Solving Boolean Satisfiability Problems With The Quantum Approximate Optimization Algorithm

Sami Boulebnane[*,†] and Ashley Montanaro[‡]
Phasecraft Ltd., Bristol, United Kingdom

(Received 19 September 2023; revised 28 July 2024; accepted 5 August 2024; published 10 September 2024)

One of the most prominent application areas for quantum computers is solving hard constraint satisfaction and optimization problems. However, detailed analyses of the complexity of standard quantum algorithms have suggested that outperforming classical methods for these problems would require extremely large and powerful quantum computers. The quantum approximate optimization algorithm (QAOA) is designed for near-term quantum computers, yet previous work has shown strong limitations on the ability of QAOA to outperform classical algorithms for optimization problems. Here we instead apply QAOA to hard constraint satisfaction problems, where both classical and quantum algorithms are expected to require exponential time. We analytically characterize the average success probability of QAOA on a constraint satisfaction problem commonly studied using statistical physics methods: random k-SAT at the threshold for satisfiability, as the number of variables n goes to infinity. We complement these theoretical results with numerical experiments on the performance of QAOA for small n, which match the limiting theoretical bounds closely. We then compare QAOA with leading classical solvers. For random 8-SAT, we find that for more than 14 quantum circuit layers, QAOA achieves more efficient scaling than the highest-performance classical solver we tested, WalkSATlm. Our results suggest that near-term quantum algorithms for solving constraint satisfaction problems may outperform their classical counterparts.

DOI: 10.1103/PRXQuantum.5.030348

I. INTRODUCTION

Hard constraint satisfaction and optimization problems are ubiquitous in society, and are critically important in fields ranging from finance to logistics. Problems of this form require finding a joint assignment to a set of variables such that certain constraints on the variables are satisfied, or such that a cost function of the variables is minimized. Unfortunately, these problems are often extremely challenging to solve, with our best algorithms requiring exponential time in the worst case.

Quantum computers could solve problems like this more efficiently than classical computers can. Grover's quantum algorithm famously achieves a quadratic speedup over classical unstructured search [1], and can be applied to solve unstructured optimization problems [2]. This family of algorithms requires a fault-tolerant quantum computer, with very significant overheads for error correction [3,4], and can be outperformed by classical methods tailored to the problem being solved. In the setting of near-term quantum computing, the most well-studied approach to solving optimization problems is the quantum approximate optimization algorithm (QAOA) [5,6]. In a very influential pair of studies, Farhi, Goldstone, and Gutmann [6,7] found provable bounds on the performance of this algorithm for instances of the Max-Cut and Max-E3Lin2 optimization problems. In the latter case, for certain families of instances, QAOA outperformed the best classical algorithm known at the time. However, a classical algorithm was then found, which outperformed QAOA [8]. Although there have been many subsequent works on the theoretical and empirical performance of QAOA for optimization problems (see Ref. [9] for a review), none has yet shown an unambiguous advantage over the best classical algorithms.

Here, we study the performance of QAOA for hard constraint satisfaction problems. For problems of this form, we seek to find a solution that exactly satisfies all constraints, and expect the algorithm's running time to scale exponentially with the number of variables n. We focus on the fundamental Boolean satisfiability problem, in the form of random k-SAT, where one is given a randomly generated Boolean formula with k variables per clause, and aims to

[*]Contact author: sami.boulebnane.19@ucl.ac.uk
[†]Also at University College London.
[‡]Also at University of Bristol.

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

find an assignment to the variables that satisfies all clauses. We define this problem formally in Definition 1 below; an example of a k-SAT formula with $k = 2$ and $n = 4$ is

$$(x_0 \vee \overline{x_1}) \wedge (x_1 \vee x_2) \wedge (x_1 \vee \overline{x_3}), \tag{1}$$

where $x_0, \ldots, x_3$ are Boolean (0/1-valued) variables, the bar over a variable denotes negation and $\vee$ (respectively, $\wedge$) denotes disjunction (OR) [respectively, conjunction (AND)]. The formula above is, for instance, satisfied by the assignment $x_0 = x_1 = 1, x_2 = x_3 = 0$. k-SAT is NP-complete for $k \geq 3$, implying that it is expected that no efficient algorithm exists to solve arbitrary k-SAT instances.

QAOA was already applied to random k-SAT by Hogg in a pioneering work in 2000 [5]. (Hogg's "quantum heuristic" is essentially identical to QAOA, the chief difference being a prescribed set of choices for the parameters of the algorithm.) Hogg applied his algorithm to hard random 3-SAT instances, using analytic arguments relying on a mean-field approximation, which, while nonrigorous, seemed to be largely confirmed by small-scale numerical simulations. He did not find an improvement in running time compared with the best classical algorithms for 3-SAT.

In this work, we consider the performance of QAOA on random k-SAT at constant depth with fixed angles: that is, the parameters of the algorithm are not allowed to depend on the problem instance. This feature is desirable as it means the quantum algorithm need not be modified for each instance (requiring many runs on a quantum computer). Fixed angles have been shown to achieve a nontrivial approximation ratio for a typical instance of the Max-Cut problem on random graphs [10]. Here, we aim to maximize the probability p_{succ} that QAOA outputs a satisfying assignment. Repeatedly running QAOA immediately translates into an algorithm for determining satisfiability whose expected running time on that instance is $1/p_{\text{succ}}$.

In the constant-depth regime, by a simple light-cone argument [6], QAOA outputs any optimal solutions with probability decaying exponentially with the problem size. This limitation is a generic feature of constraint satisfaction problems with a constant number of variables per constraint and a constant clauses-to-variables ratio. Nevertheless, in the context of random k-SAT close to the satisfiability threshold, this result should not necessarily lead to pessimism regarding the performance of QAOA: the reason is that state-of-the-art classical algorithms also empirically require exponential running time to solve this constraint satisfaction problem [11]. The question then becomes which of QAOA or classical solvers has the smallest empirical or theoretical running time exponent.

In this work we provide a theoretical and empirical analysis of the performance of QAOA on random k-SAT. First, we propose an analytic method to estimate the mean QAOA success probability over instances in the infinite-size limit, together with a concrete algorithmic implementation. The correctness of the algorithm is rigorously proven for sufficiently small variational parameters. We underline that in this context, "small" allows for constant parameters as the problem size goes to infinity, a regime in which QAOA remains hard to classically simulate. The analytic algorithm can be used in practice for relatively large numbers of quantum circuit layers p (up to $p = 10$ shown in this work) to evaluate or even train QAOA on random k-SAT. However, we also empirically show that when it comes to finding near-optimal average-instance variational parameters, the analytic method essentially coincides with a much easier one, namely, estimating the expected success probability from an empirical average over a limited set of modest-size instances. In particular, full simulation and training of QAOA even for large p is very efficient on a classical computer at this size.

Encouraged by the agreement between analytic and numerical results, we then benchmark constant-depth QAOA, trained with the "easier" method just described, against many classical solvers for random k-SAT. We find that the WalkSATlm solver [12] is consistently the most efficient classical solver. We focus on relatively large k, as this is the regime in which we find that QAOA achieves the highest performance relative to classical algorithms. Based on both our analytic and numerical results, we estimate that for random $k = 8$ instances at the satisfiability threshold, QAOA with about 14 circuit layers would match the performance of WalkSATlm, with a running time of at most $2^{0.33n}$ to find a satisfying assignment. Notably, this is significantly faster than naïve use of Grover's algorithm, and with a far lower-depth quantum circuit. For larger numbers of layers, we predict that QAOA will start to outperform WalkSATlm. The extent of the advantage is unclear. For 60 layers, for example, numerical estimates of the median running time for small instances suggest an approximately equal to $2^{0.30n}$ scaling, whereas based on theoretical results on the average success probability, the scaling could be as low as $2^{0.19n}$; the relevant numerical data and fits are reported in Fig. 4, and see Sec. III C for a discussion. We also tested a combination of QAOA and the classical WalkSAT algorithm, but found the improvement in performance over standard QAOA to be modest. We remark that, given a fault-tolerant quantum computer, amplitude amplification [13] can be used to reduce all of these exponents by a factor of 2.

The main theoretical contribution of this work is a technique to estimate a certain family of "generalized multinomial sums," extending the standard binomial and multinomial theorems. A similar goal was very recently achieved [14], leading to an estimate of the performance of QAOA on spin-glass models. That analysis relied on a sophisticated combinatorial analysis of generalized multinomial sums combined with complex analysis

techniques, and involved the new and nontrivial concept of *well-played polynomial*. This work is similar in that it considers generalized multinomial sums involving a certain family of (exponentiated) polynomials; however, instead of the "well-played" property, it rather requires the polynomial to be expressible as a sum of perfect powers. Despite its simplicity, this assumption covers the case of QAOA applied to random k-SAT where k is a power of 2, which is the focus of this work.

More precisely, the success probability of this quantum algorithm can be expressed as a generalized multinomial sum satisfying all required properties. We then estimate generalized multinomial sums by recasting them as integrals. The asymptotic scaling of these integrals (in the limit where the problem size n goes to infinity) can in turn be rigorously estimated using the *saddle-point method*. Unfortunately, the method is only fully justified if certain parameters defining the multinomial sum are sufficiently small. In the context of QAOA, this requirement translates to sufficiently small variational angles; however, they may be held constant as $n \to \infty$, a parameter regime where classical simulation or even prediction of QAOA performance remains nontrivial in general. This additional requirement is a shortcoming compared to the method developed in Ref. [14], which remains operational unconditional on the magnitude of the QAOA angles. However, the two approaches rely on very different assumptions and presumably do not apply to the same problems.

This work is organized as follows. In Sec. II, we introduce the required elements of background on random k-SAT and QAOA for the statement of our results. Analytic and numerical results are then discussed in Sec. III, including the analytic method of evaluationg the expected success probability of random k-SAT QAOA, exposed in Proposition 1. We defer remaining technical content to appendices. Appendix A is dedicated to the proof of the last result. It starts Appendix A 1 by recasting the expected success probability of QAOA on random k-SAT as a *generalized multinomial sum* (Definition 3). The rest of the work is dedicated to analyzing generalized multinomial sums—its application is therefore not necessarily limited to random k-SAT QAOA. In Appendix A 2, which is self-contained, we introduce a trivial yet instructive toy-model example (optimizing the Hamming weight squared cost function with QAOA), which gives an accurate flavor of the general method. In fact, the analysis of random 2-SAT (among other examples) almost immediately follows from this example, as discussed in Appendix A 3 b. In the Supplemental Material [15], we outline the analysis of the general case, applying in particular to random k-SAT QAOA with k a power of 2, is given. The algorithmic implementation of our method for estimating multinomial sums, hence the success probability of random k-SAT QAOA, is made explicit in Appendix A 3 c.

A. Other background and related work

1. Random k-SAT

We refer to Ref. [16] for a thorough and accessible review of recent progress in the field, recalling only a few salient facts here. Instances of random k-SAT are generated from a random ensemble of constraint satisfaction problems, which is parametrized by a positive integer k; the precise description of this ensemble used in this work is given in Definition 1. An important fact is, the existence of solutions to random k-SAT and the complexity of algorithmically finding them are related to the ratio between the number of constraints m and the number of variables n. For each integer $k \geq 1$ and problem size n (satisfiability thresholds r are reported in Table I), there exists a threshold $r_k(n)$, known as *satisfiability ratio*, such that for all $\varepsilon > 0$, a randomly generated instance admits solutions with high probability if $(m/n) < r_k(n) - \varepsilon$, while it is almost surely unsatisfiable for $(m/n) > r_k(n)$. The thresholds $r_k(n)$ are believed to admit a limit r_k as $n \to \infty$ (k fixed). This quantity can be practically estimated to good precision, either through numerical simulations or nonrigorous analytic arguments from statistical physics [17]. It is rigorously known that for sufficiently large k, $r_k(n) \sim 2^k \log(2)$ to leading order in k [16]. Remarkably, exact algorithms or heuristics are at least empirically known to solve random k-SAT efficient for a ratio $(m/n) < 2^k((\log(k))/k)$, but not any further beyond this ratio. The last ratio is known as *algorithmic ratio* and is strictly smaller than the *satisfiability ratio*, with a discrepancy increasing with k. It is therefore an outstanding problem to understand the ultimate limitations of different types of algorithms in the region between the algorithmic and the satisfiability threshold. This possibly involves considering nonconventional computational paradigms such as quantum computing.

2. Related work on QAOA

Recently, a numerical study [18] considered the complexity of training QAOA for a parametrized constraint satisfaction problem: exact k-cover, which is distinct from random k-SAT but also admits a threshold. The authors showed that the difficulty of training the variational circuit with the goal of producing a solution with high probability dramatically increased as one approached the threshold. This led them to conjecture that QAOA, similar to classical algorithms, underwent a phase transition when approaching the satisfiability threshold.

A distinctive feature of our work compared with earlier theoretical work on Max-Cut-QAOA is that we optimize the expected success probability in the average-instance case rather than the instancewise or average-instance energy. A similar idea was explored, e.g., in Ref. [19], which proposed to optimize the Gibbs free energy instancewise, leading to an empirical improvement

on the probability of finding a high-quality solution; we recall that depending on the temperature parameter the Gibbs free energy interpolates continuously between the expected energy of a sampled solution and the probability of sampling an optimal solution.

II. DEFINITIONS AND PRELIMINARIES

A. Notation

Given an integer variable n, we shall denote by $\overset{\text{poly}(n)}{\approx}$ equality up to a factor that is at most polynomial in n. Intuitively, if one considers exponential scalings, a polynomial factor is irrelevant and this approximate equality therefore signifies the exponential scalings are the same.

Besides, for any integer r, we denote

$$[r] := \{0, 1, \ldots, r-2, r-1\} \tag{2}$$

for the set of the first r natural integers (note this differs from the one-based convention frequently used in other works).

Given an integer $n \geq 1$ and r integers $n_0, \ldots, n_{r-1}$ summing to n, we denote by

$$\binom{n}{n_0, \ldots, n_{r-1}} = \binom{n}{(n_j)_{j\in[r]}} := \frac{n!}{\prod_{j\in[r]} n_j!} \tag{3}$$

multinomial coefficients, generalizing binomial coefficients and obeying a generalization of the binomial theorem (known as *multinomial theorem*):

$$\sum_{\substack{n_0,\ldots,n_{r-1} \\ n_0+\ldots+n_{r-1}=n}} \binom{n}{n_0, \ldots, n_{r-1}} \prod_{j\in[r]} x_j^{n_j} = \left(\sum_{j\in[r]} x_j\right)^n . \tag{4}$$

B. The quantum approximate optimization algorithm

In this section, we recall the principle of the quantum approximate optimization algorithm as described by Farhi *et al*. in Ref. [6] (see also Hogg's prior work [5]). QAOA is a quantum algorithm designed to find approximate solutions to combinatorial optimization problems; for the purpose of this work, it is sufficient to think of such a problem as the task of minimizing a cost function of n bits $H(x)$ $(x \in \{0,1\}^n)$. Finding approximate minimizers of this cost function can be rephrased as finding low-energy eigenstates of the corresponding n-qubit classical Hamiltonian:

$$H_C := \sum_{x\in\{0,1\}^n} H(x) \, |x\rangle \langle x| . \tag{5}$$

QAOA attempts to achieve this task by starting with a product state corresponding to a uniform superposition of bitstrings:

$$|+\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle , \tag{6}$$

and alternating Hamiltonian evolution under H_C and the transverse-field Hamiltonian

$$H_B := \sum_{j\in[n]} X_j . \tag{7}$$

The evolution times under H_B and H_C are hyperparameters of the algorithm to be optimized. Explicitly, the variational state prepared by a p-layer QAOA ansatz can be expressed:

$$|\Psi_{\text{QAOA}}(\boldsymbol{\beta}, \boldsymbol{\gamma})\rangle = e^{-\frac{i\beta_{p-1}}{2} H_B} e^{-\frac{i\gamma_{p-1}}{2} H_C} \ldots e^{-\frac{i\beta_0}{2} H_B} e^{-\frac{i\gamma_0}{2} H_C} |+\rangle^{\otimes n} , \tag{8}$$

where the variational parameters $\beta, \gamma \in \mathbf{R}^p$ are often referred to as "QAOA angles." They are optimized in order to minimize an empirical cost function, estimated by repeatedly preparing the quantum state and measuring it in the computational basis. The expected energy achieved by the state: $\langle \Psi_{\text{QAOA}}(\boldsymbol{\beta}, \boldsymbol{\gamma}) | H_C | \Psi_{\text{QAOA}}(\boldsymbol{\beta}, \boldsymbol{\gamma}) \rangle$ is the most frequently used such function, but other candidates have been reported, including the "CVar" [20] (average over energies after discarding samples with energy above a certain threshold) or the Gibbs free energy [19]. Once satisfying variational parameters have been determined, preparing the corresponding QAOA state in Eq. (8) and measuring it in the computational basis (ideally) provides good approximate solutions to the original combinatorial problem.

C. Random k-SAT and QAOA

This work considers the performance of the quantum approximate optimization algorithm on the random k-SAT combinatorial optimization problem. An instance of k-SAT is a formula on n Boolean variables $x_0, \ldots, x_{n-1}$, which is a *conjunction* of m *clauses*; conjunction means the formula is satisfied if and only if all clauses are. Besides, each clause is a *disjunction* of k *literals*, where a literal is a Boolean variable or its negation; disjunction means the clause is satisfied if and only if at least one of its literals is. Such a formula is said to be in *conjunctive normal form* (abbreviated CNF), meaning it is expressed as a conjunction of disjunctions. An example of a k-SAT formula with $k = 2$ and $n = 4$ is

$$(x_0 \vee \overline{x_1}) \wedge (x_1 \vee x_2) \wedge (x_1 \vee \overline{x_3}) , \tag{9}$$

where the bar over a Boolean variable denotes negation and $\vee$ (respectively, $\wedge$) denotes disjunction (respectively,

conjunction). The formula above is, for instance, satisfied by the assignment $x_0 = x_1 = 1, x_2 = x_3 = 0$. An algorithmically interesting setting for k-SAT is when instances are generated at random with a number of clauses m proportional to the number of variables n, where the ratio $r := (m/n)$ is known as *clauses-to-variables ratio*. In random k-SAT, the existence of solutions to a random k-SAT instance and the hardness of finding it are determined by r [21]. In our case, rather than fixing m to be a constant multiple $\lfloor rn \rfloor$ of n, we sample it from a distribution of expectation rn peaked around this mean, namely Poisson(rn). This technical choice allows the success probability of random k-SAT QAOA to be written as a generalized multinomial sum of the form of Eq. (A30), making its analysis possible via the saddle-point method, the correctness of which is proved in detail in the Supplemental Material [15].

Definition 1 (Random k-SAT problem). Let $k \geq 1$ an integer and $r > 0$. The random k-SAT problem is a constraint satisfaction problem on n variables, temporarily denoted by $x_0, \ldots, x_{n-1}$ for convenience. A random instance of such a problem is defined as follows:

(1) Sample $m \sim \text{Poisson}(rn)$.
(2) Generate m random OR clauses $\sigma = (\sigma_0, \ldots, \sigma_{m-1})$. Each clause consists of k literals chosen uniformly (with replacement) from $\{x_0, \overline{x_0}, x_1, \overline{x_1}, \ldots, x_{n-1}, \overline{x_{n-1}}\}$. The OR clause is satisfied if and only if at least one of its literals is.
(3) The random instance thereby generated, characterized by $\sigma = (\sigma_0, \ldots, \sigma_{m-1})$, is satisfied if and only if all OR clauses σ_j are.

A problem instance generated from this random ensemble will be denoted by $\sigma \sim \text{CNF}(n, k, r)$. Besides, for $y = (y_0, \ldots, y_{n-1}) \in \{0,1\}^n$, one denotes

$$y \vdash \sigma. \tag{10}$$

to signify that assignment y of the literals satisfies all clauses in σ.

Definition 2 (Random k-SAT QAOA). Let $k \geq 1$ an integer, $r > 0$ and n a positive integer. Given a random k-SAT instance $\sigma = (\sigma_0, \ldots, \sigma_{m-1}) \sim \text{CNF}(n, k, r)$ generated according to Definition 1, we denote by

$$H[\sigma] := \sum_{y \in \{0,1\}^n} \left|\left\{j \in [m] : y \nvdash \sigma_j\right\}\right| |y\rangle\langle y| \tag{11}$$

the diagonal quantum Hamiltonian corresponding to the classical cost function counting the number of unsatisfied clauses in $\boldsymbol{\sigma}$. The diagonal elements of this Hamiltonian are in $\{0, \ldots, m\}$. For each $m' \in \{0, \ldots, m\}$, we then denote by

$$\left\{H[\sigma] = m'\right\} := \sum_{\substack{y \in \{0,1\}^n \\ \left|\left\{j \in [m] : y \nvdash \sigma_j\right\}\right| = m'}} |y\rangle\langle y| \tag{12}$$

the orthogonal projector onto the eigenspace of $H[\boldsymbol{\sigma}]$ of eigenvalue m'. In particular, $\{H[\boldsymbol{\sigma}] = 0\}$ is the orthogonal projector onto the space of satisfying assignments. Besides, for $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbf{R}^p$, we denote by

$$|\Psi(\boldsymbol{\sigma}, \boldsymbol{\beta}, \boldsymbol{\gamma})\rangle := e^{-\frac{i\beta_{p-1}}{2}\sum_{j\in[n]} X_j} e^{-\frac{i\gamma_{p-1}}{2} H[\boldsymbol{\sigma}]} \ldots e^{-\frac{i\beta_0}{2}\sum_{j\in[n]} X_j} e^{-\frac{i\gamma_0}{2} H[\boldsymbol{\sigma}]} |+\rangle^{\otimes n}. \tag{13}$$

the state prepared by level-p QAOA for the optimization problem defined by Hamiltonian $H[\boldsymbol{\sigma}]$.

III. RESULTS

A. Theoretical results

The main technical result of this work, stated in the Supplemental Material [15] in Proposition 7, allows estimation of the leading exponential contribution of "generalized multinomial sums" (precisely defined in Definition 3), extending the standard multinomial theorem

$$\sum_{\substack{n_0,\ldots,n_{r-1} \\ n_0+\ldots+n_{r-1}=n}} \binom{n}{n_0, \ldots, n_{r-1}} \prod_{j\in[r]} x_j^{n_j} = \left(\sum_{j\in[r]} x_j\right)^n. \tag{14}$$

The proof of this generalized multinomial theorem uses the saddle-point method, whereby the generalized multinomial sum is expressed as an integral, whose exponential scaling is controlled by the unique critical point of the integrand; with our methods, the existence and uniqueness of the critical point requires certain parameters in the sum to be sufficiently small. Now, as we show in Proposition 3, the expected success probability of QAOA on random k-SAT for fixed variational parameters $\boldsymbol{\beta}, \boldsymbol{\gamma}$ can be cast as a generalized multinomial sum. Now, Appendix A sketches how such a sum can be estimated via the saddle-point method—with complete proofs deferred to the Supplemental Material. In the context of QAOA, the "small parameters" assumption required by Proposition 7 in the Supplemental Material translates to small $\boldsymbol{\gamma}$ angles; however, $\boldsymbol{\beta}$ is allowed to take any finite value.

Proposition 1 (Average-case success probability of random k-SAT QAOA by the saddle-point method). Let $q \geq 1$ an integer, $p \geq 1$ an integer and $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbf{R}^p$. For $\boldsymbol{\gamma}$ sufficiently small (i.e., smaller than a constant independent of the problem size n), the expected success probability

of random-2^q-SAT QAOA admits the following scaling exponent in the infinite-size limit:

$$\lim_{n\to\infty}\frac{1}{n}\log \mathbf{E}_{\boldsymbol{\sigma}}\left[\langle\Psi(\boldsymbol{\sigma},\boldsymbol{\beta},\boldsymbol{\gamma})|\mathbf{1}\left\{H[\boldsymbol{\sigma}]=0\right\}|\Psi(\boldsymbol{\sigma},\boldsymbol{\beta},\boldsymbol{\gamma})\rangle\right] = F\left(z^*\right)+\left(2^q-1\right)\sum_{\alpha\subset[2p+1]}\left(\frac{\partial F}{\partial z_\alpha}\left(z^*\right)\right)^{2^q}, \tag{15}$$

where F is a complex-valued function of 2^{2p+1} variables, each variable being indexed by a subset of $[2p+1]$. The explicit definition of F is as follows:

$$F\left((z_\alpha)_{\alpha\subset[2p+1]}\right) := \log\sum_{s\in\{0,1\}^{2p+1}} b_s \exp\left((r/2)\sum_{\substack{\alpha\subset[2p+1]\\ \forall j,j'\in\alpha, s_j=s_{j'}}}(-c_\alpha)^{1/2^q} z_\alpha\right), \tag{16}$$

where

$$b_s = \frac{1}{2}\prod_{j\in[p]}\langle s_j|e^{\frac{i\beta_j}{2}X}|s_{j+1}\rangle\,\langle s_{2p-j-1}|e^{-\frac{i\beta_j}{2}X}|s_{2p-j}\rangle, \tag{17}$$

$$c_\alpha := (-1)^{\mathbf{1}[p\in\alpha]}\prod_{j\in\alpha:j<p}\left(e^{-\frac{i\gamma_j}{2}}-1\right)\prod_{j\in\alpha:j>p}\left(e^{\frac{i\gamma_{2p-j}}{2}}-1\right), \tag{18}$$

The definition of b_s involves products of single-qubit matrix elements, whose explicit values are given by

$$\langle s|e^{i\beta X}|s'\rangle = \begin{cases}\cos\beta & \text{if } s=s'\\ i\sin\beta & \text{if } s\neq s'\end{cases}. \tag{19}$$

Finally, $z^*\in\mathbf{C}^{2^{2p+1}}$, where F and its derivatives are evaluated, is the unique fixed point of function

$$\begin{array}{ccc}\mathbf{C}^{2^{2p+1}} & \longrightarrow & \mathbf{C}^{2^{2p+1}}\\ (z_\alpha)_{\alpha\subset\mathbf{Z}_{2p+1}} & \longmapsto & \left(-2^q\left(\frac{\partial F}{\partial z_\alpha}(z)\right)^{2^q-1}\right)_{\alpha\subset\mathbf{Z}_{2p+1}}\end{array}. \tag{20}$$

The existence and uniqueness of the fixed point are guaranteed for sufficiently small $\boldsymbol{\gamma}$.

A generalization of the multinomial theorem was already derived in Ref. [14] in the context of QAOA applied to Max-k-XOR, though with distinct assumptions and a very different method. Besides, while the results from [14] apply to arbitrary variational angles in the context of Max-k-XOR, our analysis for random-k-SAT is (in principle) limited to sufficiently small $\boldsymbol{\gamma}$ angles. However, as will be extensively discussed in Sec. IV A, the method empirically appears to be quantitatively accurate for a sufficiently wide range of angles, most interestingly for the optimal ones.

While Proposition 1 establishes a rigorous scaling for the expected success probability of QAOA on random k-SAT under certain assumptions, this quantity may not be the most natural to consider to benchmark QAOA against other algorithms. In fact, it may be more natural to consider the *median running time* of the algorithm, which is a common method to benchmark classical SAT solvers, see e.g., Ref. [3]. The choice of the median running time as a benchmark, as opposed, for instance, to the expected or maximum running time, addresses an important difficulty: since QAOA is based on sampling bitstrings from the quantum state until one finds a satisfying assignment, the algorithm will never terminate if the problem instance is unsatisfiable. This would lead to an infinite expected running time as soon as a randomly generated instance has a finite probability of being unsatisfiable, which is the case for the random ensemble introduced in Definition 1. In contrast, the median running time will remain finite and is besides less sensitive to outliers with finite, yet unusually large, running time.

In addition, the expected success probability gives a lower bound on the median running time. First, the expected success probability can be related to the median success probability μ via the following straightforward argument:

$$\begin{aligned}&\mathbf{E}_{\boldsymbol{\sigma}}\left[p_{\mathrm{succ}(\boldsymbol{\sigma})}\right]\\ &=\sum_{\boldsymbol{\sigma}}P(\boldsymbol{\sigma})p_{\mathrm{succ}}(\boldsymbol{\sigma})\\ &=\sum_{\boldsymbol{\sigma},p_{\mathrm{succ}}(\boldsymbol{\sigma})<\mu}P(\boldsymbol{\sigma})p_{\mathrm{succ}}(\boldsymbol{\sigma})+\sum_{\boldsymbol{\sigma},p_{\mathrm{succ}}(\boldsymbol{\sigma})\geq\mu}P(\boldsymbol{\sigma})p_{\mathrm{succ}}(\boldsymbol{\sigma})\\ &\geq\sum_{\boldsymbol{\sigma},p_{\mathrm{succ}}(\boldsymbol{\sigma})\geq\mu}P(\boldsymbol{\sigma})p_{\mathrm{succ}}(\boldsymbol{\sigma})\geq\frac{\mu}{2}.\end{aligned} \tag{21}$$

Hence

$$\frac{1}{\mathbf{E}_{\boldsymbol{\sigma}}\left[p_{\mathrm{succ}(\boldsymbol{\sigma})}\right]}\leq\frac{2}{\mu} \tag{22}$$

and the latter quantity is upper bounded by twice the median running time via Jensen's inequality for medians. Namely, this inequality, precisely stated and derived in Ref. [22], informally says

$$f\ (\text{median of}\ X)\leq\text{median of}\ f\ (X) \tag{23}$$

for a convex function f and an appropriate random variable X. In this case, we used the convexity of $x\longrightarrow$

$1/x$ and the fact that for all problem instances, the expected running time is exactly the inverse of the success probability.

B. Validation of the analytic algorithm

Next we validate the theoretical formula given by Proposition 1 for the expected success probability of QAOA on a random k-SAT instance.

First, we compare the limiting average success probabilities for random k-SAT predicted by Proposition 1 with actual average success probabilities determined by numerical experiments for small n. We sample up to 10000 instances from $\mathrm{CNF}(n,k,r)$ for each value of k and problem size $n \in \{12,\ldots,20\}$ and retain only satisfiable instances. QAOA is then *evaluated* (and not trained) on each of these instances using angles previously determined to achieve a good average success probability, as detailed in Sec. IV A below. Note that the instances used to train QAOA are much less numerous (100) and smaller (12) than the ones used to validate the performance here. For each k and n, we compute the average success probability and median running time on the relevant set of random instances. For each problem size n, the instances generated for evaluation achieve an empirical uncertainty of order $< 0.5\%$ on the expected success probability at size n. This translates to an error of order 10% with the 100 instances used for training, confirming the latter provide a rather coarse approximation of the success probability. The results are shown in Fig. 1 for the case $k = 8$, where we perform a linear least-squares fit on the experimental data and compare against the scaling predicted from the theoretical results. As the constant factor in this scaling is unknown, we assume that this is equal to 1 in the plot.

Second, we exploit the fact, established in Proposition 4, that for $p = 1$ the expected success probability of random k-SAT QAOA at finite size n can be computed in time $O(n^3)$, allowing for a practical evaluation, and even optimization of the expected QAOA success probability for large instance sizes of order 100. Unlike the analytic prediction for the infinite-size scaling exponent, the finite-size calculation at $p = 1$ applies to all k (not only k a power of 2) and arbitrary angles (not only sufficiently small $\boldsymbol{\gamma}$). We may therefore extract the empirical scaling exponent of this expected success probability by an exponential fit and compare it with the infinite-size scaling exponent predicted by Proposition 1. Here, the empirical scaling of the success probability at n is defined as the ratio between success probabilities at size $n+1$ and n, taken to the logarithm:

$$\log \frac{\mathbf{E}_{\boldsymbol{\sigma}\sim\mathrm{CNF}(n+1,k,r)}\left[\left\langle \Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\beta,\gamma)\middle|\mathbf{1}\left\{H[\boldsymbol{\sigma}]=0\right\}\middle|\Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\beta,\gamma)\right\rangle\right]}{\mathbf{E}_{\boldsymbol{\sigma}\sim\mathrm{CNF}(n,k,r)}\left[\left\langle \Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\beta,\gamma)\middle|\mathbf{1}\left\{H[\boldsymbol{\sigma}]=0\right\}\middle|\Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\beta,\gamma)\right\rangle\right]}. \tag{24}$$

Although less robust than an exponential fit, this metric is usable in this context as the expectations can be evaluated exactly. Besides, it has the advantage of sharply capturing the *local* scaling of the success probability around each n. The result of this comparison (between *excess scaling exponents* rather than exponents themselves, see Sec. IV A) is represented for k-SAT, $k \in \{2,4,8,16\}$, with instance sizes ranging from 10 to 70, in Fig. 1. The plot shows that, as expected, the error decreases as n increases. For a fixed size, the relative error incurred by the finite-size approximation worsens as k increases. However, the asymptotic decay rate of the error seems comparable between the different k values considered.

While these comparisons were performed for (empirically) optimal average-instance parameters, it is also instructive to consider the full optimization landscape of $p = 1$ QAOA for the same values of k, to determine how well the analytic and empirical results match. In addition, one may wonder whether the empirically optimal parameters are close to the limiting optimal parameters. Our experimental results to address these questions are included in Appendix B.

C. Algorithm scaling

Having developed confidence that the analytic and empirical scaling exponents are close, we use our analytic formulae to determine the behavior of the exponent in terms of p (for a fixed set of parameters, determined for each p using a small-scale experiment). This behavior strongly suggests power-law decay; performing a fit to the data allows us to extrapolate the performance of QAOA to larger values of p than are accessible to our algorithm. The results are shown in Fig. 2.

We also studied whether the inverse of the expected success probability provides an accurate reflection of the median running time, by comparing these two quantities in numerical experiments. Using an exponential fit, we extract a scaling exponent for both the success probability and the median running time as functions of n. Observe that while the success probability was shown to admit a scaling exponent for sufficiently small variational angles in Proposition 1, no such rigorous statement exists for the median running time; the exponent for the latter quantity should therefore be regarded as purely empirical. The results are also shown in Fig. 2 for various choices of p.

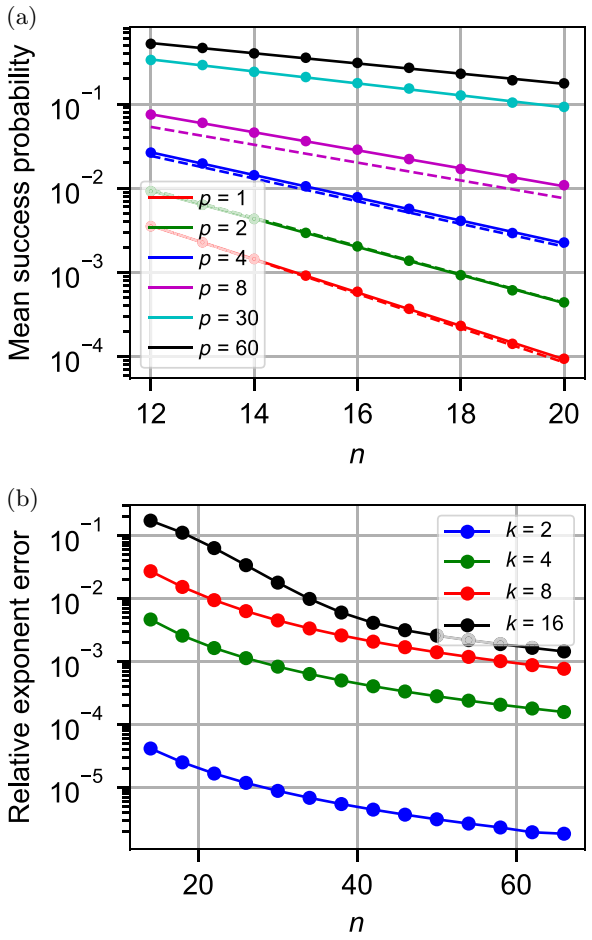


FIG. 1. Comparison of numerical results to limiting theoretical predictions. Top: points are empirical averages. Solid lines are fits to empirical averages, dashed lines are scaling predicted by theory (assuming unknown constant factor is 1). Error bars are too small to be seen. Bottom: points are relative differences in excess scaling exponents between numerical and theoretical results, solid lines are added to guide the eye. (a) $k = 8$, varying p; (b) $p = 1$, varying k.

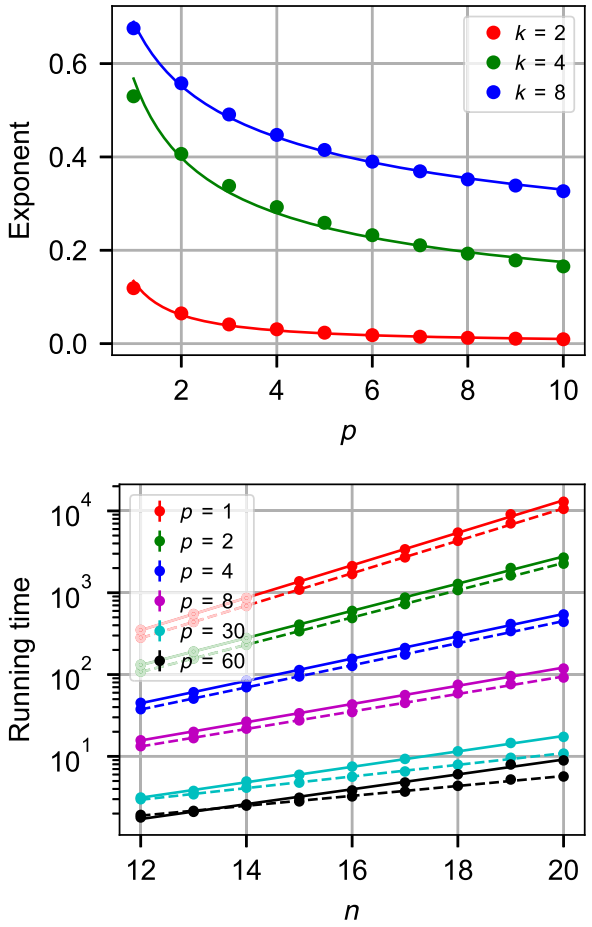


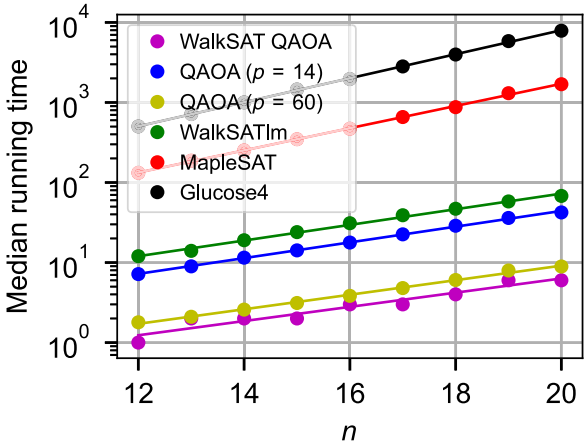
FIG. 2. Scaling behavior of QAOA on random k-SAT. Top: analytic scaling exponents c in terms of p, such that success probability is predicted to be 2^{-cn} up to lower-order terms. Fit to a power law for each k. Fits are $c \approx 0.13p^{-1.12}$ ($k = 2$), $c \approx 0.57p^{-0.51}$ ($k = 4$), $c \approx 0.69p^{-0.32}$ ($k = 8$). Bottom: median running time (solid line) compared with running time estimated from average success probability for random 8-SAT instances (dashed line). Lines are linear fits. Error bars are too small to be seen.

We observe that for small p, the two complexity measures are well aligned, while for large p, their slope appears to differ. One possible explanation for this divergence is that training to maximize the average success probability does not necessarily minimize the median running time. This may be a particular issue in the scenario where p is large and n is small, because the QAOA success probability may be close to 1 for many "easy" instances. Optimizing the average success probability may lead to finding parameters that are good for these easy instances, while performing poorly for harder instances. For example, taking $p = 60$, $k = 8$, $n = 12$, the mean success probability was approximately 0.56.

All in all, these results seem to provide theoretical backing for the approach described in Sec. IV A below of obtaining good average-case parameters for QAOA by estimating averages empirically on a small dataset of modest size instances.

D. Comparison of fixed-parameter QAOA with classical SAT solvers

Having built up confidence that our limiting theoretical results are well aligned with numerical benchmarks for small n, we compare the performance of QAOA to a variety of classical solvers for k-SAT. We choose to focus on the case $k = 8$, motivated by a trade-off between the need for a sufficiently large k (making the problem hard enough for classical solvers) and the practical requirement



Solver	Fit	Error
WalkSAT QAOA	$-3.232 + 0.295n$	0.011
QAOA ($p = 14$)	$-1.064 + 0.326n$	0.008
QAOA ($p = 60$)	$-2.842 + 0.302n$	0.007
`walksatlm`	$-0.309 + 0.325n$	0.008
`maplesat`	$1.531 + 0.461n$	0.004
`glucose4`	$2.998 + 0.498n$	0.005

FIG. 3. Scaling behavior of median running times of selected classical and quantum algorithms for 8-SAT. WalkSAT QAOA uses $p = 60$.

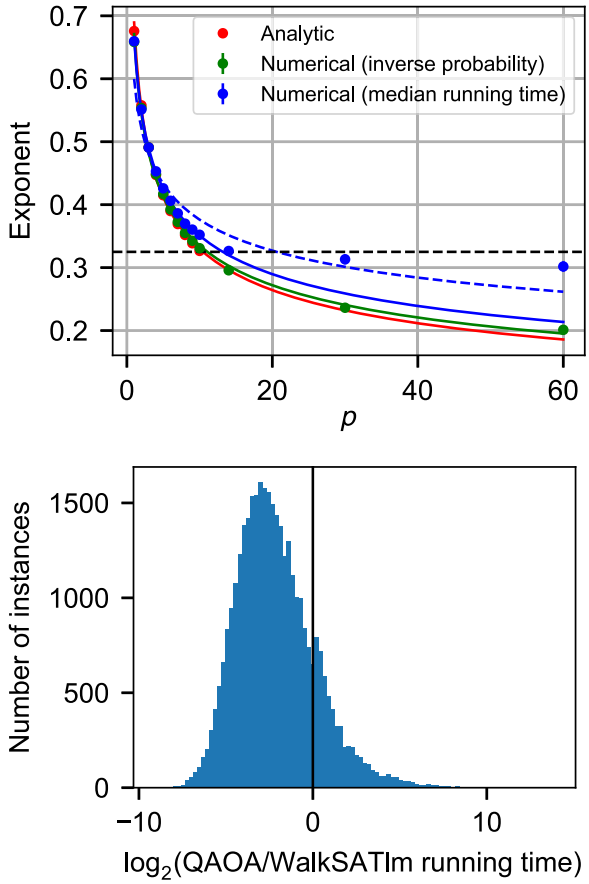


FIG. 4. Running times of QAOA compared with WalkSATlm for random 8-SAT. Top: scaling exponent α in running time approximately $2^{\alpha n}$ for QAOA estimated by inverting analytic ($p \leq 10$) and numerical ($p \leq 60$) results on average probabilities, and from numerical results on median running times for $n \in \{12, \ldots, 20\}$. Horizontal line is experimentally estimated WalkSATlm median running time scaling exponent. Other lines are fits. Blue dashed line is fitting based on all p, blue solid line is using $p \leq 10$. Error bars are too small to be seen. Bottom: histogram of ratios of running times of QAOA ($p = 60$) and WalkSAT for $n = 20$ instances.

to store all generated instances (recalling that the clauses-to-variables ratio at satisfiability threshold increases exponentially with k [17]), together with our theoretical results only being available for k a power of 2.

We benchmarked QAOA against the simple local search algorithm WalkSAT [23–25], the optimized local search algorithm WalkSATlm [12] and the suite of state-of-the-art SAT solvers pySAT [26]. WalkSATlm has demonstrated leading performance on random 5-SAT and 7-SAT instances with clauses to variables ratio close to the satisfiability threshold [12]. In the most recent SAT competition, which included a track for randomly generated instances [27], although WalkSATlm did not compete, the winning solver was based on the Sparrow solver, which performed significantly less well than WalkSATlm in previous experiments [12]. We additionally considered comparing QAOA to the well-known *survey propagation algorithm* introduced in Ref. [28]. This work empirically showed this message passing algorithm to outperform competitors close to the satisfiability threshold. However, these conclusions were only reported for relatively small k (3 or 4), and initial experiments carried out at $k = 8$ using a publicly available implementation [29] revealed the algorithm to be impractical to run at $k = 8$ due to the high number of constraints per variable and the large degree of the constraint graph. We therefore did not include this contender in our comparisons.

We also studied a combination of QAOA and WalkSAT, whereby assignments sampled from QAOA (hence, not necessarily satisfying) are given as a starting guess to WalkSAT. More precisely, the algorithm consists of sampling an assignment from QAOA, apply a single iteration of WalkSAT [consisting of a $\Theta(n)$ steps walk], and declare success or failure according to whether the walk

TABLE I. Random k-SAT satisfiability thresholds for values of k considered in this work.

k	Satisfiability threshold
2	1.0
4	9.93
8	176.54
10	708.92
16	45425.2

updated the initial assignment to a satisfying one. Similarly to WalkSAT, this process is repeated until success. Each given instantiation of the algorithm has a success probability, and the expected number of instantiations is the inverse of this probability.

Similarly to results presented in Sec. III B, we extracted the median running time of the solver, with respect to our randomly generated instances, for each instance size $n \in \{12, \ldots, 20\}$ before determining an estimated scaling exponent based on a fit to this data. We illustrate this method for four of the most efficient classical solvers, together with some quantum ones, in Fig. 3; a more complete list of empirical scaling exponents is given in Table II in Appendix B 2. We found that, in each case, a simple exponential fit was highly accurate. Consistent with previous experiments [12], WalkSATlm's median running time, as measured by the number of input formula evaluations, was the lowest among the solvers considered. In addition, an instance-by-instance analysis determined that WalkSATlm was the most efficient algorithm on all but a few percent of the instances considered. In the case of quantum algorithms, we found that the combination of WalkSAT and QAOA was not substantially more efficient than the use of QAOA alone: for example, as seen in Fig. 3, the two scaling exponents are within error bars for $p = 60$. We therefore focused on the comparison between WalkSATlm and QAOA.

Figure 4 shows the result of this comparison. We found that for $p \approx 14$, QAOA matches the performance of WalkSATlm. For larger p, the predicted running time scaling based on our theoretical bounds on success probability well matches the predicted scaling from empirical success probabilities. However, the median running time diverges from this prediction, and outperforms WalkSATlm more modestly. As discussed in the previous section, one possible cause of this is that the QAOA parameters used, which optimize the average success probability for small n, are less effective at optimizing the median running time for larger n. Further experiments would be required to determine whether this is the case, or if there truly is a gap between the running time predicted by the average success probability, and the median running time. We also show in Fig. 4 a histogram of the ratios between running times of QAOA ($p = 60$) and WalkSATlm for $n = 20$ instances. We observe that there are rather substantial tails, corresponding to instances where one solver or the other has a large advantage over its counterpart.

IV. DISCUSSION

Our results suggest that it may be possible to outperform the scaling of state-of-the-art classical algorithms for solving constraint satisfaction problems using a relatively simple and low-depth quantum algorithm. However, achieving a speedup using a quantum computer in practice depends on many other factors, including the performance of the hardware in terms of parameters such as clock speed and latency, and any overheads needed for quantum error correction.

The approach studied in this work is likely to be significantly more suitable for near-term quantum computers because of the low depth of the quantum circuits considered, when compared with the exponential-depth circuits required for quantum algorithms like Grover's algorithm. Yet outperforming classical computers is likely to require higher-performance hardware than is currently available, or problem instances, which demonstrate a larger separation than we have shown for random k-SAT. In particular, in this work we did not consider the effect of errors on the quantum algorithm's success probability, because the problem sizes required to achieve a quantum-classical separation, while far smaller than suggested by previous work, are large enough to be out of reach without quantum error correction. However, there is significant scope for algorithmic improvements to reduce this crossover point.

Our theoretical techniques are most importantly limited by the specific form required of the generalized multinomial sum representing the algorithm's success probability. This limits the number of constraint satisfaction problems accessible to our approach; for instance, our techniques would not apply to Max-k-XOR, where the performance of QAOA was recently studied using different techniques [14]. Unifying these techniques to capture a broader variety of constraint satisfaction problems is an interesting direction for future work. In addition, while our analysis captures the average success probability of the algorithm, it falls short of predicting more interesting metrics such as the median or mean running time, as further discussed in Methods. Finally, the argument establishing the existence of an exponential scaling crucially relies on the ansatz depth p being constant, leaving open the practically still relevant cases $p = O(\log n)$ or $p = \mathrm{poly}(n)$. Addressing these points would be a significant step towards characterizing the power of QAOA, and perhaps to finding larger quantum speedups for problems of practical interest.

A. Methods

1. Random k-SAT instances

Random k-SAT instances are generated from the random ensemble described in Definition 1. When not explicitly specified, the clauses-to-variables ratio r is set to (an approximation of) the satisfiability threshold. For $k \leq 15$, we use the same reference values for this threshold as Ref. [3], while for $k = 16$, we estimate the threshold using the third-order expansion from Ref. [17, Appendix], in agreement with the method used by the former work for lower k. For reference, we report in Table I the relevant threshold values used in our numerical study.

2. Comparison with classical algorithms

WalkSAT is an algorithm based on a randomized local search approach. Given an assignment which does not satisfy the formula, the algorithm picks a clause, which is not satisfied, picks one of the variables within that clause, and flips it. Various strategies have been proposed for choosing clauses and variables; here we simply pick the clause at random (among unsatisfied clauses) and pick a random variable within that clause to flip. WalkSATlm is a modification of WalkSAT with a more complex cost function. For WalkSATlm, trial and error led to the choice of hyperparameters $p = 0.15, w_1 = 6, w_2 = 5$, see Ref. [12] for their detailed meaning. For QAOA, we used pseudo-optimal parameters obtained from 100 size 12 random problem instances as described in Sec. IV A.

3. Parameter optimization

Since QAOA is a variational algorithm depending on angles $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$, all numerical experiments require either optimization of these parameters or evaluation at well-chosen guessed parameters. In this work, we choose the second option to demonstrate the potential of using QAOA without classical optimization loop. We therefore look for variational parameters achieving good success probability on an average instance. More precisely, this means identifying parameters that maximize the expected success probability of QAOA on a random instance of k-SAT.

Given mixer angles $\boldsymbol{\beta}$, this probability is evaluated analytically for small enough $\boldsymbol{\gamma}$ by Proposition 4. Besides, under this small-angle assumption, the fixed-point finding procedure is numerically efficient, requiring only time $\mathcal{O}(\log(1/\varepsilon))$ for a tolerated error ε (see discussion of the Hamming weight squared toy example in Appendix A 2 for an explicit illustration of this fact). This suggests to use Proposition 4 and its implementation detailed in Algorithms 1 and 2 to perform this average-instance optimization. Unfortunately, this approach presents two difficulties. First, we do not know whether optimal parameters are sufficiently small that Proposition 4 applies. Second, even if this assumption was satisfied, we empirically observed that for moderately large p, e.g., $p \geq 5$, variational optimization could not be performed in reasonable time (with gradient descent, allowing hundreds of iterations and a time budget of order hour) using the analytic algorithm to evaluate the success probability.

We therefore chose to estimate the expected success probability of QAOA empirically, using sets of randomly generated instances. More specifically, given a number of variables per clause k, a clauses-to-variables ratio r and a finite instance size n, one generates a set of 100 random instances sampled from $\mathrm{CNF}(n, k, r)$. Then, for each set of angles $\boldsymbol{\beta}, \boldsymbol{\gamma}$, the fixed-parameters, average-instance success probability of QAOA is estimated by empirically averaging the success probability of QAOA over the set of instances; angles $\boldsymbol{\beta}, \boldsymbol{\gamma}$ are then updated accordingly. The set of instances does not change between iterations of the classical optimization algorithm. In this study, we initialize the optimization by setting all $\boldsymbol{\beta}$ to 0.01 and $\boldsymbol{\gamma}$ to -0.01, which we conjecture correspond to the correct signs of the optimal angles based on many optimization trials from randomly starting points. The small magnitude 0.01 was chosen following the observation that excessively large angles, e.g., 1, led to false maxima and barren plateaus, notably for large p. The classical optimal algorithm we used is a simple gradient descent, which we conjecture

TABLE II. Empirical exponential fits for all SAT solvers for 8-SAT. We further report the correlation coefficient of the fit. Besides, we estimate the error on the scaling exponent as described in Sec. IV A.

Solver	Fit	Correlation coefficient	Estimated exponent error
walksat_qaoa	$-3.232 + 0.295n$	0.963143	0.011
eval_qaoa ($p = 14$)	$-1.064 + 0.326n$	0.999422	0.005
eval_qaoa ($p = 60$)	$-2.842 + 0.302n$	0.998161	0.007
walksatlm	$-0.309 + 0.325n$	0.997503	0.007
maplesat	$1.531 + 0.461n$	0.999626	0.004
glucose3	$2.998 + 0.498n$	0.999826	0.004
glucose4	$2.998 + 0.498n$	0.999826	0.004
gluecard3	$2.998 + 0.498n$	0.999826	0.006
gluecard4	$2.998 + 0.498n$	0.999826	0.005
mergesat3	$2.974 + 0.500n$	0.999846	0.004
lingeling	$2.681 + 0.505n$	0.999700	0.005
cadical	$2.702 + 0.518n$	0.999645	0.005
minicard	$2.725 + 0.523n$	0.999689	0.004
minisat22	$2.725 + 0.523n$	0.999689	0.005
minisatgh	$2.725 + 0.523n$	0.999689	0.005
maple_chrono	$2.557 + 0.533n$	0.999814	0.005
maple_cm	$-0.713 + 0.581n$	0.989826	0.005
schoning	$-2.657 + 0.649n$	0.999826	0.006

always converges to the optimum from this initial guess. Besides, for each k and r, 100 random instances of size 12 were generated to evaluate the empirical average success probability.

A limitation of this empirical average method is that the average success probability is only approximated rather than exactly calculated. In fact, the very limited number of samples: 100 used for this approximation is not even sufficient to achieve acceptable statistical significance on the estimation of the success probability itself, as discussed in Sec. III B. Besides, individual instance sizes must be limited for the method to remain practical. Despite these apparent weaknesses, we find that the empirical technique provides near-optimal angles (at least for small p) when compared to the analytic one introduced in Proposition 1. This justifies *a posteriori* the approach of determining near-optimal angles from averages over few sample instances. Therefore, we always use angles optimized by this technique in the rest of the study.

The evaluation and optimization of QAOA were carried out using the Yao.jl quantum circuit simulation framework [30]. Among other advantages, this library combines execution speed and seamless integration of differentiable programming, making it particularly suitable for the study of variational quantum circuits.

4. Comparison metrics

In the presentation of our numerical results, we have made comparisons between different analytic and empirical methods to estimate the performance of QAOA on random k-SAT. We will also contrast QAOA against several classical algorithms tackling the k-SAT problem. We discuss here some aspects of the metrics used for these comparisons.

The main figure of merit we use to characterize QAOA is the scaling exponent of its success probability, averaged over problem instances. Such a scaling exponent is rigorously known to exist for sufficiently small $\boldsymbol{\gamma}$ according to Proposition 1. It can also be estimated through an exponential fit of the success probability against the instance size. In fact, rather than the scaling exponent itself, we occasionally consider its *excess* over the value it would take for random assignment. Recalling that random assignment is the special case of QAOA where $\boldsymbol{\beta} = \mathbf{0}$ or $\boldsymbol{\gamma} = \mathbf{0}$, Eq. (A8) gives a scaling exponent $-2^{-k}r$ in this case; therefore, we systematically subtract this quantity from all scaling exponents and call the resulting values *excess scaling exponents*. This adjustment prevents excessive optimism when comparing different algorithms to estimate scaling exponents. Precisely, in the case that QAOA does little better than random assignment (for instance, for small angles) and two exponent estimation algorithms capture this feature, these methods will simultaneously return a value close to $-2^{-k}r$, leading to a small relative error between the methods. However, should they differ more substantially when it comes to the excess scaling exponents, this would go relatively unnoticed if only comparing exponents.

We now discuss how the running time on a problem instance is quantified for QAOA and classical algorithms. In both cases, we used the median running time over randomly generated instances as a figure of merit. However, the running time of the algorithm on a single instance is defined differently for QAOA and classical solvers. In the case of QAOA, the instance running time is simply defined as the inverse $1/(p_{\mathrm{succ}})$ of the success probability p_{succ} (probability of sampling a satisfying assignment) of QAOA on this instance. This is indeed the expected number of samples one needs to draw to obtain a satisfying assignment, and corresponds to the "time-to-solution" (TTS) metric used in experimental comparison of algorithms, taking a success probability of $1/2$. As for classical solvers, the instance running time is understood as the number of evaluations of the Boolean formula defining the SAT problem. For algorithms from the pySAT suite [26], this number is determined according to the information returned by the library after execution of the solver.

5. Exponential fits

Our main analytic result for random k-SAT QAOA, Proposition 1, predicts an exponential scaling in the infinite size limit for the average-instance fixed-parameter success probability. To compare this result to numerical experiments, empirical scaling exponents need to be extracted using an exponential fit. In practice, in this work, the exponential fit is a least-squares linear regression on the logarithm of the quantity to fit. For each problem size, we obtain an empirical average success probability by averaging over instances. The fit is then performed on these empirical average success probabilities as a function of problem size. To estimate the error on the parameters returned by the fit (in particular, on the scaling exponent), one uses resampling. Precisely, one recalculates empirical average success probabilities using only half of the sample instances, where the half is chosen uniformly at random. This resampling process is repeated several times (typically 100), leading to a probability distribution for the fit parameters. The error on each fit parameter is then estimated as the standard deviation of the distribution of this parameter.

ACKNOWLEDGMENTS

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant Agreement No. 817581). This work was supported by the EPSRC Centre for Doctoral Training in Delivering Quantum Technologies, Grant Ref. EP/S021582/1.

APPENDIX A: DERIVATION OF ANALYTIC FORMULAE

In this section, we derive an integral representation for the average-instance success probability of k-SAT QAOA given in Proposition 1. This representation lends itself to estimation via the saddle-point method in the infinite instance size limit $n \longrightarrow \infty$. We only show here the results in the case of 2-SAT; the formulae for 2^q-SAT, $q \geq 2$ and the proof of correctness of the saddle-point method, as deferred to the Supplemental Material [15]. However, we start with a short derivation of an analogous result for a toy example (QAOA applied to the Hamming weight squared cost function), which gives an accurate flavor of the general method.

1. Random k-SAT QAOA expectations as generalized multinomial sums

We establish an expression for the expected success probability of random k-SAT QAOA (Definition 2) using a slight generalization of Proposition 28 from Ref. [11]. According to this proposition, recalled here with adapted notations.

Proposition 2. Let a random constraint satisfaction problem be defined by a set of clauses $\boldsymbol{\sigma}$ and a diagonal quantum Hamiltonian $H[\boldsymbol{\sigma}]$; one temporarily denotes $\boldsymbol{\sigma} \sim \mathrm{CSP}(n)$ for a set of clauses on n variables sampled from the random ensemble. For $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbf{R}^p$, let $|\Psi(\boldsymbol{\beta}, \boldsymbol{\gamma})\rangle$ the state prepared by level-p QAOA for this combinatorial optimization problem. Assume that for all $y^{(0)}, \ldots, y^{(2p)} \in \{0,1\}^n$,

$$\mathbf{E}_{\boldsymbol{\sigma}\sim\mathrm{CSP}(n)} \left[\bigotimes_{j\in[p]} e^{\frac{i\gamma_j}{2} H[\boldsymbol{\sigma}]} \otimes \{H[\boldsymbol{\sigma}] = 0\} \otimes \bigotimes_{j\in[p]} e^{-\frac{i\gamma_{p-1-j}}{2} H[\boldsymbol{\sigma}]} \right] \bigotimes_{j\in[2p+1]} |y^{(2p-j)}\rangle \tag{A1}$$

depends only on the numbers:

$$n_s := \left| \left\{ i \in [n] : \forall j \in [2p+1], y_i^{(j)} = s_j \right\} \right|, \left(s \in \{0,1\}^{2p+1}\right). \tag{A2}$$

[Note that the quantity in Eq. (A1) is always colinear to $\bigotimes_{j\in[2p+1]} |y^{(2p-j)}\rangle$ since $H[\boldsymbol{\sigma}]$ is diagonal.] In this case, we introduce the notation:

$$E\left((n_s)_{s\in\{0,1\}}\right) := \bigotimes_{j\in[2p+1]} \langle y^{(2p-j)}| \, \mathbf{E}_{\boldsymbol{\sigma}\sim\mathrm{CSP}(n)} \left[\bigotimes_{j\in[p]} e^{\frac{i\gamma_j}{2} H[\boldsymbol{\sigma}]} \otimes \{H[\boldsymbol{\sigma}] = 0\} \otimes \bigotimes_{j\in[p]} e^{-\frac{i\gamma_{p-1-j}}{2} H[\boldsymbol{\sigma}]} \right] \bigotimes_{j\in[2p+1]} |y^{(2p-j)}\rangle . \tag{A3}$$

Then,

$$\mathbf{E}_{\boldsymbol{\sigma}\sim\mathrm{CSP}(n)} \langle \Psi(\boldsymbol{\sigma}, \boldsymbol{\beta}, \boldsymbol{\gamma}) | \{H[\boldsymbol{\sigma}] = 0\} | \Psi(\boldsymbol{\sigma}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \rangle = \frac{1}{2^n} \sum_{\substack{(n_s)_{s\in\{0,1\}^{2p+1}} \\ \sum_s n_s = n}} \binom{n}{(n_s)_s} \prod_{s\in\{0,1\}^{2p+1}} B_{\boldsymbol{\beta},s}^{n_s} E\left((n_s)_s\right), \tag{A4}$$

where

$$B_{\boldsymbol{\beta},s} := (-1)^{\mathbf{1}[s_0 \neq s_p]} \prod_{j\in[p]} \left(\cos\frac{\beta_j}{2}\right)^{\mathbf{1}[s_j = s_{j+1}] + \mathbf{1}[s_{2p-j} = s_{2p-j-1}]} \prod_{j\in[p]} \left(i \sin\frac{\beta_j}{2}\right)^{\mathbf{1}[s_j \neq s_{j+1}] + \mathbf{1}[s_{2p-j} \neq s_{2p-j-1}]} \tag{A5}$$

depends only on the mixing angles $\boldsymbol{\beta}$ (in particular, not on the dephasing angles $\boldsymbol{\gamma}$ or the constraint satisfaction problem) and

$$\binom{n}{(n_s)_{s\in\{0,1\}^{2p+1}}} = \frac{n!}{\prod_{s\in\{0,1\}^{2p+1}} n_s!} \tag{A6}$$

is a multinomial coefficient.

The idea of understanding QAOA by averaging over instances, leading to an expression in terms of *configuration basis numbers* $(n_s)_{s\in\{0,1\}^{2p+1}}$, was first proposed in Ref. [31]. This earlier work considered a distinct constraint satisfaction problem: the Sherrington-Kirkpatrick (SK) model and evaluated its fraction of satisfied constraints. This differs from the current work's approach of estimating the probability of finding a satisfying assignment. Both approaches are natural given a random k-SAT instance (below satisfiability threshold) is satisfiable with high probability while a SK instance is not. We will need the following lemma, which is a slight adaptation of the reasoning in Ref. [32, Section 3.1] to average over random k-SAT instances.

Lemma 1 (Averaging over random k-SAT clauses). Let $k \geq 1$ and $p \geq 1$ integers and let σ be an OR clause on n variables $x_0, \ldots, x_{n-1}$ sampled as defined in Definition 1, i.e., by choosing k literals uniformly at random among $\{x_0, \overline{x_0}, \ldots, x_{n-1}, \overline{x_{n-1}}\}$. Let J bitstrings (representing literal assignment) $y^{(j)} \in \{0,1\}^n, j \in [J]$.

$$\forall y \in \{0,1\}^n, \mathbf{E}_\sigma \left[\prod_{j\in[J]} \mathbf{1}\left[y^{(j)} \not\models \sigma\right] \right] = \left(\frac{\left|y^{(0)} \cap \ldots \cap y^{(J-1)}\right|}{2n} \right)^k, \tag{A7}$$

where $y^{(0)} \cap \ldots \cap y^{(J-1)} := \left\{ i \in [n] \,:\, y_i^{(0)} = y_i^{(1)} = \ldots = y_0^{(J-1)} \right\}$ is the set of indices where bitstrings $y^{(j)}$ all coincide. It is easily seen that in the degenerate case $J = 0$, it suffices to replace $\left|y^{(0)} \cap \ldots \cap y^{(J-1)}\right| \longrightarrow 2n$ in the equation above for it to remain correct.

Proof. It suffices to observe that for all k-literal OR clause σ, σ is simultaneously *unsatisfied* by $y^{(0)}, \ldots, y^{(J-1)}$ if and only if:

(1) all literals from σ have variables in $y^{(0)} \cap \ldots \cap y^{(J-1)}$;
(2) a variable appearing in the clause is negated if and only if it is set to 1 in $y^{(0)}, \ldots, y^{(J-1)}$ (the value must be common between these bitstrings by definition of $y^{(0)} \cap \ldots \cap y^{(J-1)}$).

Using this fact, and since the probability of choosing one such literal among $2n$ possible literals is $((\left|y^{(0)} \cap \ldots \cap y^{(J-1)}\right|)/2n)$, the probability of choosing k such literals is $\left(((\left|y^{(0)} \cap \ldots \cap y^{(J-1)}\right|)/2n)\right)^k$ by independence of literal choices. ■

Proposition 3. Let $k \geq 1$, $p \geq 1$ integers and let $r > 0$. Let $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbf{R}^p$. The success probability of level-p QAOA on random k-SAT with n variables and expected clauses-to-variables ratio r (see Definitions 1 and 2) is given by

$$\begin{aligned}
&\mathbf{E}_{\boldsymbol{\sigma}\sim \mathrm{CNF}(n,k,r)} \left[\left\langle \Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\boldsymbol{\beta},\boldsymbol{\gamma}) \middle| \mathbf{1}\left\{H[\boldsymbol{\sigma}] = 0\right\} \middle| \Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\boldsymbol{\beta},\boldsymbol{\gamma}) \right\rangle \right] \\
&= \frac{1}{2^n} e^{-2^{-k} rn \left(1 + 4\sum_{j\in[p]} \sin^2 \frac{\gamma_j}{4}\right)} \sum_{\substack{(n_s)_{s\in\{0,1\}^{2p+1}} \\ \sum_s n_s = n}} \binom{n}{(n_s)_s} \prod_s B_{\boldsymbol{\beta},s}^{n_s} \exp\left(rn \sum_{\substack{J\subset[2p+1] \\ |J|\geq 2}} c_J \left(\frac{1}{2n} \sum_{\substack{s\in\{0,1\}^{2p+1} \\ \forall j,j'\in J, s_j = s_{j'}}} n_s \right)^k \right),
\end{aligned} \tag{A8}$$

where

$$c_J := (-1)^{\mathbf{1}[p\in J]} \prod_{j\in J\,:\, j<p} \left(e^{-\frac{i\gamma_j}{2}} - 1 \right) \prod_{j\in J\,:\, j>p} \left(e^{\frac{i\gamma_{2p-j}}{2}} - 1 \right), \tag{A9}$$

and $B_{\boldsymbol{\beta},s}$ is defined in Proposition 2.

Proof. In order to apply Proposition 2, we start by computing

$$\mathbf{E}_{\boldsymbol{\sigma}\sim\mathrm{CNF}(n,k,r)}\left[\bigotimes_{j\in[p]} e^{\frac{i\gamma_j}{2}H[\boldsymbol{\sigma}]}\otimes\{H[\boldsymbol{\sigma}]=0\}\otimes\bigotimes_{j\in[p]} e^{-\frac{i\gamma_{p-1-j}}{2}H[\boldsymbol{\sigma}]}\right]\bigotimes_{j\in[2p+1]}|y^{(2p-j)}\rangle$$
$$=\sum_{m\geq 0}\frac{e^{-rn}(rn)^m}{m!}\mathbf{E}_{\boldsymbol{\sigma}=(\sigma_0,\ldots,\sigma_{m-1})}\left[\bigotimes_{j\in[p]} e^{\frac{i\gamma_j}{2}H[\boldsymbol{\sigma}]}\otimes\{H[\boldsymbol{\sigma}]=0\}\otimes\bigotimes_{j\in[p]} e^{-\frac{i\gamma_{p-1-j}}{2}H[\boldsymbol{\sigma}]}\right]\bigotimes_{j\in[2p+1]}|y^{(2p-j)}\rangle\,.$$

We analyze the terms in the sum for fixed m.

$$\bigotimes_{j\in[p]} e^{\frac{i\gamma_j}{2}H[\boldsymbol{\sigma}]}\otimes\{H[\boldsymbol{\sigma}]=0\}\otimes\bigotimes_{j\in[p]} e^{-\frac{i\gamma_{p-1-j}}{2}H[\boldsymbol{\sigma}]}\bigotimes_{j\in[2p+1]}|y^{(2p-j)}\rangle$$
$$=\left(\bigotimes_{j\in[p]} e^{\frac{i\gamma_j}{2}\sum_{z^{(2p-j)}\in\{0,1\}^n}\sum_{l\in[m]}\mathbf{1}\left[z^{(2p-j)}\nvdash\sigma_l\right]|z^{(2p-j)}\rangle\langle z^{(2p-j)}|}\otimes\sum_{z^{(p)}\in\{0,1\}^n}\prod_{l\in[m]}\mathbf{1}\left[z^{(p)}\vdash\sigma_l\right]|z^{(p)}\rangle\langle z^{(p)}|\otimes\right.$$
$$\left.\bigotimes_{j\in[p]} e^{-\frac{i\gamma_{p-1-j}}{2}\sum_{z^{(p-1-j)}\in\{0,1\}^n}\sum_{l\in[m]}\mathbf{1}\left[z^{(p-1-j)}\nvdash\sigma_l\right]|z^{(p-1-j)}\rangle\langle z^{(p-1-j)}|}\right)\bigotimes_{j\in[2p+1]}|y^{(2p-j)}\rangle$$
$$=\prod_{j\in[p]} e^{\frac{i\gamma_j}{2}\sum_{l\in[m]}\mathbf{1}\left[y^{(2p-j)}\nvdash\sigma_l\right]}\prod_{l\in[m]}\mathbf{1}\left[y^{(p)}\vdash\sigma_l\right]\prod_{j\in[p]} e^{-\frac{i\gamma_{p-1-j}}{2}\sum_{l\in[m]}\mathbf{1}\left[y^{(p-1-j)}\nvdash\sigma_l\right]}\bigotimes_{j\in[2p+1]}|y^{(2p-j)}\rangle$$
$$=\prod_{l\in[m]}\left\{\mathbf{1}\left[y^{(p)}\nvdash\sigma_l\right]\exp\left(\sum_{j\in[p]}\frac{i\gamma_j}{2}\mathbf{1}\left[y^{(2p-j)}\nvdash\sigma_l\right]-\sum_{j\in[p]}\frac{i\gamma_{p-1-j}}{2}\mathbf{1}\left[y^{(p-1-j)}\nvdash\sigma_l\right]\right)\right\}\bigotimes_{j\in[2p+1]}|y^{(2p-j)}\rangle\,.$$

By the factorization in $l\in[m]$ just obtained and independence of clause choices, it suffices to average independently over each random clause $\sigma_0,\ldots,\sigma_{m-1}$. We then compute

$$\mathbf{E}_{\sigma_0}\left\{\mathbf{1}\left[y^{(p)}\vdash\sigma_0\right]\exp\left(\sum_{j\in[p]}\frac{i\gamma_j}{2}\mathbf{1}\left[y^{(2p-j)}\nvdash\sigma_0\right]-\sum_{j\in[p]}\frac{i\gamma_{p-1-j}}{2}\mathbf{1}\left[y^{(p-1-j)}\nvdash\sigma_0\right]\right)\right\}$$
$$=\mathbf{E}_{\sigma_0}\left\{\mathbf{1}\left[y^{(p)}\vdash\sigma_0\right]\prod_{j\in[p]}\exp\left(\frac{i\gamma_j}{2}\mathbf{1}\left[y^{(2p-j)}\nvdash\sigma_0\right]\right)\prod_{j\in[p]}\exp\left(-\frac{i\gamma_{p-1-j}}{2}\mathbf{1}\left[y^{(p-1-j)}\nvdash\sigma_0\right]\right)\right\}$$
$$=\mathbf{E}_{\sigma_0}\left\{\prod_{j\in[p]}\left(1+\left(e^{i\gamma_j/2}-1\right)\mathbf{1}\left[y^{(2p-j)}\nvdash\sigma_0\right]\right)\prod_{j\in[p]}\left(1+\left(e^{-i\gamma_{p-1-j}/2}-1\right)\mathbf{1}\left[y^{(p-1-j)}\nvdash\sigma_0\right]\right)\right.$$
$$\left.\times\left(1-\mathbf{1}\left[y^{(p)}\nvdash\sigma_0\right]\right)\right\}$$
$$=\mathbf{E}_{\sigma_0}\left\{\sum_{J\subset[2p+1]}c_J\prod_{j\in J}\mathbf{1}\left[y^{(j)}\nvdash\sigma_0\right]\right\},$$

where J records the j where we chose the term with $\mathbf{1}\left[y^{(j)}\nvdash\sigma_0\right]$ when expanding the parenthesis, and

$$c_J:=(-1)^{\mathbf{1}[p\in J]}\prod_{j\in J:j<p}\left(e^{-\frac{i\gamma_j}{2}}-1\right)\prod_{j\in J:j>p}\left(e^{\frac{i\gamma_{2p-j}}{2}}-1\right).$$

Using Lemma 1 to average over σ_0 then gives

$$\mathbf{E}_{\sigma_0}\left\{\sum_{J\subset[2p+1]} c_J \prod_{j\in J} \mathbf{1}\left[y^{(j)} \nvdash \sigma_0\right]\right\} = 1 + \sum_{\varnothing\subsetneq J\subset[2p+1]} c_J \left(\frac{\left|\bigcap_{j\in J} y^{(j)}\right|}{2n}\right)^k.$$

Therefore, averaging the original m-clause expression over $\sigma_0,\ldots,\sigma_{m-1}$ yields

$$\begin{aligned}&\mathbf{E}_{\boldsymbol{\sigma}=(\sigma_0,\ldots,\sigma_{m-1})}\left\{\prod_{l\in[m]} \mathbf{1}\left[y^{(p)} \vdash \sigma_l\right] \exp\left(\sum_{j\in[p]} \frac{i\gamma_j}{2}\mathbf{1}\left[y^{(2p-j)} \nvdash \sigma_l\right] - \sum_{j\in[p]} \frac{i\gamma_{p-1-j}}{2}\mathbf{1}\left[y^{(p-1-j)} \nvdash \sigma_l\right]\right)\right\}\\ &= \left(1 + \sum_{\varnothing\subsetneq J\subset[2p+1]} c_J \left(\frac{\left|\bigcap_{j\in J} y^{(j)}\right|}{2n}\right)^k\right)^m,\end{aligned}$$

which, after averaging over $m \sim \mathrm{Poisson}(rn)$, becomes

$$\begin{aligned}&\sum_{m\geq 0} \frac{e^{-rn}(rn)^m}{m!}\mathbf{E}_{\boldsymbol{\sigma}=(\sigma_0,\ldots,\sigma_{m-1})}\left\{\prod_{l\in[m]} \mathbf{1}\left[y^{(p)} \vdash \sigma_l\right] \exp\left(\sum_{j\in[p]} \frac{i\gamma_j}{2}\mathbf{1}\left[y^{(2p-j)} \nvdash \sigma_l\right] - \sum_{j\in[p]} \frac{i\gamma_{p-1-j}}{2}\mathbf{1}\left[y^{(p-1-j)} \nvdash \sigma_l\right]\right)\right\}\\ &= \exp\left(rn \sum_{\varnothing\subsetneq J\subset[2p+1]} c_J \left(\frac{\left|\bigcap_{j\in J} y^{(j)}\right|}{2n}\right)^k\right).\end{aligned}$$

Defining $(n_s)_{s\in\{0,1\}}$ as in Eq. (A2) for bitstrings $y^{(0)},\ldots,y^{(2p)}$, the above can be rewritten as

$$\exp\left(rn \sum_{\varnothing\subsetneq J\subset[2p+1]} c_J \left(\frac{1}{2n} \sum_{\substack{s\in\{0,1\}^{2p+1}\\ \forall j,j'\in J, s_j = s_{j'}}} n_s\right)^k\right).$$

This shows that random k-SAT satisfies the permutation invariance assumption from Proposition 2. One can slightly simplify the expression above by distinguishing the singleton $J = \{j\}, j \in [2p+1]$ from other J. Indeed, for these J,

$$\begin{aligned}\sum_{\substack{s\in\{0,1\}^{2p+1}\\ \forall j',j''\in J, s_{j'} = s_{j''}}} n_s &= \sum_{s\in\{0,1\}^{2p+1}} n_s\\ &= n,\end{aligned}$$

giving the total contribution in the exponential

$$
\begin{aligned}
&rn \sum_{j\in[2p+1]} c_{\{j\}} \left(\frac{n}{2n}\right)^k \\
&= rn \sum_{j\in[2p+1]} 2^{-k} \begin{cases} e^{-\frac{i\gamma_j}{2}} - 1 & \text{if } j < p \\ -1 & \text{if } j = p \\ e^{\frac{i\gamma_{2p-j}}{2}} - 1 & \text{if } j > p \end{cases} \\
&= 2^{-k} rn \left(\sum_{j\in[p]} \left(2\cos\frac{\gamma_j}{2} - 2\right) - 1 \right) \\
&= -2^{-k} rn \left(4 \sum_{j\in[p]} \sin^2\frac{\gamma_j}{4} + 1 \right).
\end{aligned}
$$

■

Proposition 4. For single-layer ($p = 1$) QAOA, Eq. (A8) for the expected success probability of random k-SAT QAOA specializes as follows:

$$
\begin{aligned}
&\mathbf{E}_{\boldsymbol{\sigma}\sim \mathrm{CNF}(n,k,r)} \left[\left\langle \Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\beta,\gamma) | \mathbf{1}\left\{H[\boldsymbol{\sigma}] = 0\right\} | \Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\beta,\gamma) \right\rangle \right] \\
&= e^{-2^{-k} rn \left(1 + 4\sin^2\frac{\gamma}{4}\right)} \sum_{\substack{n_a,n_b,n_c,n_d \\ n_a+n_b+n_c+n_d=n}} \binom{n}{n_a, n_b, n_c, n_d} \left(\cos^2\frac{\beta}{2}\right)^{n_a} \left(\sin^2\frac{\beta}{2}\right)^{n_b} \left(\frac{i\sin\beta}{2}\right)^{n_c} \left(-\frac{i\sin\beta}{2}\right)^{n_d} \\
&\quad \times \exp\left\{ rn \left[4\sin^2\frac{\gamma}{4} \left(\left(\frac{n_b + n_a}{2n}\right)^k - \left(\frac{n_a}{2n}\right)^k \right) + \left(1 - e^{-i\gamma/2}\right) \left(\frac{n_c + n_a}{2n}\right)^k \right.\right. \\
&\qquad \left.\left. + \left(1 - e^{i\gamma/2}\right) \left(\frac{n_d + n_a}{2n}\right)^k \right] \right\}.
\end{aligned} \tag{A10}
$$

Proof. In the $p = 1$ case, it is easy to enumerate the terms in the J sum of the exponential in Eq. (A8). There are four such terms given by $J = \{0,1\}, J = \{0,2\}, J = \{1,2\}$ and $J = \{0,1,2\}$. We therefore explicit

$$
\begin{aligned}
c_{\{0,1\}} &= 1 - e^{-i\gamma/2}, \\
c_{\{0,2\}} &= 4\sin^2\frac{\gamma}{4}, \\
c_{\{1,2\}} &= 1 - e^{i\gamma/2}, \\
c_{\{0,1,2\}} &= -4\sin^2\frac{\gamma}{4}.
\end{aligned}
$$

Also,

$$
\begin{aligned}
B_{\beta,000} &= B_{\beta,111} = \cos^2\frac{\beta}{2}, \\
B_{\beta,001} &= B_{\beta,110} = -\frac{i}{2}\sin\beta, \\
B_{\beta,010} &= B_{\beta,101} = \sin^2\frac{\beta}{2}, \\
B_{\beta,011} &= B_{\beta,100} = \frac{i}{2}\sin\beta.
\end{aligned}
$$

Plugging it into the formula from Proposition 3,

$$\begin{aligned}
&\mathbf{E}_{\boldsymbol{\sigma}\sim\mathrm{CNF}(n,k,r)}\left[\left\langle\Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\beta,\gamma)\right|\mathbf{1}\left\{H[\boldsymbol{\sigma}]=0\right\}\left|\Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\beta,\gamma)\right\rangle\right]\\
&=\frac{e^{-2^{-k}rn\left(1+4\sin^2\frac{\gamma}{4}\right)}}{2^n}\sum_{\substack{(n_s)_{s\in\{0,1\}^3}\\ \sum_s n_s=n}}\binom{n}{(n_s)_s}\left(\cos^2\frac{\beta}{2}\right)^{n_{000}+n_{111}}\left(\sin^2\frac{\beta}{2}\right)^{n_{010}+n_{101}}\left(\frac{i\sin\beta}{2}\right)^{n_{011}+n_{100}}\\
&\quad\times\left(-\frac{i\sin\beta}{2}\right)^{n_{001}+n_{110}}\exp\left\{rn\left[4\sin^2\frac{\gamma}{4}\left(\left(\frac{n_{000}+n_{010}+n_{101}+n_{111}}{2n}\right)^k-\left(\frac{n_{000}+n_{111}}{2n}\right)^k\right)\right.\right.\\
&\quad\left.\left.+\left(1-e^{-i\gamma/2}\right)\left(\frac{n_{000}+n_{100}+n_{011}+n_{111}}{2n}\right)^k+\left(1-e^{i\gamma/2}\right)\left(\frac{n_{000}+n_{001}+n_{110}+n_{111}}{2n}\right)^k\right]\right\}.
\end{aligned}$$

We now observe that for all $s\in\{0,1\}^3$, variable n_s always appears in the form $n_s+n_{\bar{s}}$, where $\bar{s}$ denotes s with all bits flipped. (For instance, n_{000} always appears as part of $n_{000}+n_{111}$. Therefore, we may apply the standard multinomial theorem to each pair of variables $(n_s,n_{\bar{s}})$, reducing the summed variables to four instead of eight. Renaming these variables as follows:

$$\begin{aligned}
n_{000}+n_{111}&\longrightarrow n_a,\\
n_{010}+n_{101}&\longrightarrow n_b,\\
n_{100}+n_{011}&\longrightarrow n_c,\\
n_{001}+n_{110}&\longrightarrow n_d,
\end{aligned}$$

the expression above becomes

$$\begin{aligned}
&\mathbf{E}_{\boldsymbol{\sigma}\sim\mathrm{CNF}(n,k,r)}\left[\left\langle\Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\beta,\gamma)\right|\mathbf{1}\left\{H[\boldsymbol{\sigma}]=0\right\}\left|\Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\beta,\gamma)\right\rangle\right]\\
&=e^{-2^{-k}rn\left(1+4\sin^2\frac{\gamma}{4}\right)}\sum_{\substack{n_a,n_b,n_c,n_d\\ n_a+n_b+n_c+n_d=n}}\binom{n}{n_a,n_b,n_c,n_d}\left(\cos^2\frac{\beta}{2}\right)^{n_a}\left(\sin^2\frac{\beta}{2}\right)^{n_b}\left(\frac{i\sin\beta}{2}\right)^{n_c}\left(-\frac{i\sin\beta}{2}\right)^{n_d}\\
&\quad\times\exp\left\{rn\left[4\sin^2\frac{\gamma}{4}\left(\left(\frac{n_b+n_a}{2n}\right)^k-\left(\frac{n_a}{2n}\right)^k\right)+\left(1-e^{-i\gamma/2}\right)\left(\frac{n_c+n_a}{2n}\right)^k\right.\right.\\
&\qquad\left.\left.+\left(1-e^{i\gamma/2}\right)\left(\frac{n_d+n_a}{2n}\right)^k\right]\right\}.
\end{aligned}$$

■

2. Generalized multinomial sums: A warm-up example

In this section, we estimate the leading exponential contribution of the success probability of QAOA applied to the Hamming weight squared Hamiltonian:

$$H_C:=\left(\sum_{j\in[n]}\frac{1-Z_j}{2}\right)^2. \tag{A11}$$

The success probability is defined as the probability of sampling the all-zero string from the QAOA state:

$$p_{\mathrm{succ}}(n):=\left|\langle 0|e^{-\frac{i\beta H_B}{2}}e^{-\frac{i\gamma H_C}{2}}|+\rangle\right|^2. \tag{A12}$$

Applying all operators in the computational basis, one can easily derive:

$$\begin{aligned}
&\langle 0|e^{-\frac{i\beta H_B}{2}}e^{-\frac{i\gamma H_C}{2}}|+\rangle \\
&= \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}\left(\cos\frac{\beta}{2}\right)^{n-\sum_{j\in[n]}x_j}\left(-i\sin\frac{\beta}{2}\right)^{\sum_{j\in[n]}x_j} \\
&\qquad\times e^{-\frac{i\gamma}{2}\left(\sum_{j\in[n]}x_j\right)^2}.
\end{aligned} \tag{A13}$$

Note that the general term of the sum over bitstrings $x \in \{0,1\}^n$ depends only on the Hamming weight of x and the above can therefore be rewritten

$$\begin{aligned}
&\langle 0|e^{-\frac{i\beta H_B}{2}}e^{-\frac{i\gamma H_C}{2}}|+\rangle \\
&= \frac{1}{\sqrt{2^n}}\sum_{0\le k\le n}\binom{n}{k}\left(\cos\frac{\beta}{2}\right)^{n-k}\left(-i\sin\frac{\beta}{2}\right)^{k}e^{-\frac{i\gamma k^2}{2}}
\end{aligned} \tag{A14}$$

and evaluated in time $\mathcal{O}(n)$. Unfortunately, the infinite-size limit is not immediate due to the exponential of square $e^{-((i\gamma k^2)/2)}$ preventing from applying the usual binomial theorem. However, this difficulty can be remedied at the expense of introducing an additional integral. Indeed, using

$$\int_{\mathbf{R}} \mathrm{d}\theta\, \frac{e^{-\theta^2/4}}{\sqrt{4\pi}}e^{\theta x} = e^{x^2}, \tag{A15}$$

$$\begin{aligned}
&\langle 0|e^{-\frac{i\beta H_B}{2}}e^{-\frac{i\gamma H_C}{2}}|+\rangle \\
&= \frac{1}{\sqrt{2^n}}\sum_{0\le k\le n}\binom{n}{k}\left(\cos\frac{\beta}{2}\right)^{n-k}\left(-i\sin\frac{\beta}{2}\right)^{k} \\
&\qquad\times\int_{\mathbf{R}}\mathrm{d}\theta\,\frac{e^{-\theta^2/4}}{\sqrt{4\pi}}e^{\theta\sqrt{-\frac{i\gamma}{2}}k} \\
&= \frac{1}{\sqrt{2^n}}\int_{\mathbf{R}}\mathrm{d}\theta\,\frac{e^{-\theta^2/4}}{\sqrt{4\pi}}\frac{1}{\sqrt{2^n}} \\
&\quad\times\sum_{0\le k\le n}\binom{n}{k}\left(\cos\frac{\beta}{2}\right)^{n-k}\left(-i\sin\frac{\beta}{2}\right)^{k}e^{\theta\sqrt{-\frac{i\gamma}{2}}k} \\
&= \frac{1}{\sqrt{2^n}}\int_{\mathbf{R}}\mathrm{d}\theta\,\frac{e^{-\theta^2/4}}{\sqrt{4\pi}}\left(\cos\frac{\beta}{2}-i\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\gamma}{2}}}\right)^n \\
&= \frac{1}{\sqrt{2^n}}\int_{\mathbf{R}}\mathrm{d}\theta\,\frac{e^{-\theta^2/4}}{\sqrt{4\pi}}\left(\cos\frac{\beta}{2}-i\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2n}}}\right)^n \\
&\qquad\left(\gamma\longleftarrow\frac{\widetilde{\gamma}}{n}\right) \\
&= \sqrt{\frac{n}{2^n}}\int_{\mathbf{R}}\mathrm{d}\widetilde{\theta}\,\frac{e^{-n\widetilde{\theta}^2/4}}{\sqrt{4\pi}}\left(\cos\frac{\beta}{2}-i\sin\frac{\beta}{2}e^{\widetilde{\theta}\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right)^n \\
&\qquad\left(\widetilde{\theta}\longleftarrow\theta\sqrt{n}\right) \\
&= \sqrt{n}\frac{1}{\sqrt{2^n}}e^{-\frac{i\beta n}{2}}\int_{\mathbf{R}}\mathrm{d}\widetilde{\theta}\,\frac{e^{-n\widetilde{\theta}^2/4}}{\sqrt{4\pi}} \\
&\qquad\times\left(e^{\frac{i\beta}{2}}\left(\cos\frac{\beta}{2}-i\sin\frac{\beta}{2}e^{\widetilde{\theta}\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right)\right)^n \\
&= \sqrt{\frac{n}{4\pi}}\int_{\mathbf{R}}\mathrm{d}\widetilde{\theta}\,\exp\left\{n\left[-\frac{\log 2}{2}-\frac{i\beta}{2}-\frac{\widetilde{\theta}^2}{4}\right.\right. \\
&\qquad\left.\left.+\log\left(e^{\frac{i\beta}{2}}\left(\cos\frac{\beta}{2}-i\sin\frac{\beta}{2}e^{\widetilde{\theta}\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right)\right)\right]\right\} \\
&=: \sqrt{\frac{n}{4\pi}}\int_{\mathbf{R}}\mathrm{d}\widetilde{\theta}\,\exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\widetilde{\theta})\right),
\end{aligned} \tag{A16}$$

where in the fourth line we prescribe the scaling $(\widetilde{\gamma}/n)$, $\widetilde{\gamma}$ constant, for γ, allowing for a change of integration variable $\theta \longrightarrow \widetilde{\theta}$ such that the argument of the integrated exponential in the last line is the product of n and an n-independent function of $\widetilde{\theta}$; this is precisely the setting in which the saddle-point method applies. The sixth line introduces uses the dummy identity $1 = e^{-((i\beta n)/2)}e^{((i\beta n)/2)}$, which allows the argument of the logarithm to be 1 when $\widetilde{\theta} = 0$ or $\widetilde{\gamma} = 0$. In the latter case, the logarithm vanishes for all θ and the integral is the last line is simply Gaussian with value $((e^{-((i\beta n)/2)})/(\sqrt{2^n}))$, as could have been more directly found using $H_B|+\rangle = n|+\rangle$: $\langle 0|e^{-((i\beta H_B)/n)}e^{-((i\gamma H_C)/2)}|+\rangle = \langle 0|e^{-((i\beta H_B)/2)}|+\rangle = e^{-((i\beta n)/2)}\langle 0|+\rangle = ((e^{-((i\beta n)/2)})/(\sqrt{2^n}))$. The logarithm is understood as the principal determination of the complex logarithm, i.e., $\log(\rho e^{i\varphi}) := \log\rho + i\varphi$ for $\rho > 0, \varphi \in [-\pi,\pi)$; it has a discontinuity across the negative real axis.

Remark 1. It is easily verified numerically that for large enough real θ, the argument of the logarithm crosses the negative real axis; however, the exponential of the logarithm, hence $\exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\theta)\right)$ is still analytic in θ on the whole complex plane. Concretely, only the analyticity of the logarithm around $\theta = 0$ will be relevant to apply the saddle-point method, while the integrand will require only a crude bound (not relying on the analyticity of the log) for large θ.

We now estimate the integral in the limit $n \to \infty$ using the saddle-point method. The results we show apply only to small enough (but still constant) $\widetilde{\gamma}$; we will not dedicate much effort to accurately estimating this upper bound as numerical experiments suggest that our estimate for $\langle 0|e^{e^{-((i\beta H_B)/2)}}e^{-((i\gamma H_C)/2)}|+\rangle$ holds beyond the assumptions

required for the proofs anyway [33]. In the following, we always assume $\widetilde{\gamma}$ is positive without loss of generality since $(\beta,\gamma) \longrightarrow (-\beta,-\gamma)$ is equivalent to conjugating the amplitude. We first show that for small enough $\widetilde{\gamma}$, $\Phi_{\beta,\widetilde{\gamma}}$ has a single critical point and provide an accurate estimate for it. To achieve that, it will be convenient to look at critical points of $\Phi_{\beta,\widetilde{\gamma}}$ as the fixed points of a certain function. Namely, by simple differentiation,

$$\Phi'_{\beta,\widetilde{\gamma}}(\theta) = 0 \iff \theta = -\frac{i\sqrt{-2i\widetilde{\gamma}}\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}}{\cos\frac{\beta}{2} - i\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}}. \quad \text{(A17)}$$

To justify the existence and uniqueness of the critical point as well as estimate it, we need only to show the function on the right-hand side is contractive with a sufficiently small constant.

Lemma 2. Let $F(\theta) := -\frac{i\sqrt{-2i\widetilde{\gamma}}\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}}{\cos\frac{\beta}{2} - i\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}}$. There exists a universal constant $c > 0$ such that for all $\widetilde{\gamma} < c$ and all $\theta,\theta' \in \mathbf{C}$ with $|\theta|, |\theta'| \le 2\sqrt{\widetilde{\gamma}}$, the following holds:

(1) $|F(\theta)|, |F(\theta')| \le 2\sqrt{\widetilde{\gamma}}$.
(2) $\left|F(\theta) - F(\theta')\right| \le \gamma|\theta - \theta'|$.

Proof. Let us prove the first statement:

$$\begin{aligned}
|F(\theta)| &= \left|-\frac{i\sqrt{-2i\widetilde{\gamma}}\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}}{\cos\frac{\beta}{2} - i\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}}\right| \\
&\le \sqrt{2\widetilde{\gamma}}\frac{\left|e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right|}{\left|\cos\frac{\beta}{2} - i\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right|} \\
&= \sqrt{2\widetilde{\gamma}}\frac{\left|e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right|}{\left|e^{-\frac{i\beta}{2}} - i\sin\frac{\beta}{2}\left(e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}} - 1\right)\right|} \\
&\le \sqrt{2\widetilde{\gamma}}\frac{e^{\sqrt{2\widetilde{\gamma}}}}{1 - \left(e^{\sqrt{2\widetilde{\gamma}}} - 1\right)} \qquad (|\theta| \le 2\sqrt{\widetilde{\gamma}}) \\
&\le 2\sqrt{\widetilde{\gamma}} \qquad \text{(for small enough } \gamma).
\end{aligned}$$

As for the second statement, we consider the derivative:

$$F'(\eta) = -\frac{e^{\eta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\sin\beta\widetilde{\gamma}}{2\left(\cos\frac{\beta}{2} - i\sin\frac{\beta}{2}e^{\eta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right)^2}. \quad \text{(A18)}$$

Similarly to the previous calculation, this is $\le \widetilde{\gamma}$ for small enough $\widetilde{\gamma}$ and $|\eta| \le 2\sqrt{\widetilde{\gamma}}$. This completes the proof. ■

The existence and uniqueness of the fixed point of F (corresponding to the critical point of $\Phi_{\beta,\widetilde{\gamma}}$) then follows from the Banach fixed-point theorem [34]:

Theorem 1 (Banach fixed-point theorem). Let (X,d) a nonempty complete metric space. Let $T : X \longrightarrow X$ a contraction mapping; that is, there exists $k \in [0,1)$ such that for all $x,x' \in X$, $d(T(x),T(x')) \le kd(x,x')$. Then T has a unique fixed point x^* in X. Besides, x^* can be obtained by iteratively applying T to an arbitrary element of X: for $x_0 \in X$, $x_n := T(x_{n-1})$ $(n \ge 1)$, $x_n \longrightarrow x^*$.

Proposition 5. There exists a universal constant $c > 0$ such that for $\widetilde{\gamma} < c$, $\Phi_{\beta,\widetilde{\gamma}}$ has a unique critical point θ^* satisfying $\left|\theta^* - e^{(i\beta/2)-((3\pi i)/4)}\sin(\beta/2)\sqrt{2\widetilde{\gamma}}\right| \le \mathcal{O}\left(\widetilde{\gamma}^{3/2}\right)$.

Proof. By an earlier observation, θ is a critical point of $\Phi_{\beta,\widetilde{\gamma}}$ if and only if it is a fixed point of F defined in Lemma 2. Applying the Banach fixed-point theorem to F, for small enough $\widetilde{\gamma}$, F has a unique fixed point θ^*, which can be obtained by iteratively applying F to 0. The first iterate is $F(0) = ((-i\sqrt{-2i\widetilde{\gamma}}\sin(\beta/2))/(\cos\frac{\beta}{2} - i\sin\frac{\beta}{2})) = e^{(i\beta/2)-((3\pi i)/4)}\sin(\beta/2)\sqrt{2\widetilde{\gamma}}$. We now bound the distance between this and θ^*:

$$\begin{aligned}
\left|\theta^* - F(0)\right| &= \left|F(\theta^*) - F(0)\right| \\
&\le \left|F(\theta^*) - F(F(0))\right| + |F(F(0)) - F(0)| \\
&\le \widetilde{\gamma}\left|\theta^* - F(0)\right| + |F(F(0)) - F(0)| \\
&\qquad \text{(Lemma 2)}
\end{aligned}$$

so that

$$\begin{aligned}
\left|\theta^* - F(0)\right| &\le \frac{|F(F(0)) - F(0)|}{1 - \widetilde{\gamma}} \\
&\le \frac{\widetilde{\gamma}\,|F(0) - 0|}{1 - \widetilde{\gamma}} \\
&\le \frac{\sqrt{2}\widetilde{\gamma}^{3/2}}{1 - \widetilde{\gamma}} = \mathcal{O}(\widetilde{\gamma}^{3/2}).
\end{aligned}$$

■

We now provide several additional estimates of $\Phi_{\beta,\widetilde{\gamma}}$ itself and its higher-order derivatives that will make the Gaussian approximation rigorous.

Lemma 3. Let the conditions of Proposition 5 be satisfied and θ^* the unique fixed point of $\Phi_{\beta,\widetilde{\gamma}}$. The following

estimates hold:

$$\forall\theta\in\mathbf{C},\quad \Phi_{\beta,\widetilde{\gamma}}(\theta)$$
$$=-\frac{\log(2)}{2}-\frac{i\beta}{2}-\frac{\theta^2}{4}$$
$$-e^{\frac{i\pi}{4}}\sin\frac{\beta}{2}e^{\frac{i\beta}{2}}\sqrt{\frac{\widetilde{\gamma}}{2}}\theta+\mathcal{O}\left(\widetilde{\gamma}\theta^2\right), \tag{A19}$$
$$\forall\theta\in\mathbf{C},\quad \Phi''_{\beta,\widetilde{\gamma}}(\theta) \tag{A20}$$
$$=-\frac{1}{2}+\mathcal{O}\left(\widetilde{\gamma}\right), \tag{A21}$$
$$\Phi_{\beta,\widetilde{\gamma}}(\theta^*)=-\frac{i\beta}{2}-\frac{\log 2}{2}+\frac{i}{2}\sin^2\frac{\beta}{2}e^{i\beta}\widetilde{\gamma}+\mathcal{O}\left(\widetilde{\gamma}^2\right), \tag{A22}$$
$$\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)=-\frac{1}{2}+\mathcal{O}\left(\widetilde{\gamma}\right), \tag{A23}$$
$$\forall\theta\in\mathbf{C},\quad \left|\exp(n\Phi_{\beta,\widetilde{\gamma}}(\theta))\right|$$
$$\leq\exp\left(n\left(\frac{\log 2}{2}-\frac{\Re\left(\theta^2\right)}{4}+|\theta|\sqrt{\frac{\widetilde{\gamma}}{2}}\right)\right),$$
$$\forall\theta\in\mathbf{C},|\theta|=o\left(\frac{1}{\sqrt{\widetilde{\gamma}}}\right) \tag{A24}$$
$$\Longrightarrow\ \left|\Phi'''_{\beta,\widetilde{\gamma}}(\theta)\right|=\mathcal{O}\left(\widetilde{\gamma}^{3/2}\right). \tag{A25}$$

Proof. The first and second result are systematic Taylor expansions. More precisely, an error $\mathcal{O}\left(\widetilde{\gamma}\theta^2\right)$ is obtained in the first equation since $\Phi_{\beta,\widetilde{\gamma}}(\theta)$ can be written as the sum of $-((\log(2))/2)-(i\beta/2)-((\theta^2)/4)$ and a function of $\sqrt{\widetilde{\gamma}}\theta$; the first-order Taylor expansion of the latter gives the stated error term.

The third and fourth results then follow from plugging the estimate of the critical point from Proposition 5 in the Taylor expansions of $\Phi_{\beta,\widetilde{\gamma}}$, $\Phi''_{\beta,\widetilde{\gamma}}$ just derived.

The fifth result follows from a crude bound:

$$\left|\exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\theta)\right)\right|$$
$$=\left|\frac{1}{\sqrt{2^n}}e^{-n\theta^2/4}\left(\cos\frac{\beta}{2}-i\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right)^n\right|$$
$$\leq\frac{1}{\sqrt{2^n}}e^{-n\Re(\theta^2)/4}\left(1+\left|e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right|\right)^n$$
$$\leq\frac{1}{\sqrt{2^n}}e^{-n\Re(\theta^2)/4}\left(1+e^{|\theta|\sqrt{\frac{\widetilde{\gamma}}{2}}}\right)^n$$
$$\leq\frac{1}{\sqrt{2^n}}e^{-n\Re(\theta^2)/4}\left(2e^{|\theta|\sqrt{\frac{\widetilde{\gamma}}{2}}}\right)^n$$
$$=\exp\left(n\left(\frac{\log 2}{2}-\frac{\Re(\theta^2)}{4}+|\theta|\sqrt{\frac{\widetilde{\gamma}}{2}}\right)\right).$$

For the fourth result,

$$\Phi'''_{\beta,\widetilde{\gamma}}(\theta)=\frac{e^{-\frac{3\pi i}{4}}}{2^{5/2}}$$
$$\times\frac{e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\widetilde{\gamma}^{3/2}\sin\beta\left(\cos\frac{\beta}{2}+i\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right)}{\left(\cos\frac{\beta}{2}-i\sin\frac{\beta}{2}e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\right)^3}$$
$$=\frac{e^{-\frac{3\pi i}{4}}}{2^{5/2}}\frac{e^{\theta\sqrt{-\frac{i\widetilde{\gamma}}{2}}}\widetilde{\gamma}^{3/2}\sin\beta\left(e^{\frac{i\beta}{2}}+o(1)\right)}{\left(e^{-\frac{i\beta}{2}}+o(1)\right)^3}$$
$$\left(\theta=o(\widetilde{\gamma}^{-1/2})\right)$$
$$=\mathcal{O}\left(\widetilde{\gamma}^{3/2}\right).$$

■

Proposition 6. For sufficiently small $\widetilde{\gamma}>0$ and $\gamma:=(\widetilde{\gamma}/n)$, the leading exponential contribution of $\langle 0|e^{-((i\beta H_B)/2)}e^{-((i\gamma H_C)/2)}|+\rangle$ as $n\to\infty$ is given by

$$\lim_{n\to\infty}\frac{\log\left(e^{\frac{i\beta n}{2}}\langle 0|e^{-\frac{i\beta H_B}{2}}e^{-\frac{i\gamma H_C}{2}}|+\rangle\right)}{n}$$
$$=\frac{i\beta}{2}+\Phi_{\beta,\widetilde{\gamma}}(\theta^*), \tag{A26}$$

where θ^* is the unique critical point of $\Phi_{\beta,\widetilde{\gamma}}$ defined in Proposition 5.

Proof. Recall

$$\langle 0|e^{-\frac{i\beta H_B}{2}}e^{-\frac{i\gamma H_C}{2}}|+\rangle$$
$$=\sqrt{\frac{n}{4\pi}}\int_{\mathbf{R}}\mathrm{d}\theta\ \exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\theta)\right)=:\sqrt{\frac{n}{4\pi}}I.$$

We now decompose integral I into three contributions:

$$I=\int_{\theta^*-\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}}^{\theta^*-\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}}\mathrm{d}\theta\ \exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\theta)\right)$$
$$+\int_{\theta^*-\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}}^{\theta^*+\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}}\mathrm{d}\theta\ \exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\theta)\right)$$
$$+\int_{\theta^*+\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}}^{\theta^*+\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}}\mathrm{d}\theta\ \exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\theta)\right)$$
$$=:I_1+I_2+I_3.$$

Here, the integrals are understood as complex contour integrals; I_1 and I_3 are along horizontal half-lines, while I_2 is

along an oblique segment. Deforming $\mathbf{R}$ to this path is licit by analyticity and sufficiently fast decrease (cf. estimates from Lemma 3 for the last point) of the integrand. We will now show that the dominant integral is I_2, with I_1 and I_3 being negligible.

Let us then first consider I_2. Introducing a parametrization of the integration path,

$$I_2 = \frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}} \times \int_{-1}^{1} \mathrm{d}t \, \exp\left(n\Phi_{\beta,\widetilde{\gamma}}\left(\theta^* + t\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}\right)\right).$$

By applying the Taylor expansion formula to second order:

$$f(z+h) = f(z) + hf'(z) + \frac{h^2}{2}f''(z) + h^3\int_0^1 \mathrm{d}u \, \frac{(1-u)^2}{2} f'''(z+uh),$$

we can estimate the integrand as follows:

$$\begin{aligned}
&\Phi_{\beta,\widetilde{\gamma}}\left(\theta^* + t\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''(\theta^*)}}\right) \\
&= \Phi_{\beta,\widetilde{\gamma}}(\theta^*) + t\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}\Phi'_{\beta,\widetilde{\gamma}}(\theta^*) \\
&\quad + \frac{t^2}{2}\frac{\widetilde{\gamma}^{-1/2}}{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}\Phi''_{\beta,\gamma}(\theta^*) \\
&\quad + \frac{t^3}{2}\frac{\widetilde{\gamma}^{-3/4}}{\left(-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)\right)^{3/2}} \\
&\quad \times \int_0^1 \mathrm{d}u \, (1-u)^2 \Phi'''_{\beta,\widetilde{\gamma}}\left(\theta^* + ut\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}\right) \\
&= \Phi_{\beta,\widetilde{\gamma}}(\theta^*) - \frac{t^2\widetilde{\gamma}^{-1/4}}{2} \\
&\quad + \frac{t^3}{2}\frac{\widetilde{\gamma}^{-3/4}}{\left(-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)\right)^{3/2}} \\
&\quad \times \int_0^1 \mathrm{d}u \, (1-u)^2 \Phi'''_{\beta,\widetilde{\gamma}}\left(\theta^* + ut\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}\right).
\end{aligned}$$

Now, according to the estimate on $\Phi'''_{\beta,\widetilde{\gamma}}$ from Lemma 3 and the estimate on θ^* from Proposition 5, the function in the last integral is uniformly bounded by $\mathcal{O}\left(\widetilde{\gamma}^{3/2}\right)$. Therefore,

$$\begin{aligned}
&\int_{-1}^{1} \mathrm{d}t \, \exp\left(n\Phi_{\beta,\widetilde{\gamma}}\left(\theta^* + t\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''(\theta^*)}}\right)\right) \\
&= \int_{-1}^{1} \mathrm{d}t \, \exp\left(n\left(\Phi_{\beta,\widetilde{\gamma}}(\theta^*) - \frac{t^2\widetilde{\gamma}^{-1/4}}{2} + \mathcal{O}\left(\widetilde{\gamma}^{3/4}t^3\right)\right)\right) \\
&= n^{-1/2}\exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\theta^*)\right) \qquad (t \longrightarrow \sqrt{n}t) \\
&\quad \times \int_{-\sqrt{n}}^{\sqrt{n}} \mathrm{d}t \, \exp\left(-\frac{t^2\widetilde{\gamma}^{-1/4}}{2} + \frac{1}{\sqrt{n}}\mathcal{O}\left(\widetilde{\gamma}^{3/4}t^3\right)\right).
\end{aligned}$$

Assuming $\widetilde{\gamma}$ small enough, the integrand is dominated by $\exp\left(-((t^2\widetilde{\gamma}^{-1/4})/4)\right)$; therefore, dominated convergence applies and the integral converges to $\int_{\mathbf{R}}\mathrm{d}t \, \exp\left(-((\widetilde{\gamma}^{-1/4}t^2)/2)\right) = \sqrt{2\pi}\,\widetilde{\gamma}^{1/4}$. In summary,

$$I_2 \underset{n\to\infty}{\sim} \left(\frac{4\pi}{n}\right)^{1/2} \widetilde{\gamma}^{-1/8}\left(1 + \mathcal{O}\left(\widetilde{\gamma}\right)\right)\exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\theta^*)\right).$$

We now bound I_3 (the reasoning for I_1 being similar). For that, we use the bound on $\exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\theta)\right)$ provided by Lemma 3 along the integration path $t \geq 0 \longmapsto \theta^* + ((\widetilde{\gamma}^{-1/4})/(\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)})) + t$:

$$\begin{aligned}
&\left|\exp\left(n\Phi_{\beta,\widetilde{\gamma}}\left(\theta^* + \frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}} + t\right)\right)\right| \\
&\leq \exp\left(n\left(\frac{\log 2}{2} - \frac{1}{4}\Re\left\{\left(\theta^* + \frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}} + t\right)^2\right\}\right.\right. \\
&\quad \left.\left. + \left|\theta^* + \frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}} + t\right|\sqrt{\frac{\widetilde{\gamma}}{2}}\right)\right).
\end{aligned}$$

In fact, one can show that the real part squared dominates from the following simple estimates:

$$\begin{aligned}
&\Re\left\{\left(\theta^* + \frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}} + t\right)^2\right\} \\
&= \Re\left\{\left(\mathcal{O}\left(\sqrt{\widetilde{\gamma}}\right) + \sqrt{2}\widetilde{\gamma}^{-1/4} + \mathcal{O}\left(\widetilde{\gamma}^{3/4}\right) + t\right)^2\right\}
\end{aligned}$$

$$
\begin{aligned}
&= \Re\left\{\left(\sqrt{2}\widetilde{\gamma}^{-1/4}+t\right)^2\left(1+\mathcal{O}\left(\sqrt{\widetilde{\gamma}}\right)\right)\right\}\\
&= \left(\sqrt{2}\widetilde{\gamma}^{-1/4}+t\right)^2\left(1+\mathcal{O}\left(\sqrt{\widetilde{\gamma}}\right)\right)\\
&\left|\theta^*+\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}+t\right|\sqrt{\frac{\widetilde{\gamma}}{2}}\\
&= \left|\mathcal{O}\left(\sqrt{\widetilde{\gamma}}\right)+\sqrt{2}\widetilde{\gamma}^{-1/4}+\mathcal{O}\left(\widetilde{\gamma}^{3/4}\right)+t\right|\sqrt{\frac{\widetilde{\gamma}}{2}}\\
&= \left|\sqrt{2}\widetilde{\gamma}^{-1/4}+t\right|\sqrt{\frac{\widetilde{\gamma}}{2}}\left(1+\mathcal{O}\left(\sqrt{\widetilde{\gamma}}\right)\right).
\end{aligned}
$$

It is important that in the expressions above, the error terms are uniform in $t \geq 0$. Therefore, it holds

$$
\begin{aligned}
&\frac{1}{8}\left|\Re\left\{\left(\theta^*+\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''(\theta^*)}}+t\right)^2\right\}\right|\\
&\geq \left|\theta^*+\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''(\theta^*)}}+t\right|\sqrt{\frac{\widetilde{\gamma}}{2}}
\end{aligned}
$$

as long as

$$
\left|\sqrt{2}\widetilde{\gamma}^{-1/4}+t\right| \geq \sqrt{32\widetilde{\gamma}}\left(1+\mathcal{O}\left(\sqrt{\widetilde{\gamma}}\right)\right),
$$

which is indeed verified for all $t \geq 0$ for sufficiently small $\widetilde{\gamma} > 0$. Using that the quadratic part dominates, one can simplify the bound as follows:

$$
\begin{aligned}
&\left|\exp\left(n\Phi_{\beta,\widetilde{\gamma}}\left(\theta^*+\frac{\widetilde{\gamma}^{-1/4}}{\sqrt{-\Phi''_{\beta,\widetilde{\gamma}}(\theta^*)}}+t\right)\right)\right|\\
&\leq \exp\left(-\frac{n}{8}\left(\sqrt{2}\widetilde{\gamma}^{-1/4}+t\right)^2\right)\\
&\leq \exp\left(-\frac{n\widetilde{\gamma}^{-1/2}}{4}-\frac{n\widetilde{\gamma}^{-1/4}t}{2^{3/2}}\right).
\end{aligned}
$$

It follows that $I_3 = \mathcal{O}\left(\widetilde{\gamma}^{1/4}\right)\exp\left(-\frac{\widetilde{\gamma}^{-1/2}n}{4}\right)$. Comparing it with the leading exponential contribution of I_2: $\exp\left(n\Phi_{\beta,\widetilde{\gamma}}(\theta^*)\right)$, using Lemma 3 to estimate $\Phi_{\beta,\widetilde{\gamma}}(\theta^*)$ there, we conclude that for $\widetilde{\gamma}$ sufficiently small, I_3 is exponentially negligible compared to I_2 as $n \to \infty$. The same holds for I_1. Putting all this together yields the leading exponential contribution of $\langle 0|e^{-((i\beta H_B)/2)}e^{-((i\gamma H_C)/2)}|+\rangle$ stated in the proposition. ■

Combining the last proposition with estimates on $\widetilde{\Phi}$ given in Lemma 3 leads to the following corollary:

Corollary 1. For sufficiently small $\widetilde{\gamma} > 0$ and $\gamma := (\widetilde{\gamma}/n)$, the leading exponential contribution of $\langle 0|e^{-((i\beta H_B)/2)}e^{-((i\gamma H_C)/2)}|+\rangle$ as $n \to \infty$ is given, up to first order in $\widetilde{\gamma}$, by

$$
\begin{aligned}
&\lim_{n\to\infty}\frac{\log\left(e^{\frac{i\beta n}{2}}\langle 0|e^{-\frac{i\beta H_B}{2}}e^{-\frac{i\gamma H_C}{2}}|+\rangle\right)}{n}\\
&= -\frac{\log 2}{2}+\frac{i}{2}\sin^2\frac{\beta}{2}e^{i\beta}\widetilde{\gamma}+\mathcal{O}\left(\widetilde{\gamma}^2\right). \qquad \text{(A27)}
\end{aligned}
$$

Given β, the last result informs on how to choose $\widetilde{\gamma}$ to increase the success probability (compared to the trivial random sampling case $\widetilde{\gamma} = 0$). This is equivalent to increasing the real part of the scaling exponent just derived, which is $-((\log(2))/2) - \frac{1}{2}\widetilde{\gamma}\sin^2(\beta/2)\sin\beta + \mathcal{O}\left(\widetilde{\gamma}^2\right)$. Therefore, to first order in $\widetilde{\gamma} > 0$, one needs $\beta \in [-\pi, 0]$ for the exponent to be greater than $-((\log(2))/2)$, and the optimal choice to maximize the growth rate in $\widetilde{\gamma}$ is $\beta = -(2\pi/3)$, which yields $-((\log(2))/2) + ((3^{3/2})/16)\widetilde{\gamma} + \mathcal{O}\left(\widetilde{\gamma}^2\right)$ for the exponent previously derived.

In fact, from exact evaluation of the amplitude at finite size n [which requires only to compute the n-term sum in Eq. (A14)], we conjecture the optimal parameters rather converge to $\beta = -(\pi/2)$ and $\widetilde{\gamma} = \pi$ in the limit $n \to \infty$. Besides, we conjecture that for this choice of parameters, the success probability decays at most polynomially in n. Remarkably, the analytic formula from Proposition 6 agrees with these observations (although we do not rigorously know it to apply for these parameters). Indeed, for $\beta = -(\pi/2)$ and $\widetilde{\gamma} = \pi$, the critical-point equation is easily checked to admit the simple solution $\theta^* = \sqrt{-(i\pi/2)}$. One can check that for this solution,

$$
\Phi_{\beta,\widetilde{\gamma}}\left(\theta^*\right) = \frac{i\pi}{8}. \qquad \text{(A28)}
$$

Since this has vanishing real part, the saddle-point method, if correct, would predict the amplitude to decay only polynomially in n, in agreement with empirical results at finite size.

3. Generalized multinomial sums

a. Statement of the problem

Definition 3. Let $p \geq 1$ an integer, $q \geq 1$ an integer and $\mathbf{A} = (A_{\alpha s})_{\substack{\alpha\in\mathcal{A}\\ s\in\mathcal{S}}}$, $\mathbf{b} = (b_s)_{s\in\mathcal{S}}$, $\mathbf{c} = (c_\alpha)_{\alpha\in\mathcal{A}}$ families of complex numbers indexed by two arbitrary index sets $\mathcal{S}$ and $\mathcal{A}$. Besides, assume

$$
\sum_{s\in\mathcal{S}} b_s = 1. \qquad \text{(A29)}
$$

We consider the problem of estimating:

$$S_q(\mathbf{A},\mathbf{b},\mathbf{c}\,;\,n) := \sum_{\substack{(n_s)_{s\in\mathcal{S}} \\ \sum_s n_s = n}} \binom{n}{(n_s)_s} \left(\prod_s b_s^{n_s}\right) e^{n\sum_\alpha c_\alpha \left(\frac{1}{n}\sum_s A_{\alpha s} n_s\right)^{2q}} \tag{A30}$$

as $n \to \infty$ while $\mathbf{A}, \mathbf{b}, \mathbf{c}$ are kept constant.

Example 1 (Random 2^q-SAT QAOA). The expression obtained in Proposition 3 for the expected fixed-parameter success probability of QAOA on random k-SAT is of the desired form when $k = 2^q$ up to the trivial factor $e^{-2^{-2q} rn\left(1+4\sum_{j\in[p]}\sin^2(\gamma_j/4)\right)}$. In this particular case, the parameters of the generalized multinomial sum can be taken as follows:

$$\mathcal{S} := \{0,1\}^{2p+1}, \tag{A31}$$

$$\mathcal{A} := \{J \subset [2p+1] \,:\, |J| \geq 2\}, \tag{A32}$$

$$b_s := \frac{B_{\boldsymbol{\beta},s}}{2}, \tag{A33}$$

$$c_\alpha := r(-1)^{\mathbf{1}[p\in\alpha]} \prod_{j\in\alpha, j<p} \left(e^{-i\gamma_j/2} - 1\right) \prod_{j\in\alpha, j>p} \left(e^{i\gamma_{2p-j}/2} - 1\right), \tag{A34}$$

$$A_{\alpha s} := \frac{1}{2}\mathbf{1}\left[\forall j, j' \in \alpha,\, s_j = s_{j'}\right]. \tag{A35}$$

b. The quadratic case

In the case $q = 1$ of Definition 3, the generalized multinomial sum in Eq. (A30) can be given an integral representation by Gaussian integration, similar to the toy-model example considered in Appendix A 2. Explicitly,

$$\begin{aligned}
&S_1(\mathbf{A},\mathbf{b},\mathbf{c}\,;\,n) \\
&= \sum_{\substack{(n_s)_{s\in\mathcal{S}} \\ \sum_s n_s = n}} \binom{n}{(n_s)_s} \left(\prod_s b_s^{n_s}\right) \exp\left(n\sum_\alpha c_\alpha \left(\frac{1}{n}\sum_s A_{\alpha s} n_s\right)^2\right) \\
&= \int_{\mathbf{R}^{|\mathcal{A}|}} \prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha \frac{e^{-y_\alpha^2/4}}{\sqrt{4\pi}} \sum_{\substack{(n_s)_{s\in\mathcal{S}} \\ \sum_s n_s = n}} \binom{n}{(n_s)_s} \left(\prod_s b_s^{n_s}\right) \exp\left(\sqrt{n}\sum_\alpha \sqrt{c_\alpha}\left(\frac{1}{n}\sum_s A_{\alpha s} n_s\right) y_\alpha\right) \\
&= n^{|\mathcal{A}|/2} \int_{\mathbf{R}^{|\mathcal{A}|}} \prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha \frac{e^{-ny_\alpha^2/4}}{\sqrt{4\pi}} \sum_{\substack{(n_s)_{s\in\mathcal{S}} \\ \sum_s n_s = n}} \binom{n}{(n_s)_s} \left(\prod_s b_s^{n_s}\right) \exp\left(\sum_\alpha \sqrt{c_\alpha}\left(\sum_s A_{\alpha s} n_s\right) y_\alpha\right) \\
&\qquad\qquad \text{(change variables } y_\alpha := \sqrt{n} y'_\alpha \quad \forall \alpha \in \mathcal{A}) \\
&= n^{|\mathcal{A}|/2} \int_{\mathbf{R}^{|\mathcal{A}|}} \prod_{\alpha\in\mathcal{A}} \mathrm{d}y'_\alpha \frac{e^{-n(y'_\alpha)^2/4}}{\sqrt{4\pi}} \sum_{\substack{(n_s)_{s\in\mathcal{S}} \\ \sum_s n_s = n}} \binom{n}{(n_s)_s} \prod_s \left(b_s \exp\left(\sum_\alpha \sqrt{c_\alpha} A_{\alpha s} y'_\alpha\right)\right)^{n_s}
\end{aligned}$$

$$
\begin{aligned}
&= n^{|\mathcal{A}|/2} \int_{\mathbf{R}^{|\mathcal{A}|}} \prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha \, \frac{e^{-ny_\alpha^2/4}}{\sqrt{4\pi}} \sum_{\substack{(n_s)_{s\in\mathcal{S}} \\ \sum_s n_s = n}} \binom{n}{(n_s)_s} \prod_s \left(b_s \exp\left(\sum_\alpha \sqrt{c_\alpha} A_{\alpha s} y_\alpha \right) \right)^{n_s} \\
&= \frac{n^{|\mathcal{A}|/2}}{(4\pi)^{|\mathcal{A}|/2}} \int_{\mathbf{R}^{|\mathcal{A}|}} \left(\prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha \right) \exp\left(-\frac{n}{4} \sum_{\alpha\in\mathcal{A}} y_\alpha^2 \right) \left(\sum_s b_s \exp\left(\sum_\alpha \sqrt{c_\alpha} A_{\alpha s} y_\alpha \right) \right)^n \\
&=: \frac{n^{|\mathcal{A}|/2}}{(4\pi)^{|\mathcal{S}|/2}} \int_{\mathbf{R}^{|\mathcal{A}|}} \left(\prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha \right) \exp\left(n\Phi\left((y_\alpha)_{\alpha\in\mathcal{A}} \right) \right),
\end{aligned}
$$

where

$$
\begin{aligned}
&\Phi\left((y_\alpha)_{\alpha\in\mathcal{A}} \right) \\
&:= -\frac{1}{4} \sum_{\alpha\in\mathcal{A}} y_\alpha^2 + \log \sum_{s\in\mathcal{S}} b_s \exp\left(\sum_{\alpha\in\mathcal{A}} \sqrt{c_\alpha} A_{\alpha s} y_\alpha \right).
\end{aligned} \tag{A36}
$$

This has the same formal structure as the Hamming weight squared toy example discussed in Appendix A 2, except the integral is multivariable instead of univariate. Note that the prefactor $((n^{|\mathcal{A}|/2})/((4\pi)^{|\mathcal{A}|/2}))$, similar to $\sqrt{(n/4\pi)}$ in the toy example, is irrelevant to the exponential scaling here since we assume $|\mathcal{A}|$ constant, so that $((n^{|\mathcal{A}|/2})/((4\pi)^{|\mathcal{A}|/2})) = \mathrm{poly}(n)$. The analysis performed there for small $\widetilde{\gamma}$ therefore carries over here *for sufficiently small* c_α; similarly, the analysis of the $q \geq 2$ case detailed in the Supplemental Material [15] will require $\mathbf{c}$ to be bounded by a sufficiently small constant. In the cases considered in Appendix A 1 where the generalized multinomial sum arises from a QAOA expectation, this is equivalent to assuming small (yet constant) $\boldsymbol{\gamma}$ angles but places no restriction on the $\boldsymbol{\beta}$ [35]. We then state the following proposition establishing the scaling of $S_q(\mathbf{A}, \mathbf{b}, \mathbf{c}\,; n)$ in the $q = 1$ case. We leave out the proof, which would essentially be a repetition of that of Proposition 6 from Appendix A 2; besides, the more general case $q \geq 1$ will be investigated in the Supplementary Material [15].

Proposition 7. Let a general multinomial sum $S_1(\mathbf{A}, \mathbf{b}, \mathbf{c}; n)$ be given as in Definition 3 with $q = 1$ and recall the definition of $\Phi(\cdot)$ in Eq. (A36). Then, for sufficiently small $(c_\alpha)_{\alpha\in\mathcal{A}}$,

$$
\lim_{n\to\infty} \frac{\log S_1(\mathbf{A}, \mathbf{b}, \mathbf{c}; n)}{n} = \Phi\left(\mathbf{y}^* \right), \tag{A37}
$$

where $\mathbf{y}^* = \left(y_\alpha^* \right)_{\alpha\in\mathcal{A}}$ is the unique critical point of $\Phi(\cdot)$, whose existence and uniqueness is indeed guaranteed for sufficiently small $(c_\alpha)_{\alpha\in\mathcal{A}}$. Besides, this critical point can be understood as the fixed point of the function:

$$
\begin{array}{ccc}
\mathbf{C}^{|\mathcal{A}|} & \longrightarrow & \mathbf{C}^{|\mathcal{A}|} \\
(y_\alpha)_{\alpha\in\mathcal{A}} & \longmapsto & \left(\frac{2\sqrt{c_\alpha} \sum_{s\in\mathcal{S}} b_s A_{\alpha s} \exp\left(\sum_{\alpha'\in\mathcal{A}} \sqrt{c_{\alpha'}} A_{\alpha' s} y_{\alpha'} \right)}{\sum_{s\in\mathcal{S}} b_s \exp\left(\sum_{\alpha'\in\mathcal{S}} \sqrt{c_{\alpha'}} A_{\alpha' s} y_{\alpha'} \right)} \right)_{\alpha\in\mathcal{A}}.
\end{array} \tag{A38}
$$

In particular, assuming $|c_\alpha| \leq c$ for definiteness, the following estimates hold in the limit $c \to 0$:

$$
y_\alpha^* = 2\sqrt{c_\alpha} \sum_{s\in\mathcal{S}} b_s A_{s\alpha} + \mathcal{O}\left(c^{3/2} \right), \tag{A39}
$$

$$
\begin{aligned}
&\lim_{n\to\infty} \frac{\log S_1(\mathbf{A}, \mathbf{b}, \mathbf{c}; n)}{n} \\
&= \sum_{\alpha\in\mathcal{A}} c_\alpha \left(\sum_{s\in\mathcal{S}} b_s A_{\alpha s} \right)^2 + \mathcal{O}\left(c^{3/2} \right).
\end{aligned} \tag{A40}
$$

Remark 2. Note that the leading-order contribution of the scaling exponent of $S_q(\mathbf{A}, \mathbf{b}, \mathbf{c}; n)$ in Eq. (A40) is what we would have obtained had we (nonrigorously) estimated $S_q(\mathbf{A}, \mathbf{b}, \mathbf{c}\,; n)$ by naively expanding the log in Φ to lowest order

in c for each value of $(y_\alpha)_{\alpha\in\mathcal{A}}$, before integrating the resulting Gaussian over these variables. That is,

$$
\begin{aligned}
&S_1(\mathbf{A},\mathbf{b},\mathbf{c};n)\\
&\overset{\text{poly}(n)}{\approx} \int_{\mathbf{R}^{|\mathcal{A}|}} \left(\prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha\right) \exp\left(n\Phi\left((y_\alpha)_{\alpha\in\mathcal{A}}\right)\right)\\
&= \int_{\mathbf{R}^{|\mathcal{A}|}} \left(\prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha\right) \exp\left(n\left(-\frac{1}{4}\sum_{\alpha\in\mathcal{A}} y_\alpha^2 + \log\sum_{s\in\mathcal{S}} b_s \exp\left(\sum_{\alpha\in\mathcal{A}} \sqrt{c_\alpha} A_{\alpha s} y_\alpha\right)\right)\right)\\
&\approx \int_{\mathbf{R}^{|\mathcal{A}|}} \left(\prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha\right) \exp\left(n\left(-\frac{1}{4}\sum_{\alpha\in\mathcal{A}} y_\alpha^2 + \log\left(\sum_{s\in\mathcal{S}} b_s\left(1+\sum_{\alpha\in\mathcal{A}} \sqrt{c_\alpha} A_{\alpha s} y_\alpha\right)\right)\right)\right) \quad \text{(not rigorous!)}\\
&= \int_{\mathbf{R}^{|\mathcal{A}|}} \left(\prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha\right) \exp\left(n\left(-\frac{1}{4}\sum_{\alpha\in\mathcal{A}} y_\alpha^2 + \log\left(\sum_{s\in\mathcal{S}} b_s + \sum_{s\in\mathcal{S}} b_s \sum_{\alpha\in\mathcal{A}} \sqrt{c_\alpha} A_{\alpha s} y_\alpha\right)\right)\right)\\
&= \int_{\mathbf{R}^{|\mathcal{A}|}} \left(\prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha\right) \exp\left(n\left(-\frac{1}{4}\sum_{\alpha\in\mathcal{A}} y_\alpha^2 + \log\left(1+\sum_{s\in\mathcal{S}} b_s \sum_{\alpha\in\mathcal{A}} \sqrt{c_\alpha} A_{\alpha s} y_\alpha\right)\right)\right)\\
&\approx \int_{\mathbf{R}^{|\mathcal{A}|}} \left(\prod_{\alpha\in\mathcal{A}} \mathrm{d}y_\alpha\right) \exp\left(n\left(-\frac{1}{4}\sum_{\alpha\in\mathcal{A}} y_\alpha^2 + \sum_{s\in\mathcal{S}} b_s \sum_{\alpha\in\mathcal{A}} \sqrt{c_\alpha} A_{\alpha s} y_\alpha\right)\right) \quad \text{(not rigorous!)}\\
&\overset{\text{poly}(n)}{\approx} \exp\left(n\sum_{\alpha\in\mathcal{A}} c_\alpha \left(\sum_{s\in\mathcal{S}} b_s A_{\alpha s}\right)^2\right).
\end{aligned}
$$

c. Algorithmic implementation

Proposition 1, and more generally proposition 7 from the Supplementary Material, establishes the exponential scaling of the generalized multinomial sum Eq. (A30) (for sufficiently small **c** parameters) by expressing it as the fixed point of a certain function. It remains to specify how this fixed point is found in practice. For sufficiently small c, a generalization of the argument for the toy model from Appendix A 2 shows the fixed point can be approximated to error $\mathcal{O}(\varepsilon)$ using $\mathcal{O}(\log(1/\varepsilon))$ iterations. Each iteration applies the function to the previous approximation of the fixed point, starting with lowest-order approximation 0. The procedure is explicited in Algorithm 1.

For large c, we do not rigorously know whether the fixed point introduced in Proposition 7 from the Supplemental Material [15] exists and is unique. Besides, even if this holds, we have not formally proven that this fixed point prescribes the exponential scaling of the generalized multinomial sum. However, these limitations may be a mere artifact of our proof methods; it would therefore be desirable for the fixed-point finding algorithm to extrapolate to larger c. For that purpose, we supplement Algorithm 1 with a heuristic Algorithm 2, which attempts to find the critical point for large c. Informally, the only change is to introduce some damping, controlled by a parameter $\rho \in [0,1]$, as the approximation to the fixed point is updated: instead of updating z^* to $F(z^*)$, we update it to a weighted combination of this proposal and the previous value. This simple amendment was empirically observed to solve the convergence problem and yield a fixed point varying smoothly with **c**.

We now specialize to random k-SAT QAOA an describe a more efficient implementation of the algorithm in this case. Precisely, we provide a more efficient implementation of the evaluation of F and its derivatives $\partial_\alpha F$ by exploiting the specific structure of matrix **A** of the generalized multinomial sum Eq. (A30) for the expectation value arising from that problem. In general, recalling the definition of $F(z)$ in Eq. (16), it is easily seen that naively evaluating $F(z)$ and all its derivatives requires $\mathcal{O}(|\mathcal{A}|.|\mathcal{S}|)$ multiplications. Applying that to random 2^q-SAT using the parameters stated in Example 1, this yields a complexity $\mathcal{O}(16^p)$. We now describe a more efficient naive method reducing this to $\mathcal{O}(4^p)$. For that purpose, it will be more convenient to slightly amend the definition of the generalized multinomial sum parameters in Example 1; namely, we include all subsets of $[2p+1]$ in the index set $\mathcal{A}$ (not only those with two elements or more), based on the following rewriting of Eq. (A8) for the success probability of k-SAT QAOA:

$$\mathbf{E}_{\boldsymbol{\sigma}\sim\mathrm{CNF}(n,k,r)}\left[\left\langle\Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\boldsymbol{\beta},\boldsymbol{\gamma})\right|\mathbf{1}\left\{H[\boldsymbol{\sigma}]=0\right\}\left|\Psi_{\mathrm{QAOA}}(\boldsymbol{\sigma},\boldsymbol{\beta},\boldsymbol{\gamma})\right\rangle\right]$$
$$=\frac{1}{2^n}e^{-2^{-k}rn}\sum_{\substack{(n_s)_{s\in\{0,1\}^{2p+1}}\\ \sum_s n_s=n}}\binom{n}{(n_s)_s}\prod_s B_{\boldsymbol{\beta},s}^{n_s}\exp\left(rn\sum_{J\subset[2p+1]}c_J\left(\frac{1}{2n}\sum_{\substack{s\in\{0,1\}^{2p+1}\\ \forall j,j'\in J,\, s_j=s_{j'}}}n_s\right)^k\right), \tag{A41}$$

where the definition of c_J in Proposition 3 is unchanged (and irrelevant to the argument). With this amended setup, we now show that given a 2^{2p+1}-component vector $\left(v_j\right)_{j\in\{0,1\}^{2p+1}}$, both vector $\left(\sum_{\alpha\subset[2p+1]}A_{\alpha s}v_\alpha\right)_{s\in\{0,1\}^{2p+1}}$ and vector $\left(\sum_{s\in\{0,1\}^{2p+1}}A_{\alpha s}v_s\right)_{\alpha\subset[2p+1]}$ can be compute in time $\mathcal{O}\left(p4^p\right)$ instead of the naive $\mathcal{O}\left(16^p\right)$.

In the above, we naturally identify n-bit bitstrings and subsets of $[n]$, where 1 bits denote the elements included in the set. We now observe that both these sums can be reduced to the sum-over-subsets problem [36] that we recall below:

Definition 4 (Sum-over-subsets). Given a vector $(v_\alpha)_{\alpha\subset[n]}$ of size 2^n, the sum-over-subset problem consists of computing $\sum_{\alpha\subset S}v_\alpha$ for all subsets S of $[n]$. There exists an algorithm, based on dynamic programming, performing this calculation in time $\mathcal{O}\left(n2^n\right)$.

A naive algorithm would be to iterate over subsets S and sum the v_α such that α is included in S, demanding time $\mathcal{O}\left(4^n\right)$. We now connect the sum-over-subsets problem to the calculation of the sums mentioned above. Let us start with the sum over α, which can be expanded as follows:

$$\sum_{\alpha\subset[2p+1]}A_{\alpha s}v_\alpha$$
$$=\frac{1}{2}\sum_{\substack{\alpha\subset[2p+1]\\ \forall j,j'\in\alpha,\, s_j=s_{j'}}}v_\alpha$$
$$=\frac{1}{2}\sum_{\alpha\subset\left\{j:s_j=0\right\}}v_\alpha+\frac{1}{2}\sum_{\alpha\subset\left\{j:s_j=1\right\}}v_\alpha.$$

Interpreting s as describing a subset of $[2p+1]$, the last sum can then be understood as a sum over the subsets of s, and can therefore be evaluated in time $\mathcal{O}\left((2p+1)2^{2p+1}\right)=\mathcal{O}\left(p4^p\right)$ with the DP algorithm for sum-over-subsets. The first sum can be interpreted similarly, still regarding s as a subset of $[2p+1]$, but with set elements now identified to 0 bits. Let us now consider the sums over s. By the discussion on the sums over α, this sum can be considered as sums over *supersets*. These in turn reduce to sums over subsets by passing to

ALGORITHM 1. Compute an approached fixed point z^* for sufficiently small c.

Data: q parameter, matrices/vectors $\boldsymbol{A},\boldsymbol{b},\boldsymbol{c}$, maximum number of iterations N_{iter}, relative fixed-point error threshold ε
$z^*\leftarrow 0$;
$i\leftarrow 0$;
while $i<N_{iter}$ **do**
  $z_\alpha^*\leftarrow 2^q\left(-\partial_\alpha F\left(z^*\right)\right)^{2^q-1}$;
  if $\max_{\alpha\in\mathcal{A}}\left|\frac{z_\alpha^*-\partial_\alpha F\left(z^*\right)}{z_\alpha^*}\right|<\varepsilon$ **then**
    break
  end
  $i\leftarrow i+1$;
end
return (i,z^*);

ALGORITHM 2. Try to compute an approached fixed point z^* with initial suggestion and damping.

Data: q parameter, matrices/vectors $\boldsymbol{A},\boldsymbol{b},\boldsymbol{c}$, maximum number of iterations N_{iter}, relative fixed-point error threshold ε, initial suggestion z_{init}^* for z^*, damping coefficient $\rho\in[0,1)$
$z^*\leftarrow z_{\mathrm{init}}^*$;
$i\leftarrow 0$;
while $i<N_{iter}$ **do**
  $z_\alpha^*\leftarrow\rho z_\alpha^*+(1-\rho)2^q\left(-\partial_\alpha F\left(z^*\right)\right)^{2^q-1}$;
  if $\max_{\alpha\in\mathcal{A}}\left|\frac{z_\alpha^*-\partial_\alpha F\left(z^*\right)}{z_\alpha^*}\right|<\varepsilon$ **then**
    break
  end
  $i\longleftarrow i+1$;
end
return (i,z^*);

ALGORITHM 3. Compute sum $\sum_\alpha A_{\alpha s} v_\alpha$ for random 2^q-SAT QAOA.

Data: Vector $\boldsymbol{v} = (v_\alpha)_{\alpha \subset [n]}$.
$z^{(1)} \leftarrow v$;
$z^{(0)} \leftarrow v$;
if $i \in \{0, \ldots, n-1\}$; /* Interpret 1 bits of s as designing set */ **then**
  for $s \in \{0,1\}^n$ **do**
    if $s_i = 1$ **then**
      $z_s^{(1)} \leftarrow z_s^{(1)} + z^{(1)}_{s \text{ XOR } 2^i}$;
    end
  end
  for $i \in \{0, \ldots, n-1\}$; /* Interpret 0 bits of s as designing set */ **do**
    for $s \in \{0,1\}^n$ **do**
      if $s_i = 0$ **then**
        $z_s^{(0)} \leftarrow z_s^{(0)} + z^{(0)}_{s \text{ XOR } 2^i}$
      end
    end
  end
  return $z^{(0)} + z^{(1)} - v_\varnothing$.
end

ALGORITHM 4. Compute sum $\sum_s A_{\alpha s} v_s$ for random 2^q-SAT QAOA.

Data: Vector $\boldsymbol{v} = (v_s)_{s \in \{0,1\}^n}$
$z^{(0)} \leftarrow \text{REVERSE}(v)$; /* Reversing vector is equivalent to flipping bits of bitstring coordinates */
$z^{(1)} \leftarrow v$ **for** $i \in \{0, \ldots, n-1\}$; /* Indices from α are 0 in s */ **do**
  for $\alpha \subset [n]$ **do**
    if $i \notin \alpha$ **then**
      $z_\alpha^{(0)} \leftarrow z_\alpha^{(0)} + z^{(0)}_{\alpha \sqcup \{i\}}$
    end
  end
  for $i \in \{0, \ldots, n-1\}$; /* Indices from α are 1 in s */ **do**
    for $\alpha \subset [n]$ **do**
      if $i \notin \alpha$ **then**
        $z_\alpha^{(1)} \leftarrow z_\alpha^{(1)} + z^{(1)}_{\alpha \sqcup \{i\}}$
      end
    end
  end
  return $z^{(0)} + z^{(1)} - v_\varnothing$
end

the complementary sets. Explicitly,

$$\begin{aligned}
&\sum_{s \in \{0,1\}^{2p+1}} A_{\alpha s} v_s \\
&= \frac{1}{2} \sum_{\substack{s \in \{0,1\}^{2p+1} \\ \alpha \subset \{j : s_j = 0\}}} v_s + \frac{1}{2} \sum_{s : \alpha \subset \{j : s_j = 1\}} v_s \\
&= \frac{1}{2} \sum_{\substack{s \in \{0,1\}^{2p+1} \\ [2p+1] - \{j : s_j = 0\} \subset [2p+1] - \alpha}} v_s \\
&\quad + \frac{1}{2} \sum_{\substack{s \in \{0,1\}^{2p+1} \\ [2p+1] - \{j : s_j = 1\} \subset [2p+1] - \alpha}} v_s \\
&= \frac{1}{2} \sum_{\substack{s \in \{0,1\}^{2p+1} \\ \{j : s_j = 1\} \subset [2p+1] - \alpha}} v_s + \frac{1}{2} \sum_{\substack{s \in \{0,1\}^{2p+1} \\ \{j : s_j = 0\} \subset [2p+1] - \alpha}} v_s .
\end{aligned}$$

The first sum in the last expression can be computed as a sum over subsets of $[2p+1] - \alpha$. Therefore, to evaluate it for all α, it suffices to apply the sum-over-subsets algorithm (interpreting 1 bits of $s \in \{0,1\}^{2p+1}$ as designing the elements included in the set) and reverse the resulting vector to account for the complement $[2p+1] - \alpha$. The same applies for computing the second sum, except one interprets 0 bits of s as marking elements of the set. The overall complexity is then also $O(p4^p)$. To make these observations concrete, we provide detailed Algorithms 3 and 4 to evaluate the desired sums.

APPENDIX B: ADDITIONAL NUMERICAL RESULTS

In this Appendix we present the full optimization landscape for QAOA at $p = 1$ in Figs. 5 ($k = 2$), 6 ($k = 4$), 7 ($k = 8$), 8 ($k = 16$). More precisely, for each value of k, we report three quantities as functions of β and γ, corresponding to the three columns of the figures.

(1) The expected success probability, with a logarithm taken and rescaled by the number of variables n:

$$\frac{1}{n} \log \mathbf{E}_{\boldsymbol{\sigma}} \left[\langle \Psi_{\text{QAOA}}(\boldsymbol{\sigma}, \beta, \gamma) | \mathbf{1}\{H[\boldsymbol{\sigma}] = 0\} \times |\Psi_{\text{QAOA}}(\boldsymbol{\sigma}, \beta, \gamma)\rangle \right], \tag{B1}$$

which, assuming the existence of an exponential scaling, converges to the scaling exponent as $n \to \infty$.
For this set of experiments, $n = 20$.

(2) The empirical scaling exponent determined from an exponential fit. In the results shown, the fit is performed on 11 points corresponding to problem sizes $20 \le n \le 30$.

(3) The correlation coefficient of the exponential fit.

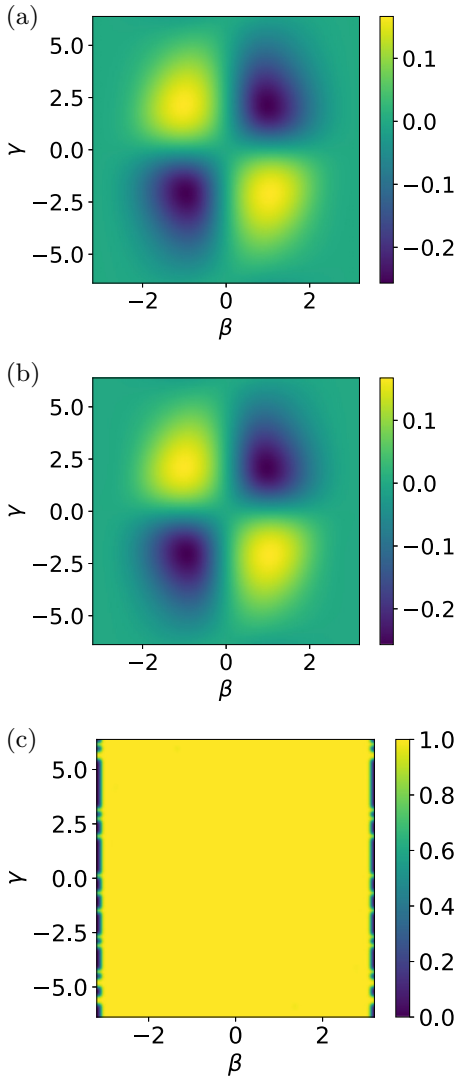


FIG. 5. QAOA optimization landscape at $p = 1$ and for $k = 2$. By periodicity (following from the integrality of the cost function), β and γ can be, respectively, restricted to $[-\pi, \pi]$ and $[-2\pi, 2\pi]$. In case the represented function is negligible except on a small part of this domain, we choose to enlarge the rectangle while keeping centered at 0; for instance, when enlarging by a factor of 2, the represented domain is $[-(\pi/2), (\pi/2)] \times [\pi, \pi]$. The central symmetry is a general feature of QAOA, applying to all cost functions and diagonal unitaries. (a) Logarithmic success probability. (b) Exponential fit. (c) Correlation coefficient.

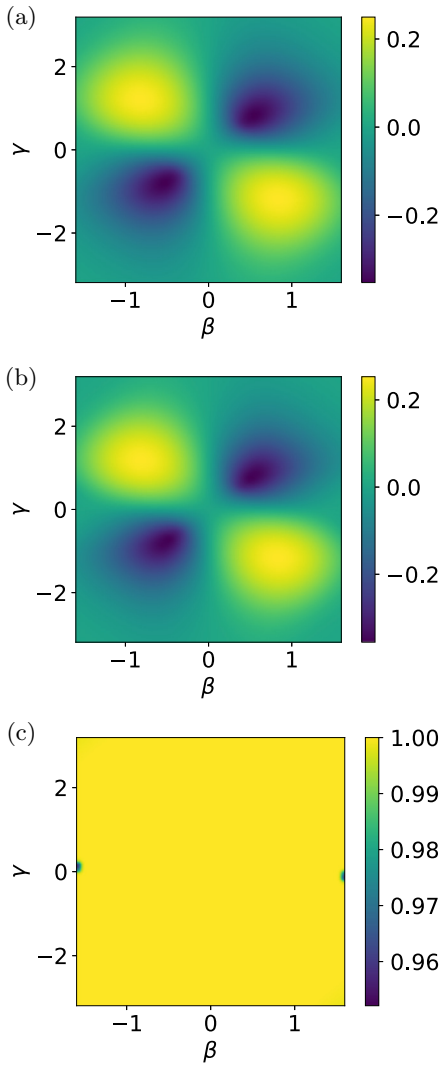


FIG. 6. QAOA optimization landscape at $p = 1$ and for $k = 4$ (enlargement: $2\times$). (a) Logarithmic success probability. (b) Exponential fit. (c) Correlation coefficient.

From these results, it first appears that the existence of the exponential scaling is not granted for all β and γ, with a correlation coefficient even approaching 0 in certain regions. This effect is particularly pronounced for larger k and these problematic regions exhibit a complex pattern that cannot be only explained by the magnitude of the angles. However, in the regions where an exponential scaling exists, the landscapes in the first two columns should coincide for sufficiently large n. We observe this is indeed the case in the figures. Interestingly, the regions where the success probability is maximized always exhibit an exponential scaling, as shown by the correlation coefficient close to unity. As for extrema, while 2-SAT and 4-SAT possess a single local maximum (up to the central symmetry), 8- and 16-SAT have spurious local maxima.

1. Quality of empirically determined optimal variational angles

In the main text, we compared exact (for $p = 1$ only) or empirical (for $p \in [1, 10]$) finite-size results to the prediction made by Proposition 1 for the infinite-size limit. These comparisons helped us build confidence that the analytic method, which is in principle restricted to sufficiently small angles, remains correct when evaluated with the pseudo-optimal angles we chose. Assuming the analytic method remains operational in a neighborhood of these variational

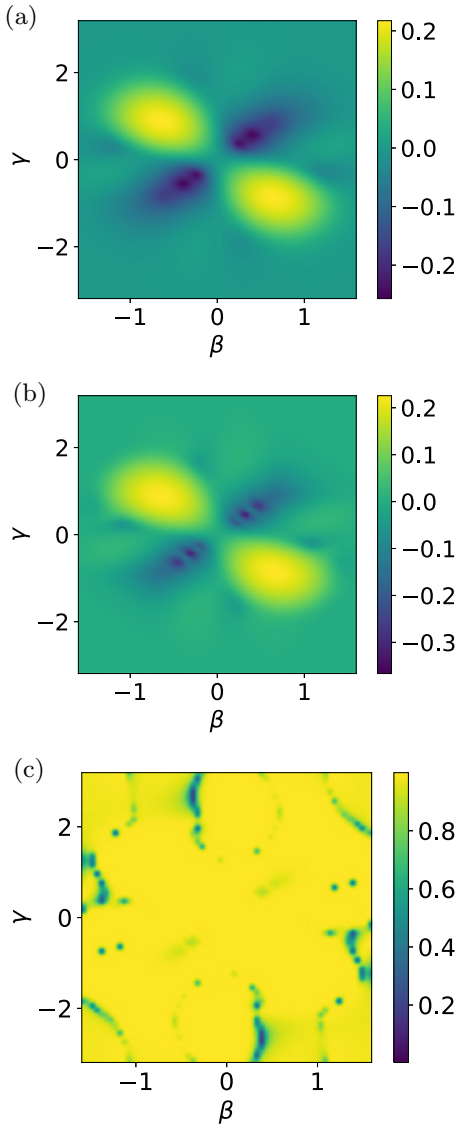


FIG. 7. QAOA optimization landscape at $p = 1$ and for $k = 8$ (enlargement: 2×). (a) Logarithmic success probability. (b) Exponential fit. (c) Correlation coefficient.

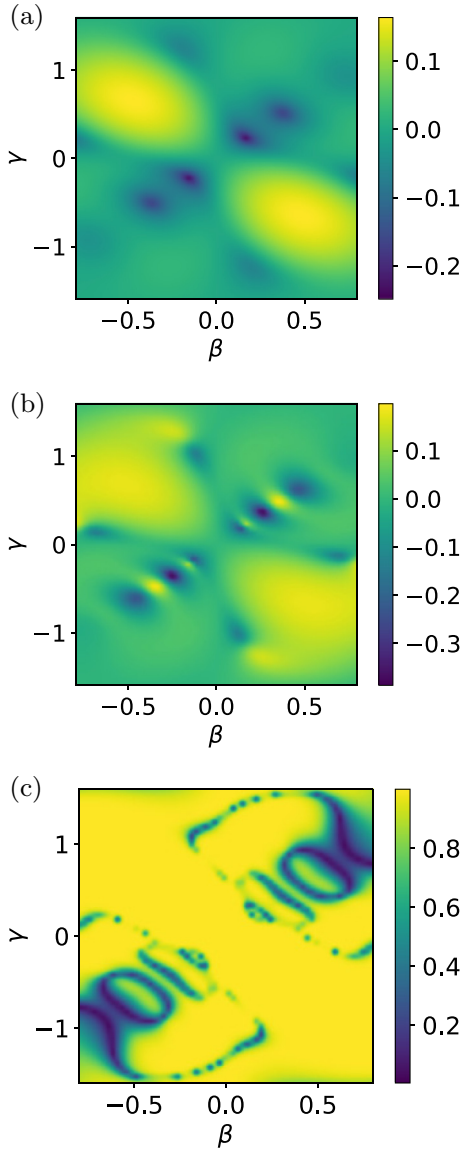


FIG. 8. QAOA optimization landscape at $p = 1$ and for $k = 16$ (enlargement: 4×). (a) Logarithmic success probability. (b) Exponential fit. (c) Correlation coefficient.

parameters, allowing for an *exact* evaluation of the exact success probability in this region, one may now wonder how far from optimal these angles are.

To investigate this, we reoptimize the success probability using the analytic formula, with the initial angles (empirically optimised for $n = 12$) as an educated guess. As usual, a simple gradient-descent algorithm is used. We then consider the relative variation of the analytically determined scaling exponent along the optimization, together with the relative variation of the angles. Figure 9 reports these results for 8-SAT and precisely defines the metrics just evoked. The p values used here are more restricted than in the main text due to the overhead of evaluations required by optimization as opposed to simple evaluation. These results show the angles determined by the simple empirical method described in Sec. IV A are very close to optimal, or at least to a local optimum. Relative variations for both the angles and the exponent along the optimization are of order a few percent. Surprisingly, the relative variation even appears to decrease with p, but we do not know how robust the last conclusion is due to the limited number of values of p considered and the dependence of these results on the small dataset used to determine approximately optimal angles.

2. Further experimental results for classical solvers for $k = 8$

We include all the classical experimental results on the complexity of solvers for 8-SAT in Table II.

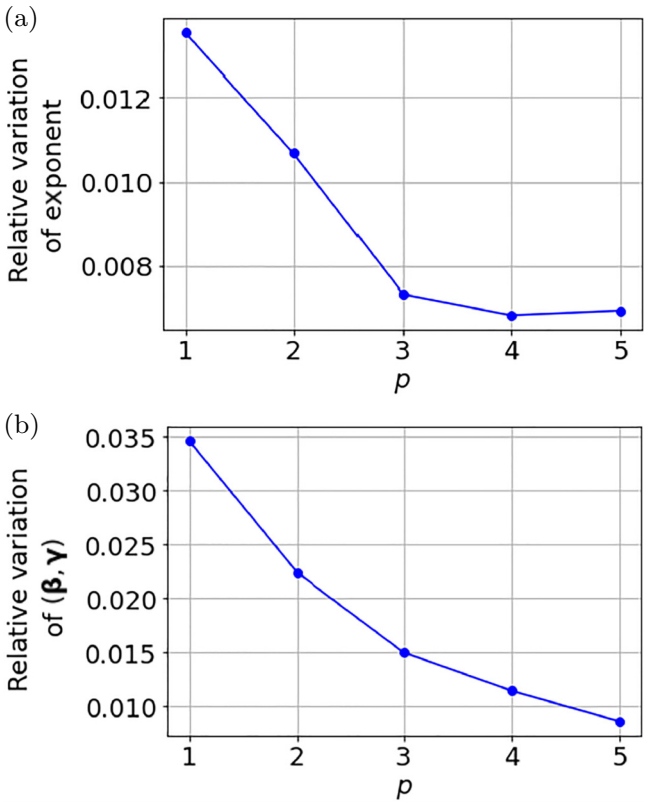


FIG. 9. Relative variation of scaling exponent and optimized angles after reoptimization from analytic method. The relative error for the exponent is defined as the ratio between the new and old value, minus 1. For angles, the distance between the old angles $(\boldsymbol{\beta}^{(\mathbf{i})}, \boldsymbol{\gamma}^{(\mathbf{i})})$ and the new ones $(\boldsymbol{\beta}^{(\mathbf{f})}, \boldsymbol{\gamma}^{(\mathbf{f})})$ is calculated by considering the representative of $(\boldsymbol{\beta}^{(\mathbf{f})}, \boldsymbol{\gamma}^{(\mathbf{f})})$ closest to the $(\boldsymbol{\beta}^{(\mathbf{i})}, \boldsymbol{\gamma}^{(\mathbf{i})})$ in order to account for 2π periodicity in the $\boldsymbol{\beta}_j$ and 4π periodicity in the $\boldsymbol{\gamma}_j$. The $\boldsymbol{\beta}$ components of the difference vector $(\boldsymbol{\beta}^{(\mathbf{f})} - \boldsymbol{\beta}^{(\mathbf{i})}, \boldsymbol{\gamma}^{(\mathbf{f})} - \boldsymbol{\gamma}^{(\mathbf{i})})$ are then rescaled by $(1/(\pi\sqrt{2p}))$, and the $\boldsymbol{\gamma}$ components by $(1/(2\pi\sqrt{2p}))$, mapping the 2-norm of the resulting vector into [0, 1]. (a) Exponent. (b) Angles.

[1] L. Grover, Quantum mechanics helps in searching for a needle in a haystack, Phys. Rev. Lett. **79**, 325 (1997).

[2] C. Dürr and P. Høyer, A quantum algorithm for finding the minimum, arXiv:quant-ph/9607014.

[3] E. Campbell, A. Khurana, and A. Montanaro, Applying quantum algorithms to constraint satisfaction problems, Quantum **3**, 167 (2019).

[4] R. Babbush, J. R. McClean, M. Newman, C. Gidney, S. Boixo, and H. Neven, Focus beyond quadratic speedups for error-corrected quantum advantage, PRX Quantum **2**, 010103 (2021).

[5] T. Hogg, Quantum search heuristics, Phys. Rev. A **61**, 052311 (2000).

[6] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, arXiv:1411.4028.

[7] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm applied to a bounded occurrence constraint problem, arXiv:1412.6062.

[8] B. Barak, A. Moitra, R. O'Donnell, P. Raghavendra, O. Regev, D. Steurer, L. Trevisan, A. Vijayaraghavan, D. Witmer, and J. Wright, Beating the random assignment on constraint satisfaction problems of bounded degree, arXiv:1505.03424.

[9] 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).

[10] F. G. S. L. 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:1812.04170.

[11] S. Boulebnane and A. Montanaro, Predicting parameters for the quantum approximate optimization algorithm for max-cut from the infinite-size limit, arXiv:2110.10685.

[12] S. Cai, C. Luo, and K. Su, Improving walksat by effective tie-breaking and efficient implementation, Comput. J. **58**, 2864 (2014).

[13] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, Quantum amplitude amplification and estimation, quantum computation and quantum information: A millennium volume, 53 (2002), arXiv:quant-ph/0005055.

[14] J. Basso, D. Gamarnik, S. Mei, and L. Zhou, Performance and limitations of the QAOA at constant levels on large sparse hypergraphs and spin glass models, arXiv:2204.10306.

[15] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PRXQuantum.5.030348 for detailed proofs of analytic results.

[16] A. Coja-Oghlan, Random constraint satisfaction problems, Electron. Proc. Theor. Comput. Sci. **9**, 32 (2009).

[17] S. Mertens, M. Mézard, and R. Zecchina, Threshold values of randomk-SAT from the cavity method, Random Struct. Algor. **28**, 340 (2006).

[18] B. Zhang, A. Sone, and Q. Zhuang, Quantum computational phase transition in combinatorial problems, arXiv:2109.13346.

[19] L. Li, M. Fan, M. Coram, P. Riley, and S. Leichenauer, Quantum optimization with a novel Gibbs objective function and ansatz architecture search, Phys. Rev. Res. **2**, 023074 (2020).

[20] P. K. Barkoutsos, G. Nannicini, A. Robert, I. Tavernelli, and S. Woerner, Improving variational quantum optimization using CVar, Quantum **4**, 256 (2020).

[21] A. Coja-Oghlan, Random constraint satisfaction problems, EPTCS 9, 2009, pp. 32–37 (2009), arXiv:0911.2322.

[22] M. Merkle, Jensen's inequality for medians, Stat. Probab. Lett. **71**, 277 (2005).

[23] C. H. Papadimitriou, in *Proceedings of the 32nd Annual Symposium on Foundations of Computer Science*, SFCS '91 (IEEE Computer Society, USA, 1991), p. 163.

[24] B. Selman, H. Kautz, and B. Cohen, Local search strategies for satisfiability testing, cliques, coloring, and satisfiability: 2nd DIMACS implementation challenge, 521 (1993).

[25] T. Schoning, in *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)* (IEEE Computer Society, USA, 1999).

[26] A. Ignatiev, A. Morgado, and J. Marques-Silva, in *SAT* (Springer International Publishing, Cham, 2018), p. 428, https://link.springer.com/chapter/10.1007/978-3-319-94144-8_26.

[27] M. J. H. Heule, M. Järvisalo, and M. Suda, SAT 2018 competitions and evaluations, J. Satisf., Boolean Model. Comput. **11**, 133 (2019).

[28] R. Marino, G. Parisi, and F. Ricci-Tersenghi, The backtracking survey propagation algorithm for solving random k-SAT problems, Nat. Commun. **7**, 12996 (2016).

[29] T. Sejourne, Survey propagation, https://github.com/thibsej/SurveyPropagation (2018).

[30] X.-Z. Luo, J.-G. Liu, P. Zhang, and L. Wang, Yao.jl: Extensible, Efficient Framework for Quantum Algorithm Design, Quantum **4**, 341 (2020).

[31] E. Farhi, J. Goldstone, S. Gutmann, and L. Zhou, The quantum approximate optimization algorithm and the Sherrington-Kirkpatrick model at infinite size, Quantum **6**, 759 (2022).

[32] D. Achlioptas and C. Moore, Random k-SAT: Two moments suffice to cross a sharp threshold, SIAM J. Comput. **36**, 740 (2006).

[33] Notably, the prediction for the leading exponential contribution to the scaling of the amplitude still holds numerically even when $\Phi_{\beta,\widetilde{\gamma}}$. In this case, we still conjecture that $\Phi_{\beta,\widetilde{\gamma}}$ has smaller real value when evaluated at the secondary critical point, constituting a plausible explanation for the robustness of the result. However, a rigorous application of the saddle-point method remains challenging since the large $\widetilde{\gamma}$ regime also requires to consider more complicated integration contours in the complex plane than the straight line, which suffices for small $\widetilde{\gamma}$.

[34] S. Banach, Sur les operations dans les ensembles abstraits et leur application aux equations integrales, Fundam. Math. **3**, 133 (1922).

[35] However, it may be that to each choice of $\boldsymbol{\beta}$ angles corresponds an upper bound on the γ_j for the method to work, and that the upper bound be more favourable for some $\boldsymbol{\beta}$ than for others. Still, this upper bound will be constant for each $\boldsymbol{\beta}$ and not, e.g., $\mathcal{O}\left(n^{-1}\right)$.

[36] B. Kostka, *Sports programming in practice*, Master's thesis, University of Wroclaw, 2021.