quantum Information **10**, 30 (2024).

[20] L. Schmidbauer, K. Wintersperger, E. Lobe, and W. Mauerer, Polynomial reduction methods and their impact on qaoa circuits, in *IEEE International Conference on Quantum Software (QSW)* (2024).

[21] B. A. Cipra, The ising model is np-complete, SIAM News **33**, 1 (2000).

[22] A. Lucas, Ising formulations of many NP problems, Frontiers in physics **2**, 5 (2014).

[23] S. Arora and B. Barak, *Computational Complexity: A Modern Approach* (Cambridge University Press, 2009).

[24] P. Díez-Valle, D. Porras, and J. J. García-Ripoll, Connection between single-layer quantum approximate optimization algorithm interferometry and thermal distribution sampling, Frontiers in Quantum Science and Technology **3**, 1321264 (2024).

[25] M. Streif and M. Leib, Comparison of qaoa with quantum and simulated annealing, arXiv preprint arXiv:1901.01903 (2019).

[26] S. Bravyi, A. Kliesch, R. Koenig, and E. Tang, Obstacles to variational quantum optimization from symmetry protection, Physical review letters **125**, 260505 (2020).

[27] F. G. Brandao, M. Broughton, E. Farhi, S. Gutmann, and H. Neven, For fixed control parameters the quantum approximate optimization algorithm's objective function value concentrates for typical instances, arXiv preprint arXiv:1812.04170 (2018).

[28] M. Streif and M. Leib, Training the quantum approximate optimization algorithm without access to a quantum processing unit, Quantum Science and Technology **5**, 034008 (2020).

[29] S. H. Sack and M. Serbyn, Quantum annealing initializa-

tion of the quantum approximate optimization algorithm, quantum **5**, 491 (2021).

[30] A. Galda, E. Gupta, J. Falla, X. Liu, D. Lykov, Y. Alexeev, and I. Safro, Similarity-based parameter transferability in the quantum approximate optimization algorithm, Frontiers in Quantum Science and Technology **2**, 1200975 (2023).

[31] J. Sud, S. Hadfield, E. Rieffel, N. Tubman, and T. Hogg, Parameter-setting heuristic for the quantum alternating operator ansatz, Phys. Rev. Res. **6**, 10.1103/PhysRevResearch.6.023171 (2024).

[32] L. Bittel and M. Kliesch, Training Variational Quantum Algorithms Is NP-Hard, Physical Review Letters **127**, 120502 (2021), publisher: American Physical Society.

[33] M. Fernández-Pendás, E. F. Combarro, S. Vallecorsa, J. Ranilla, and I. F. Rúa, A study of the performance of classical minimizers in the Quantum Approximate Optimization Algorithm, Journal of Computational and Applied Mathematics **404**, 113388 (2022).

[34] S. Hadfield, Z. Wang, B. O'Gorman, E. G. Rieffel, D. Venturelli, and R. Biswas, From the quantum approximate optimization algorithm to a quantum alternating operator ansatz, Algorithms **12**, 10.3390/a12020034 (2019).

[35] V. Akshay, D. Rabinovich, E. Campos, and J. Biamonte, Parameter concentrations in quantum approximate optimization, Physical Review A **104**, L010401 (2021).

[36] P. C. Lotshaw, T. S. Humble, R. Herrman, J. Ostrowski, and G. Siopsis, Empirical performance bounds for quantum approximate optimization, Quantum Information Processing **20**, 403 (2021), arXiv:2102.06813 [physics, physics:quant-ph].

[37] J. Weidenfeller, L. C. Valor, J. Gacon, C. Tornow, L. Bello, S. Woerner, and D. J. Egger, Scaling of the quantum approximate optimization algorithm on superconducting qubit based hardware, Quantum **6**, 870 (2022).

[38] M. S. Alam, F. A. Wudarski, M. J. Reagor, J. Sud, S. Grabbe, Z. Wang, M. Hodson, P. A. Lott, E. G. Rieffel, and D. Venturelli, Practical verification of quantum properties in quantum-approximate-optimization runs, Physical Review Applied **17**, 10.1103/physrevapplied.17.024026 (2022).

[39] M. P. Harrigan, K. J. Sung, M. Neeley, K. J. Satzinger, F. Arute, K. Arya, J. Atalaya, J. C. Bardin, R. Barends, S. Boixo, M. Broughton, B. B. Buckley, D. A. Buell, B. Burkett, N. Bushnell, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, S. Demura, A. Dunsworth, D. Eppens, A. Fowler, B. Foxen, C. Gidney, M. Giustina, R. Graff, S. Habegger, A. Ho, S. Hong, T. Huang, L. B. Ioffe, S. V. Isakov, E. Jeffrey, Z. Jiang, C. Jones, D. Kafri, K. Kechedzhi, J. Kelly, S. Kim, P. V. Klimov, A. N. Korotkov, F. Kostritsa, D. Landhuis, P. Laptev, M. Lindmark, M. Leib, O. Martin, J. M. Martinis, J. R. McClean, M. McEwen, A. Megrant, X. Mi, M. Mohseni, W. Mruczkiewicz, J. Mutus, O. Naaman, C. Neill, F. Neukart, M. Y. Niu, T. E. O'Brien, B. O'Gorman, E. Ostby, A. Petukhov, H. Putterman, C. Quintana, P. Roushan, N. C. Rubin, D. Sank, A. Skolik, V. Smelyanskiy, D. Strain, M. Streif, M. Szalay, A. Vainsencher, T. White, Z. J. Yao, P. Yeh, A. Zalcman, L. Zhou, H. Neven, D. Bacon, E. Lucero, E. Farhi, and R. Babbush, Quantum approximate optimization of non-planar graph problems on a planar superconducting processor, Nature Physics **17**, 332–336 (2021).

[40] R. Herrman, L. Treffert, J. Ostrowski, P. C. Lotshaw, T. S.

Humble, and G. Siopsis, Globally optimizing qaoa circuit depth for constrained optimization problems, Algorithms **14**, 294 (2021).

[41] J. Lee, A. B. Magann, H. A. Rabitz, and C. Arenz, Progress toward favorable landscapes in quantum combinatorial optimization, Physical Review A **104**, 10.1103/physreva.104.032401 (2021).

[42] M. Medvidović and G. Carleo, Classical variational simulation of the quantum approximate optimization algorithm, npj Quantum Information **7**, 10.1038/s41534-021-00440-z (2021).

[43] M. Willsch, D. Willsch, F. Jin, H. De Raedt, and K. Michielsen, Benchmarking the quantum approximate optimization algorithm, Quantum Information Processing **19**, 10.1007/s11128-020-02692-8 (2020).

[44] Z. Wang, N. C. Rubin, J. M. Dominy, and E. G. Rieffel, Xy mixers: Analytical and numerical results for the quantum alternating operator ansatz, Physical Review A **101**, 10.1103/physreva.101.012320 (2020).

[45] A. Bengtsson, P. Vikstål, C. Warren, M. Svensson, X. Gu, A. F. Kockum, P. Krantz, C. Križan, D. Shiri, I.-M. Svensson, G. Tancredi, G. Johansson, P. Delsing, G. Ferrini, and J. Bylander, Improved success probability with greater circuit depth for the quantum approximate optimization algorithm, Physical Review Applied **14**, 10.1103/physrevapplied.14.034010 (2020).

[46] D. Wecker, M. B. Hastings, and M. Troyer, Training a quantum optimizer, Physical Review A **94**, 10.1103/physreva.94.022309 (2016).

[47] Z. Jiang, E. G. Rieffel, and Z. Wang, Near-optimal quantum circuit for grover's unstructured search using a transverse field, Physical Review A **95**, 10.1103/physreva.95.062317 (2017).

[48] M. E. S. Morales, J. D. Biamonte, and Z. Zimborás, On the universality of the quantum approximate optimization algorithm, Quantum Information Processing **19**, 10.1007/s11128-020-02748-9 (2020).

[49] W. Lechner, Quantum approximate optimization with parallelizable gates, IEEE Transactions on Quantum Engineering **1**, 1–6 (2020).

[50] Y. Pan, Y. Tong, S. Xue, and G. Zhang, Efficient depth selection for the implementation of noisy quantum approximate optimization algorithm, Journal of the Franklin Institute **359**, 11273–11287 (2022).

[51] G. G. Guerreschi and A. Y. Matsuura, Qaoa for maxcut requires hundreds of qubits for quantum speed-up, Scientific Reports **9**, 10.1038/s41598-019-43176-9 (2019).

[52] M. Gogeißl, H. Safi, and W. Mauerer, Quantum data encoding patterns and their consequences, in Proceedings of the Workshop on Quantum Computing and Quantum-Inspired Technology for Data-Intensive Systems and Applications, QDSM '24 (2024).

[53] G. Pagano, A. Bapat, P. Becker, K. S. Collins, A. De, P. W. Hess, H. B. Kaplan, A. Kyprianidis, W. L. Tan, C. Baldwin, L. T. Brady, A. Deshpande, F. Liu, S. Jordan, A. V. Gorshkov, and C. Monroe, Quantum approximate optimization of the long-range ising model with a trapped-ion quantum simulator, Proceedings of the National Academy of Sciences **117**, 25396–25401 (2020).

[54] D. J. Egger, J. Mareček, and S. Woerner, Warm-starting quantum optimization, Quantum **5**, 479 (2021).

[55] V. Vijendran, A. Das, D. E. Koh, S. M. Assad, and P. K. Lam, An expressive ansatz for low-depth quantum approximate optimisation, Quantum Science and Technology **9**,

[56] H. Safi, K. Wintersperger, and W. Mauerer, Influence of hw-sw-co-design on quantum computing scalability, in *IEEE International Conference on Quantum Software (QSW)* (2023) pp. 104–115.

[57] K. Wintersperger, H. Safi, and W. Mauerer, Qpu-system co-design for quantum hpc accelerators, in *Lecture Notes in Computer Science* (Springer International Publishing, 2022) p. 100–114.

[58] M. Dupont, N. Didier, M. J. Hodson, J. E. Moore, and M. J. Reagor, Entanglement perspective on the quantum approximate optimization algorithm, Physical Review A **106**, 10.1103/physreva.106.022423 (2022).

[59] C. C. McGeoch and P. Farré, Milestones on the quantum utility highway: Quantum annealing case study, ACM Transactions on Quantum Computing **5**, 10.1145/3625307 (2023).

[60] A. Awasthi, F. Bär, J. Doetsch, H. Ehm, M. Erdmann, M. Hess, J. Klepsch, P. A. Limacher, A. Luckow, C. Niedermeier, L. Palackal, R. Pfeiffer, P. Ross, H. Safi, J. Schönmeier-Kromer, O. von Sicard, Y. Wenger, K. Wintersperger, and S. Yarkoni, Quantum computing techniques for multi-knapsack problems, in *Intelligent Computing* (Springer Nature, 2023).

[61] A. Bärtschi and S. Eidenbenz, Grover Mixers for QAOA: Shifting complexity from mixer design to state preparation, in *Proc. IEEE QCE* (2020) pp. 72–82.

[62] R. Tate, J. Moondra, B. Gard, G. Mohler, and S. Gupta, Warm-Started QAOA with Custom Mixers Provably Converges and Computationally Beats Goemans-Williamson's Max-Cut at Low Circuit Depths, Quantum **7**, 1121 (2023).

[63] J. R. Finžgar, A. Kerschbaumer, M. J. Schuetz, C. B. Mendl, and H. G. Katzgraber, Quantum-informed recursive optimization algorithms, PRX Quantum **5**, 10.1103/prxquantum.5.020327 (2024).

[64] S. Singhal, V. Srivastava, P. Rohith, P. Jain, and D. Bhowmik, Performance comparison for quantum approximate optimization algorithm (qaoa) across noiseless simulation, experimentally benchmarked noisy simulation, and experimental hardware platforms, in *2024 8th IEEE Electron Devices Technology & Manufacturing Conference (EDTM)* (2024) pp. 1–3.

[65] D. Stilck França and R. García-Patrón, Limitations of optimization algorithms on noisy quantum devices, Nature Physics **17**, 1221–1227 (2021).

[66] R. Shaydulin, C. Li, S. Chakrabarti, M. DeCross, D. Herman, N. Kumar, J. Larson, D. Lykov, P. Minssen, Y. Sun, Y. Alexeev, J. M. Dreiling, J. P. Gaebler, T. M. Gatterman, J. A. Gerber, K. Gilmore, D. Gresh, N. Hewitt, C. V. Horst, S. Hu, J. Johansen, M. Matheny, T. Mengle, M. Mills, S. A. Moses, B. Neyenhuis, P. Siegfried, R. Yalovetzky, and M. Pistoia, Evidence of scaling advantage for the quantum approximate optimization algorithm on a classically intractable problem, Science Advances **10**, 10.1126/sciadv.adm6761 (2024).

[67] A. Ozaeta, W. van Dam, and P. L. McMahon, Expectation values from the single-layer quantum approximate optimization algorithm on ising problems, Quantum Science and Technology **7**, 045036 (2022).

[68] Z. Jiang, E. G. Rieffel, and Z. Wang, Near-optimal quantum circuit for grover's unstructured search using a transverse field, Phys. Rev. A **95**, 10.1103/PhysRevA.95.062317 (2017).

[69] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, arXiv preprint arXiv:1411.4028 (2014).

[70] N. Jain, B. Coyle, E. Kashefi, and N. Kumar, Graph neural network initialisation of quantum approximate optimisation, Quantum **6**, 861 (2022).

[71] E. Wybo and M. Leib, Missing puzzle pieces in the performance landscape of the quantum approximate optimization algorithm, arXiv preprint arXiv:2406.14618 (2024).

[72] R. Monasson, R. Zecchina, S. Kirkpatrick, B. Selman, and L. Troyansky, Determining computational complexity from characteristic 'phase transitions', Nature **400**, 133 (1999).

[73] T. Hogg, Refining the phase transition in combinatorial search, Artificial Intelligence **81**, 127 (1996).

[74] P. R. Pari, J. Lin, L. Yuan, and G. Qu, Generating 'random' 3-sat instances with specific solution space structure, in *Proceedings of the Nineteenth National Conference on Artificial Intelligence, Sixteenth Conference on Innovative Applications of Artificial Intelligence, July 25-29, 2004, San Jose, California, USA*, edited by D. L. McGuinness and G. Ferguson (AAAI Press / The MIT Press, 2004) pp. 960–961.

[75] D. Achlioptas, A. Coja-Oghlan, and F. Ricci-Tersenghi, On the solution-space geometry of random constraint satisfaction problems, Random Structures & Algorithms **38**, 251 (2011).

[76] C. Ansótegui, J. Giráldez-Cru, and J. Levy, The community structure of SAT formulas, in *Theory and Applications of Satisfiability Testing - SAT 2012 - 15th International Conference, Trento, Italy, June 17-20, 2012. Proceedings*, Lecture Notes in Computer Science, Vol. 7317, edited by A. Cimatti and R. Sebastiani (Springer, 2012) pp. 410–423.

[77] G. Kochenberger, J.-K. Hao, F. Glover, M. Lewis, Z. Lü, H. Wang, and Y. Wang, The unconstrained binary quadratic programming problem: a survey, Journal of Combinatorial Optimization **28**, 58–81 (2014).

[78] M. Aramon, G. Rosenberg, E. Valiante, T. Miyazawa, H. Tamura, and H. G. Katzgraber, Physics-inspired optimization for quadratic unconstrained problems using a digital annealer, Frontiers in Physics **7**, 10.3389/fphy.2019.00048 (2019).

[79] K. Henke, E. Pelofske, G. Hahn, and G. Kenyon, Sampling binary sparse coding qubo models using a spiking neuromorphic processor, in *Proceedings of the 2023 International Conference on Neuromorphic Systems*, ICONS '23 (Association for Computing Machinery, New York, NY, USA, 2023).

[80] M. Schönberger, I. Trummer, and W. Mauerer, Quantum-inspired digital annealing for join ordering, Proc. VLDB Endow. **17**, 511–524 (2023).

[81] M. Z. Alom, B. Van Essen, A. T. Moody, D. P. Widemann, and T. M. Taha, Quadratic unconstrained binary optimization (qubo) on neuromorphic computing system, in *2017 International Joint Conference on Neural Networks (IJCNN)* (2017) pp. 3922–3929.

[82] O. Şeker, N. Tanoumand, and M. Bodur, Digital annealer for quadratic unconstrained binary optimization: A comparative performance analysis, Applied Soft Computing **127**, 109367 (2022).

[83] V. Ganesh and M. Y. Vardi, On the unreasonable effectiveness of sat solvers, in *Beyond the Worst-Case Analysis of Algorithms* (Cambridge University Press, 2020) p. 547–566.

[84] T. Gabor, S. Zielinski, S. Feld, C. Roch, C. Seidel, F. Neukart, I. Galter, W. Mauerer, and C. Linnhoff-Popien, Assessing solution quality of 3sat on a quantum annealing platform, Proc. Int. Workshop on Quantum Technology and Optimization Problems (QTOP) (2019).

[85] I. Sax, S. Feld, S. Zielinski, T. Gabor, C. Linnhoff-Popien, and W. Mauerer, Approximate approximation on a quantum annealer, in *Proceedings of the 17th ACM International Conference on Computing Frontiers* (Association for Computing Machinery, New York, NY, USA, 2020) p. 108–117.

[86] T. Krüger and W. Mauerer, Quantum annealing-based software components: An experimental case study with SAT solving, in *Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops*, ICSEW'20 (Association for Computing Machinery, New York, NY, USA, 2020) p. 445–450.

[87] D. Willsch, M. Willsch, F. Jin, K. Michielsen, and H. De Raedt, Gpu-accelerated simulations of quantum annealing and the quantum approximate optimization algorithm, Computer Physics Communications **278**, 108411 (2022).

[88] S. Zielinski, J. Nüßlein, J. Stein, T. Gabor, C. Linnhoff-Popien, and S. Feld, Pattern qubos: Algorithmic construction of 3sat-to-qubo transformations, Electronics **12**, 3492 (2023).

[89] G. Audemard and L. Simon, Predicting learnt clauses quality in modern sat solvers (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2009) p. 399–404.

[90] M. Periyasamy, A. Plinge, C. Mutschler, D. D. Scherer, and W. Mauerer, Guided-spsa: Simultaneous perturbation stochastic approximation assisted by the parameter shift rule (2024), arXiv:2404.15751 [quant-ph].

[91] *Reproducibility and Replicability in Science* (National Academies Press, 2019).

[92] W. Mauerer and S. Scherzinger, 1-2-3 reproducibility for quantum software experiments, in *IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)* (2022) pp. 1247–1248.