npj | quantum information

Article

Published in partnership with The University of New South Wales

https://doi.org/10.1038/s41534-025-01082-1

Toward a linear-ramp QAOA protocol: evidence of a scaling advantage in solving some combinatorial optimization problems

Check for updates

J. A. Montañez-Barrera[1] ✉ & Kristel Michielsen[1,2,3]

The quantum approximate optimization algorithm (QAOA) is a promising algorithm for solving combinatorial optimization problems (COPs), with performance governed by variational parameters $\{\gamma_i, \beta_i\}_{i=0}^{p-1}$. While most prior work has focused on classically optimizing these parameters, we demonstrate that fixed linear ramp schedules, linear ramp QAOA (LR-QAOA), can efficiently approximate optimal solutions across diverse COPs. Simulations with up to $N_q = 42$ qubits and $p = 400$ layers suggest that the success probability scales as $P(x^*) \approx 2^{-\eta(p)N_q + C}$, where $\eta(p)$ decreases with increasing p. For example, in Weighted Maxcut instances, $\eta(10) = 0.22$ improves to $\eta(100) = 0.05$. Comparisons with classical algorithms, including simulated annealing, Tabu Search, and branch-and-bound, show a scaling advantage for LR-QAOA. We show results of LR-QAOA on multiple QPUs (IonQ, Quantinuum, IBM) with up to $N_q = 109$ qubits, $p = 100$, and circuits requiring 21,200 CNOT gates. Finally, we present a noise model based on two-qubit gate counts that accurately reproduces the experimental behavior of LR-QAOA.

Finding high-quality solutions for COPs is perceived as one of the main applications of quantum computation in the near future. In the gate-based regime, QAOA[1] has become one of the most studied quantum algorithms for solving COPs. There are different factors for the extensive study of QAOA. Firstly, parametric unitary gates can effectively represent the Hamiltonian of the COPs, where the ground state encodes the optimal solution of the problem[2,3]. Moreover, QAOA has a performance guarantee in the limit of infinite layers resembling the quantum adiabatic algorithm[1,4]. Additionally, QAOA needs fewer resources (e.g., number of gates and qubits) compared to other quantum algorithms and can be tested on current state-of-the-art quantum hardware[5–7]. Furthermore, classical methods find it hard to solve large instances of COPs with practical applications[8], and therefore, finding alternative ways to solve them is needed. Ultimately, the goal of quantum optimization algorithms, as exemplified by QAOA, is to demonstrate advantages in solving optimization problems, be it in terms of energy efficiency, time-to-solution (TTS), or solution quality compared to classical methods.

In the simplest version of QAOA, the cost Hamiltonian of a combinatorial optimization problem is encoded in a parametric unitary gate along with a "mixer", a second parametric unitary gate that does not commute with the first unitary gate. In this context, parameters $\gamma = [\gamma_0, \ldots, \gamma_{p-1}]$ and $\beta = [\beta_0, \ldots, \beta_{p-1}]$ for the cost and mixer Hamiltonians, respectively, are adjusted to minimize the expectation value of the cost Hamiltonian for $i = 0, \ldots, p-1$ layers of QAOA. Since its conception, a classical algorithm was suggested to find the QAOA γ and β parameters[1]. This makes QAOA fall in the category of variational quantum algorithms (VQA)[9]. However, these algorithms have exhibited a limited/poor performance advantage as the classical optimization part finds it hard to escape local minima when searching for γ and β parameters[10,11]. The barren plateau is another challenge in QAOA. It refers to regions in the cost function landscape where the gradient is nearly zero, making it hard to find QAOA parameters via gradient-based optimization[12].

Modest progress has been made by considering QAOA as a VQA, with major studies conducted in regimes of a few qubits and shallow circuits[5]. Deep QAOA circuits, when viewed through the lens of VQA, lead to a pessimistic conclusion regarding their universal applicability[13]. Moreover, implementations on real hardware face an even greater challenge; the noise inherent in current quantum devices makes the search for the minima of the objective function unfeasible after only a few QAOA layers[13,14].

[1]Jülich Supercomputing Centre, Institute for Advanced Simulation, Forschungszentrum Jülich, 52425 Jülich, Germany. [2]AIDAS, 52425 Jülich, Germany. [3]RWTH Aachen University, 52056 Aachen, Germany. ✉e-mail: j.montanez-barrera@fz-juelich.de

Alternatively to this methodology, one can fix the γ and β parameters following some protocol, similar to what quantum annealing (QA) does[15,16]. In this scenario, no further classical optimization is needed.

Initial evidence supporting the effectiveness of fixed-parameter QAOA was presented by Brandao et al.[17]. They demonstrated that fixed parameters exhibit consistent performance regardless of the problem or problem size, suggesting the potential reduction of the outer loop of classical optimization in QAOA.

Various protocols have been proposed to fix these parameters. In ref. 11, Krementski et al. found a set of QAOA parameters with consistent performance to find optimal solutions using a fixed LR-QAOA protocol. They tested this methodology using the Hamiltonian of different molecules, an Ising Hamiltonian, and the 3-SAT problem for intermediate to large p. Another attempt to fix the QAOA protocol is proposed in ref. 18, in which the authors presented QAOA as a second-order time discretization of QA referred to as approximate quantum annealing (AQA). In ref. 19 Hess et al. proposed the Trotterized adiabatic evolution (TAE), an idea similar to AQA but using a fixed sinusoidal schedule. LR-QAOA can be considered as an AQA protocol with a linear annealing schedule.

In ref. 20, we proposed fixed schedules transferring optimal γ and β parameters between different COPs. We found that sometimes γ and β parameters that work well for a COP in the form of Eq. (2) give good results on other COPs with different structures. Specifically, we found that parameters optimized for the bin packing problem (BPP) can be translated to Maxcut, Maximal independent set (MIS), portfolio optimization (PO), and traveling salesman problem (TSP), giving a quadratic speedup over random guessing on all of them. This suggests that there are effective QAOA protocols that work for different problems. This information led us to the results of the present work.

Recently, Kremenetski et al. have explained the behavior of LR-QAOA and, in general, of the gradually changing schedules using the discrete adiabatic theorem involving a wrap-around phenomenon[21].

In this paper, we extend the study of LR-QAOA schedules to different COPs, presenting numerical and experimental evidence that LR-QAOA constitutes an effective QAOA protocol, i.e., the set of γ and β parameters from a linear ramp schedule works effectively for many problems and problem sizes in combinatorial optimization. We test this protocol using MIS, BPP, TSP, Maxcut, weighted maximum cut (WMaxcut), 3-regular graph maximum cut (3-Maxcut), Knapsack (KP), PO, maximum 2 Boolean satisfiability problem (Max-2-SAT), and maximum 3-SAT (Max-3-SAT). We use random instances of these COPs with problem sizes ranging from 4 to 42 qubits and p from 3 to 400. For large problems, we simulate them using JUQCS-G software[18] on JUWELS Booster, a cluster of 3744 NVIDIA A100 Tensor Core GPUs, integrated into the modular supercomputer JUWELS[22,23].

In these cases, the average probability of success over the 100 random instances seems to follow a scaling that can be described by probability $(x^*) = 2^{-\eta(p)N_q+C}$ for a $\eta(p)$ decreasing with p and a constant C. We extend the analysis to fully connected random WMaxcut. The WMaxcut is both NP-Hard[24] and APX-Hard[25] problems.

We find a scaling improvement in terms of the time-to-solution (TTS)[14] when using LR-QAOA compared to SA, TABU search, and B&B for solving random instances of WMaxcut. This evidence complements the recent findings in ref. 26 where a scaling advantage is observed in a fixed QAOA protocol for solving a classical intractable COP known as low autocorrelation binary sequences (LABS) and on k-SAT problems[27].

We extend the analysis to real quantum hardware. Using IonQ Aria (*ionq_aria*), Quantinuum H2-1 (*quantinuum_H2*)[28], IBM Brisbane (*ibm_brisbane*), IBM Kyoto (*ibm_kyoto*), IBM Osaka (*ibm_osaka*), and IBM fez (*ibm_fez*), we run WMaxcut problems ranging from 5 to 109 qubits and p from 3 to 100. We find that there is an effective number of layers, p_{eff}, for which the best performance is obtained using each device.

In the case of IBM devices, $p_{\text{eff}} = 10$, on *ionq_aria* $p_{\text{eff}} = 10$, and on *quantinuum_H2* $p_{\text{eff}} = 50$. Remarkably, for the largest problem size, 109 qubits and $p = 100$, we observe that LR-QAOA still possesses an improvement over random sampling in *ibm_kyoto* and *ibm_osaka*. For a comparative analysis between the different vendors, we test a 25-qubit WMaxcut problem on them, *quantinuum_H2* gives the best performance with a probability$(x^*) = 0.08$ at $p = 50$.

We present a noise model of LR-QAOA that fits depolarizing noise simulation and experiments on ibm_fez and an emulator of Quantinuum H1-1. The noise model depends only on the number of 2-qubit gates and a noise parameter associated with the QPU. We observe that there is an interplay between the noise pushing the system towards a maximally mixed state and LR-QAOA driving the system towards the minimum energy of the cost Hamiltonian.

Results

Simulations

Figure 1a–d shows the average probability of success for different COPs vs. the number of qubits. We test 100 random problems with 10–35 qubits using the LR-QAOA with $p = 10$ to 200. The Δ_γ and Δ_β values are scanned for each problem size for one instance, and that value is used for the remaining 99 instances. Although this methodology is not optimal, since ideally, individual parameters should be tuned for each case, we adopt this approach to manage the growing simulation costs associated with a large number of qubits. Moreover, this strategy highlights an important feature of LR-QAOA, even without fine-tuning parameters for each instance, the algorithm consistently achieves good performance. In Supplementary Note 1, we present in detail the methodology and the values used. In these problems, we observe that the number of layers affects the scale of the probability of success with a relation that can be described by

$$Probability\,(x^*) = 2^{-\eta(p)N_q+C}, \tag{1}$$

with an $\eta(p)$ that is a function of p, and C a constant. The perceived scaling still needs to be corroborated at a larger problem size to confirm that the probability of success indeed decreases exponentially. A similar scaling QAOA behavior has been observed for k-SAT problems in ref. 27. If this holds, it means that there is an exponential improvement in LR-QAOA achieved by increasing the number of layers linearly. This does not necessarily mean that the problems are hard and that the best classical solvers for them exhibit the perceived exponential scaling of LR-QAOA.

Figure 1e shows the fitting values of $\eta(p)$ for each p using the information of Fig. 1a–d. The four models exhibit similar $\eta(p)$ behavior, decaying quickly with the number of layers. The fitted $\eta(p)$ implies an exponential improvement as the number of layers increases. For example, if the WMaxcut scale holds at $N_q = 100$, using LR-QAOA with $p = 10$ the probability $(x^*) = 2 \times 10^{-7}$ while probability $(x^*) = 0.2$ with $p = 200$. Figure 1f shows the relative error of the difference between the value predicted by the fitting curve and the actual values for each p, i.e., $\varepsilon = |1 - \frac{2^{-\eta(p)N_q+C}}{probability\,(x^*)}|$. As the number of layers increases, the error decreases in 3 out of 4 cases and remains below 6%.

In Supplementary Note 2, we present the details about the scaling factor $\eta(p)$ and its relation with the minimum number of qubits used in the cutoff of the fitting function. The previous version of this paper included a conjecture that the probability of success scales as probability $(x^*) = 2^{-\eta N_q/p}$, with the further evidence of this section, the model of the scale of probability (x^*) has changed. Previous results and extended simulations are presented in Supplementary Note 3. In the next section, we present a comparative analysis of LR-QAOA and several classical solvers.

Scaling comparison

In Fig. 2a—left, we show the TTS of SA vs. the TTS of TABU search. There is a high correlation coefficient (PCC = 0.7)[29] between the solvers' TTS, indicating that random problems requiring longer TTS for one solver tend to require longer TTS for the other as well. In contrast, the PCC between TABU and CPLEX is 0.23, reflecting a weak correlation; thus, what is

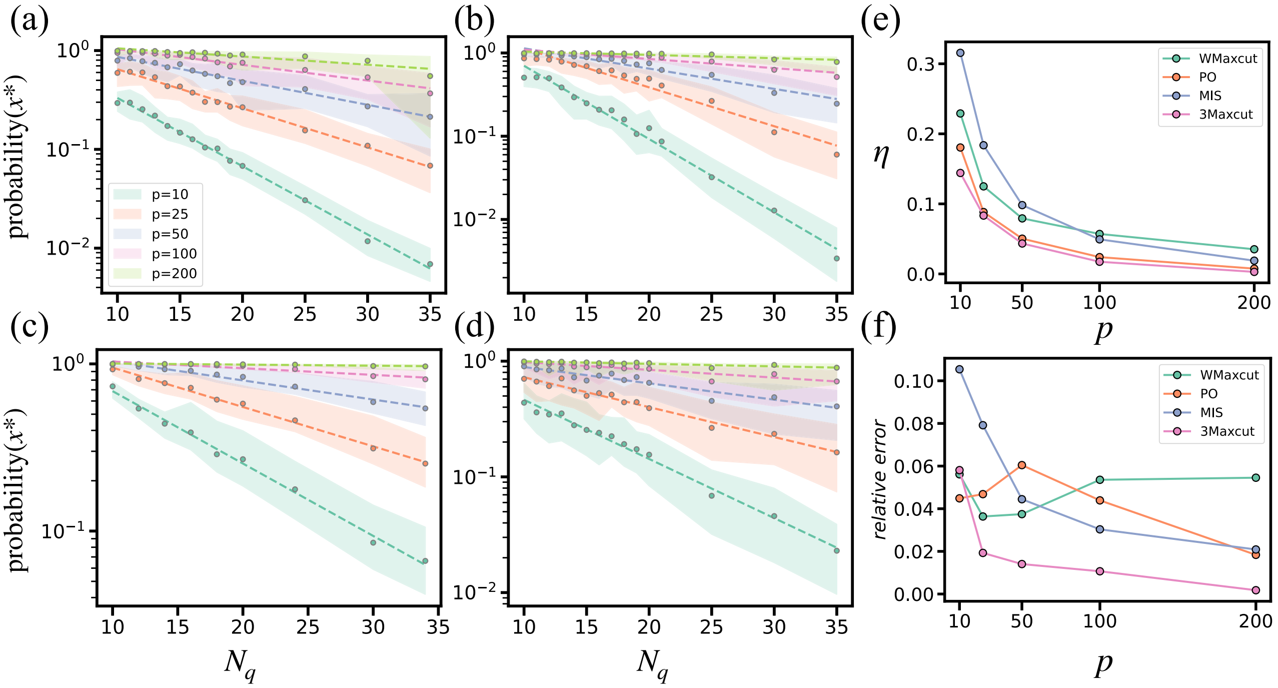

Fig. 1 | Simulation of LR-QAOA for different COPs. Probability of success for 100 random instances of **a** WMaxcut, **b** MIS, **c** 3-Maxcut, and **d** PO. The shaded region represents the quartiles 1 and 3 over the 100 cases. The different colors represent the number of LR-QAOA layers (see legend). Dashed lines represent the conjectured scaling $2^{-\eta(p)N_q+C}$ for each p. **e** Fitted η vs. number of LR-QAOA layers for the mean value of the problems. **f** Relative error of probability (x^*) calculated using the fitting parameters vs. the number of LR-QAOA layers.

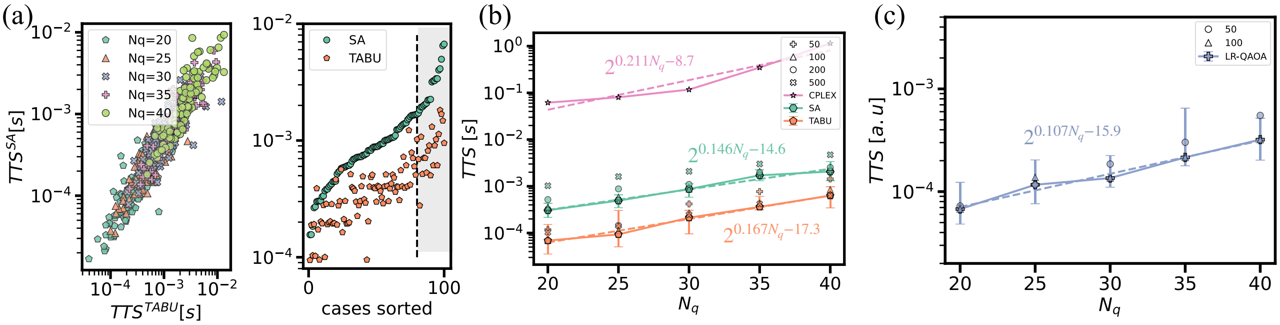

Fig. 2 | Scaling comparison in terms of TTS vs. number of qubits for classical solvers and LR-QAOA for fully connected random problems of WMaxcut. a The left plot shows the TTS of SA vs. TABU search for different numbers of qubits. Each marker represents a problem. The right plot shows the TTS vs. the 100 random cases sorted by TTS of SA for 40-qubit size problems. The shaded region highlights the 20 problems with the longest TTS. **b** TTS of SA, TABU search, and CPLEX. Three configurations are used for SA and TABU in each case: 50, 100, 200, and 500 sweeps, and the lowest TTS is used for the scaling. **c** TTS of LR-QAOA for $p = 50$ and 100, the scaling is taken over the minimum TTS at each time-point. The markers represent the median value, and the error bars represent the first and third quartiles of the 20 cases distribution.

considered difficult for TABU is not necessarily difficult for CPLEX. Based on this information, we select the 20 cases with the longest TTS for SA out of the 100 random instances generated for each problem size. The shaded region in Fig. 2a—right highlights the problems selected for the 40-qubit problem size.

Figure 2b shows the scaling of SA and TABU search, and CPLEX B&B in seconds. We use sweeps ranging from 50 to 500 for the heuristic solvers and choose the minimum TTS in each problem size case. In the case of SA, the best TTS is found using 50–100 sweeps. This is a consequence of the time required to implement the sweeps. While more sweeps generally lead to better solutions, the improvement does not always justify the additional evaluation time. Therefore, there is a tradeoff between the number of sweeps and the evaluation time, indicating that an optimal number of sweeps exists for a given problem size.

The case of TABU is similar to SA; for a small number of qubits, the optimal number of sweeps is around 100. However, as the number of qubits increases, the best configurations shift. At $N_q = 40$, both 200 and 500 iterations yield similar TTS. The scaling for SA is slightly better than TABU search, with a shorter TTS in all cases. At the problem sizes considered, the advantage of TABU search in maintaining a list of previously visited solutions does not appear to be necessary. As a result, the additional computational cost of comparing against this list at each iteration might affect the TTS. The TTS of CPLEX is several orders of magnitude higher than that of the other solvers, and the corresponding fitting function does not appear to be reliable.

Figure 2c presents the TTS of LR-QAOA in arbitrary units, which must be rescaled according to the two-qubit gate time, t_g, of a given quantum computer. We use $p = 50$ and $p = 100$ and choose the best TTS from them. For visualization, we use a $t_g = 2.5 \times 10^{-9}$ that matches the time of TABU search at $N_q = 20$ and corresponds to a gate time of $t_g = 2.5$ ns. Comparing the models, LR-QAOA shows a potential scaling advantage over the other solvers. Achieving competitive scaling with LR-QAOA would likely require depths beyond $p > 100$. The relative error of the perceived scaling is 0.211 for CPLEX, 0.023 for SA, 0.019 for TABU, and 0.011 for LR-QAOA.

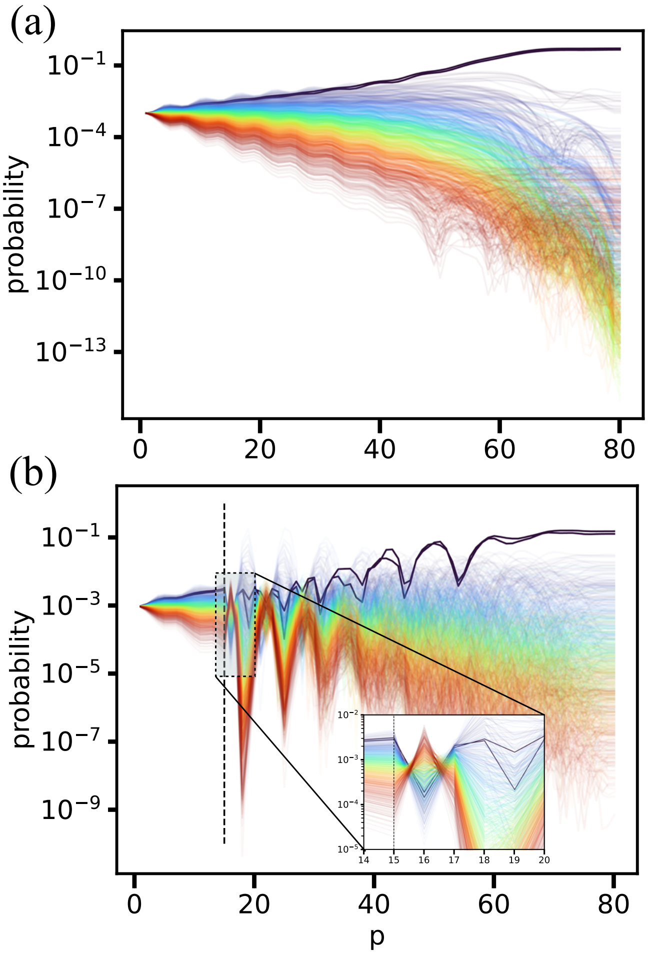

Fig. 3 | Bitstring solutions evolution under the LR-QAOA protocol. Probability of observing the different bitstring solutions of a 10-qubit MIS problem for the LR-QAOA protocol with $p = 80$ **a** noiseless evolution **b** a layer of X gates applied at $p = 15$. The lines on the graph represent various eigenvalues, with darker blue indicating lower energy and darker red indicating higher energy. The two values highlighted in dark blue correspond to the optimal solutions for the given problem.

Experiments

In this section, we show numerically and experimentally how noise affects LR-QAOA. Before moving to numerical simulations of LR-QAOA under depolarizing noise, we want to show LR-QAOA's ability to overcome errors. In Fig. 3a, the noiseless evolution of the eigenvalues of the cost Hamiltonian for LR-QAOA is presented. In Fig. 3b, the same protocol is shown, but this time depicts the evolution under full inversion of the qubits using a layer of X gates applied at $p = 15$. At $p = 16$, the eigenvalues experience a full inversion of probabilities, with high-energy bitstrings now having a large probability. This is quickly corrected by LR-QAOA, increasing the probability of getting the optimal solution. This inversion comes with the price of a reduction in the success probability from 96.1% in (a) to 28.5% in (b). Therefore, even if noisy conditions deteriorate the success probability, the errors do not completely remove the logic of the circuit.

Figure 4 shows simulations of different LR-QAOA configurations for the WMaxcut varying p, λ, N_q, and E_d under the depolarizing noise model. We make a distinction in this figure by the number of qubits, but the markers represent cases with different p, λ, and E_d as well. Therefore, even if different parameters could have an impact on the solution, the noise can be well described by the number of 2-qubit gates, the depolarizing noise, and a single fitting parameter. The fitting parameter in Eq. (19) is $k_0 = 1.82$ for WMaxcut; it might be COP dependent, but further analysis is needed.

We use this noisy model on experimental results for ibm_fez and H1-1E. The results are shown in Fig. 4b for 5 to 15-qubit problems of fully connected WMaxcut. For the case of ibm_fez, we use the parity twine chains (PTC)[30,31] strategy to encode the LR-QAOA quantum circuit into a 1D-chain of qubits of the QPU. The number of 2-qubit gates for each layer of LR-QAOA is $N_{2q} = N_q(N_q - 1)/2$ for H1-1E and $N_{2q} = N_q^2 - 1$ for ibm_fez. Based on the noise model Eq. (15), the fitted average error rates are $\lambda = 2.5 \times 10^{-4}$ for H1-1E and $\lambda = 25 \times 10^{-4}$ for ibm_fez. The λ can be interpreted as the average 2-qubit error of the QPU for the LR-QAOA on WMaxcut problems. These errors are similar to the average 2-qubit gate error measured by Randomized benchmarking (RB)[32], which is 36×10^{-4} for ibm_fez and 9×10^{-4} for H1-1.

Figure 4c shows the estimated overlap probability, p_{ovl}, as a function of the number of two-qubit gates, under the noise levels observed in ibm_fez and H1-1E. To achieve an overlap of $p_{ovl} = 0.1$ (i.e., 10% of the ideal probability), the maximum number of two-qubit gates should be limited to approximately 733 for ibm_fez and 7331 for H1-1E. This limit is independent of the problem size, and therefore, can be used as an estimate of how many 2-qubit gates one can use until the overlap is too short to observe the optimal solution. Even if the error grows exponentially with the number of 2-qubit gates, the probability of success also grows exponentially as the

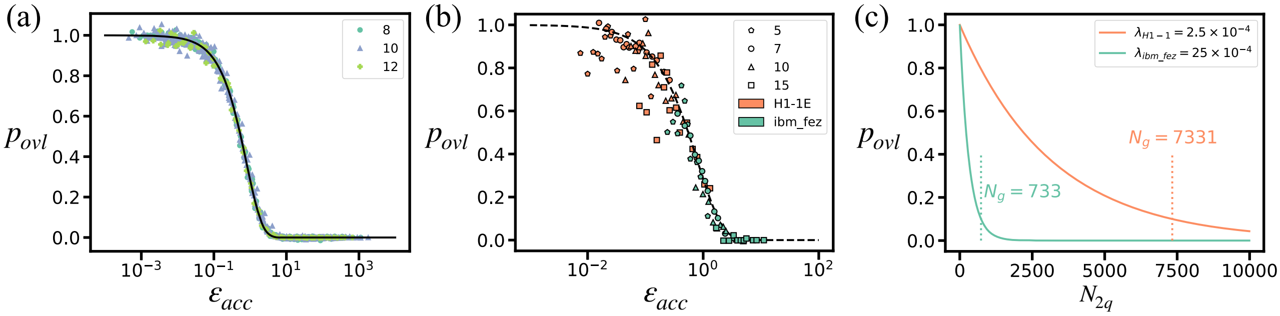

Fig. 4 | Overlap between the success probability of LR-QAOA under noise and the ideal probability vs the accumulated error. a Noise simulation using depolarizing noise for 8, 10, 12-qubit problems with $p = 10, 20, 40, 25$ depolarizing noise errors between $\lambda = 10^{-5}$ and 1, and 3 random graphs with edge density $E_d = 0.2, 0.5$, and 1. The fitting parameter in Eq. (19) is $k_0 = 1.82$, which corresponds to the black line. **b** Overlap of the success probability on real QPUs, ibm_fez (green) and Quantinuum H1-1E (orange) for 5–15 qubits. The errors perceived using the noise model are $\lambda_{H1-1E} = 2.5 \times 10^{-4}$ and $\lambda_{ibm\_fez} = 25 \times 10^{-4}$. **c** Overlap probability vs. the number of two-qubit gates for noise models of ibm_fez and H1-1E. The dotted line represents the number of 2-qubit gates where an overlap of 10% is reached.

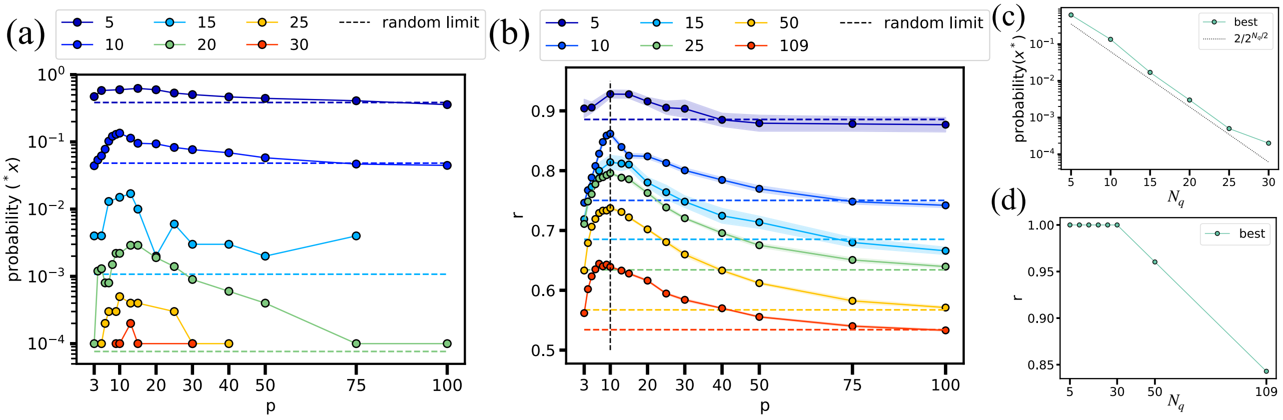

Fig. 5 | Experimental results of the LR-QAOA protocol on *ibm_brisbane* for WMaxcut problems. a Probability of success vs. number of layers of the LR-QAOA protocol. Colors represent the number of qubits from 5 to 30 qubits. The dashed lines "random limit" represent the success probability of a random sampler after the mitigation technique is applied for each problem with the same color. **b** Approximation ratio vs. LR-QAOA layers. Colors represent problems from 5 to 109 qubits. The shaded region represents the standard deviation over 10 sets of 1000 shots. **c** Best probability of success of **a** vs. number of qubits. The dashed line representing a quadratic speedup is added as a reference. **d** Best approximation ratio of all the samples in the experiment vs. number of qubits.

number of layers grows. Therefore, there is a point where the trade-off between noise and LR-QAOA reaches an equilibrium point, and a maximum probability of success is obtained. A decrement in one order of magnitude in the noise leads to an increment in 1 order of magnitude in the number of 2-qubit gates that can be used, for instance, a $\lambda = 2.5 \times 10^{-5}$ and expecting an overlap of 10% allows to the use $N_g = 73, 310$. In the Supplementary Note 4, we extend the depolarizing noise model study to the IonQ Aria QPU.

Figure 5a shows the probability of success vs. the number of LR-QAOA layers of random cases of the WMaxcut for variables from 5 up to 30 running on *ibm_brisbane*. We use 10,000 samples for each problem size. We do not include information for larger problem sizes because no optimal solution is observed for them. The dashed line represents the probability of success of mitigated samples of a random sampler (see Section Mitigation: Hamming distance 1). In other words, circles above the dashed line of its respective color cannot be explained as the result of a random process and therefore can be attributed to LR-QAOA. To contextualize our outcomes, observing the optimal solution for the 30 qubits problem with a random sampler requires, in the worst case $2^{30}/2 = 536, 870, 912$ evaluations of the cost function. In our experiment, we find the optimal solution 2 times at 13 layers using LR-QAOA on a noisy device using 10, 000 samples and the mitigation technique (see Section Mitigation: Hamming distance 1 for the mitigation technique). This means 10, 000 × 30 further evaluations, representing an improvement over random guessing of 536.870.912/ 310.000 ≈1732 times.

Figure 5b shows the approximation ratio of the instances of WMaxcut from 5 to 109 qubits using LR-QAOA on *ibm_brisbane*. The vertical dashed line at $p = 10$ indicates the number of layers for which the best performance of LR-QAOA is obtained. After $p = 10$, the system is slowly moved toward a maximally mixed state. At $p = 100$, it is reached in all the cases. We attribute this phenomenon to the nature of LR-QAOA, which initially improves faster than the destructive effects of noise. However, above a particular noise threshold, noise begins to dominate, leading to a monotonic decrease in the quality of the solutions obtained. This leads to an interesting behavior, for instance, at $p = 3$ the approximation ratio is the same as that at $p = 40$ for the 109-qubit case, despite the latter requiring roughly 13 times more time and 2-qubit gates than the former.

Figure 5c shows the maximum probability over all layers vs. the number of qubits for the 1D WMaxcut experiment. The dashed line that represents the quadratic speedup $2/2^{N_q/2}$ over random sampling is added as a reference. In the experiments, the highest probability occurs within the range of $p = 10$ to 13. The experiments hold a similar decay to the quadratic speedup, with a shift that can be attributed to the mitigation technique. Additionally, Fig. 5d shows the best approximation ratio among all the samples vs. the number of qubits. The maximum average approximation ratio for the 109-qubit experiment is $r = 0.64$, with the best sample having a $r = 0.84$.

Figure 6a shows the approximation ratio of the 109 qubits WMaxcut problem using LR-QAOA from $p = 3$ to $p = 100$. At $p = 10$, the maximum approximation ratio is reached for the three devices *ibm_brisbane*, *ibm_kyoto*, and *ibm_osaka*. The noise at larger p leads the system towards a maximally mixed state, so we include the dashed line that represents the approximation ratio $r = 0.5326$ of a random sampler after the mitigation technique is applied. Unexpectedly, at $p = 100$, results for *ibm_kyoto* and *ibm_osaka* still deviate from the random sampler and therefore some information of the LR-QAOA protocol is present. The circuit used requires 21200 CNOT gates and a total time of ≈132 μs. This is an indication of the resilience of the LR-QAOA to noise.

At this scale, we surpass the point where exact classical simulation of LR-QAOA is feasible both in terms of the number of qubits and depth of the circuit. The 1D connectivity of the graph makes the simulation of the LR-QAOA suited for approximation methods based on tensor networks. If the problem connectivity is increased, making it hard to be simulated classically, this experiment might be presented as a quantum utility experiment. This means that a classical algorithm cannot mimic the sampling properties of LR-QAOA at large p and N_q. There are different techniques proposed for addressing the simulation of quantum supremacy[33] or utility[34] experiments after their publication (e.g., refs. 35,36), so the validation of this remains subject to evaluation within the research community. Independent of the answer, these results indirectly imply the efficacy of LR-QAOA to solve COPs in scenarios involving more than 42 qubits.

Figure 6b presents a comparative analysis between *ionq_aria*, *quantinuum_H2*, and *ibm_brisbane* in solving a 25-qubit instance of the WMaxcut problem. The number of samples is 10000 for the noiseless simulator and *ibm_brisbane*, 1000 for *ionq_aria*, and 50 for *quantinuum_H2*. The performance of *quantinuum_H2* stands out, achieving a maximum approximation ratio of $r = 0.95$ at $p = 50$, compared to $r = 0.98$ at $p = 50$ of the noiseless simulator, *ibm_brisbane*'s $r = 0.80$ at $p = 10$, and *ionq_ariar* = 0.90 at $p = 10$.

From a time perspective, executing an instance of WMaxcut LR-QAOA for $p = 10$ on *ibm_brisbane* requires approximately ≈13.2 μs, whereas *ionq_aria* completes the same task in about ≈144 ms, and *quantinuum_H2* in ≈36 ms. The three devices successfully identify the optimal solution for this problem, with *quantinuum_H2* achieving a maximum probability of success of 0.10 at $p = 75$, *ionq_aria* achieving a maximum probability of success of 0.008 at $p = 10$, and *ibm_brisbane* reaching 0.0005

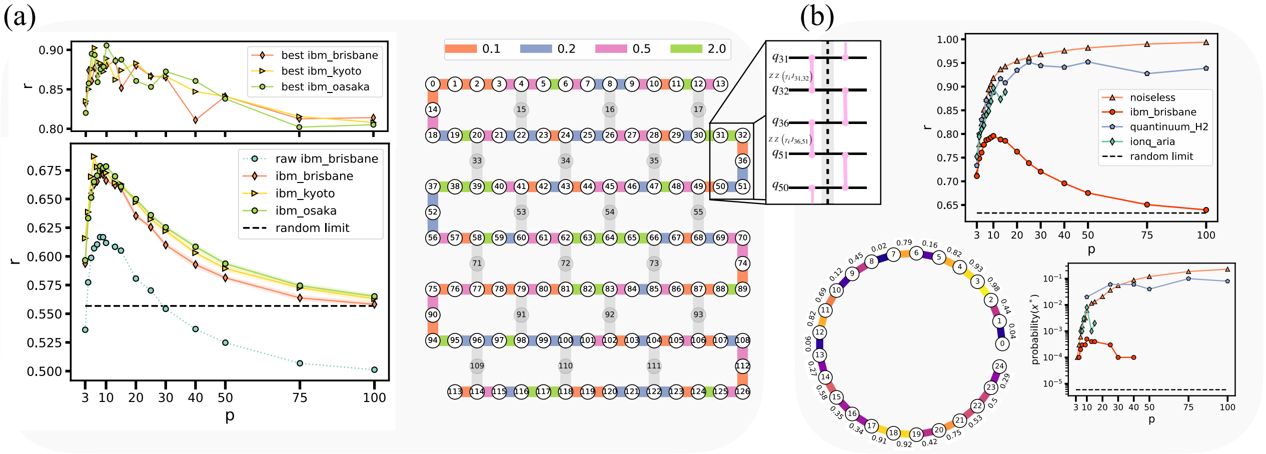

Fig. 6 | Experimental results of LR-QAOA on different quantum devices. a 109-qubit WMaxcut experiment on IBM Eagle devices using LR-QAOA. The upper-left plot represents the best approximation ratio observed from the 10,000 samples at each p. The bottom-left plot shows the average approximation ratio, where the dotted line represents the raw result from *ibm_brisbane* and the solid lines represent the mitigated results over the three different IBM devices. The black dashed line is the limit where the system reaches the maximally mixed state. The right plot shows the IBM Eagle layout with the 1D random WMaxcut on it. The colors represent the random weights chosen from 4 possible values 0.1, 0.2, 0.5, and 2.0 for each edge in the graph. The inset shows how the 2-qubit gates are implemented using only a depth of 2 for each LR-QAOA layer. The shaded region represents the standard deviation over 10 sets of 1000 shots. **b** 25-qubit WMaxcut experiment comparative analysis for 3 different vendors. The bottom-left plot presents the graph of the WMaxcut selected with the corresponding weight values. In the upper plot, the approximation ratio of the devices with the triangles, circles, diamonds, and pentagons corresponds to the mitigated results from a noiseless simulation, *ibm_brisbane*, *ionq_aria*, and *quantinuum_H2*, respectively. Because of a limitation in the maximum number of single-qubit and two-qubit gates, there are no results above $p = 15$ using *ionq_aria*. In the bottom right plot, the probability of success for the same experiment is shown.

at $p = 10$. This means that *quantinuum_H2* is 12.5 times more effective in finding the optimal solution than *ionq_aria*, and 200 than *ibm_brisbane*.

However, the accuracy gain for *quantinuum_H2* does not fully compensate for the time required for sampling. In other words, for every optimal sample obtained from *quantinuum_H2*, one could obtain approximately 2700 samples on *ibm_brisbane*. To observe an optimal solution at $p = 10$ using *ibm_brisbane* we need $\approx 13.2 \times 10^{-6}/0.0005 = 0.0264s$ while *quantinuum_H2* $\approx 36 \times 10^{-3}/0.02 = 1.8$ s. This means when *quantinuum_H2* finds a solution, *ibm_brisbane* has already found 68.

Discussion

In this work, we have presented numerical and experimental evidence that LR-QAOA constitutes an effective QAOA protocol. This means that this protocol works efficiently for the problem tested, increasing the probability of success as the number of layers increases. We simulate MIS, BPP, TSP, Maxcut, WMaxcut, 3-Maxcut, KP, PO, Max-2-SAT, and Max-3-SAT problems with up to 42 qubits and 400 layers on the modular supercomputer JUWELS. Additionally, we test LR-QAOA using WMaxcut problems from 5 to 109-qubit cases and p from 3 to 100 on real quantum hardware using *ibm_brisbane*, *ibm_osaka*, *ibm_kyoto*, *quantinuum_H2*, and *ionq_aria* finding that LR-QAOA is resilient to noise. We show that this behavior arises from the algorithm's ability to enhance solution quality at a rate that initially outpaces the accumulation of noise. While the overlap of the probability of success decreases exponentially with the number of 2-qubit gates, it is compensated by an exponential growth in the probability of success up to some p. This explains why the highest probability of success does not show up at the smallest number LR-QAOA layers in the experiments but at some other point, e.g., at $p = 10$ in the ibm_brisbane case.

One important conclusion from this work is that one can completely suppress the classical optimization step in QAOA for some COPs. With the fixed schedule in LR-QAOA, one can reduce the set of parameters to tune to only three Δ_β, Δ_γ, and p.

We show the evolution of LR-QAOA from the perspective of the amplitudes of the computational basis states. This change in framework allows us to explain the evolution of the amplitudes under the application of U_C and U_B. Under the application of U_C, each amplitude is rotated proportionally to the state energy. The case of U_B is more complex, but every amplitude evolves with contributions from the other states' amplitudes with the Hamming distance as the indicator of how to group their contribution. The annealing characteristics of LR-QAOA, along with a constant rotation of the states' amplitude (constant slope of the linear ramp) under U_C, allow the exploitation of an interference pattern that enhances the optimal solution in the different COPs.

We observe that the success probability of the optimal solutions using LR-QAOA for the different COPs seems to scale as probability $(x^*) \approx 2^{-\eta(p)N_q+C}$ for $\eta(p)$ decreasing with p and a constant C. We add further evidence to solve fully connected WMaxcut problems. We create 100 random problems and select the 20 of them that require the highest number of iterations for the SA solver. We compare SA, LR-QAOA, TABU, and CPLEX in terms of TTS for these problems, observing a better scaling in LR-QAOA.

We extend the study to Maxcut, Max-2-SAT, and Max-3-SAT with up to 42 qubits. Using $p = N_q$, we find that on average, the probability (x^*) remains nearly constant for WMaxcut, Maxcut, and Max-3-SAT. The Max-2-SAT case is an exception, using $p = N_q$, it shows an exponential decay in the probability of success, still above a quadratic speedup over random guessing. We think this is a consequence of problems with a high concentration of solutions close to the optimal solution.

Moreover, we find that LR-QAOA tolerates noise. This is important as we are at a stage where quantum computers have moderate noise. We simulate a MIS using LR-QAOA with $p = 80$, and at $p = 15$ we add a layer of X gates to see how the algorithm evolves under this noise. We find that even in this scenario, the error does not expand, and the optimal solution can still be found with high probability. We extend the study of noise using depolarizing noise on an FC WMaxcut with different numbers of qubits, layers, and E_d, and find that p_{ovl} decreases exponentially with the number of 2-qubit gates. Using the same model, we fit experimental results in a Quantinuum H1-1 emulator and ibm_fez QPU. In both cases, our model fit well with an apparent error of $\lambda = 2.5 \times 10^{-4}$ for H1-1E and $\lambda = 25 \times 10^{-4}$ for ibm_fez. These errors allow the execution of 733 gates in the case of ibm_fez and 7331 for H1-1E to have an overlap of 10% with the ideal probability.

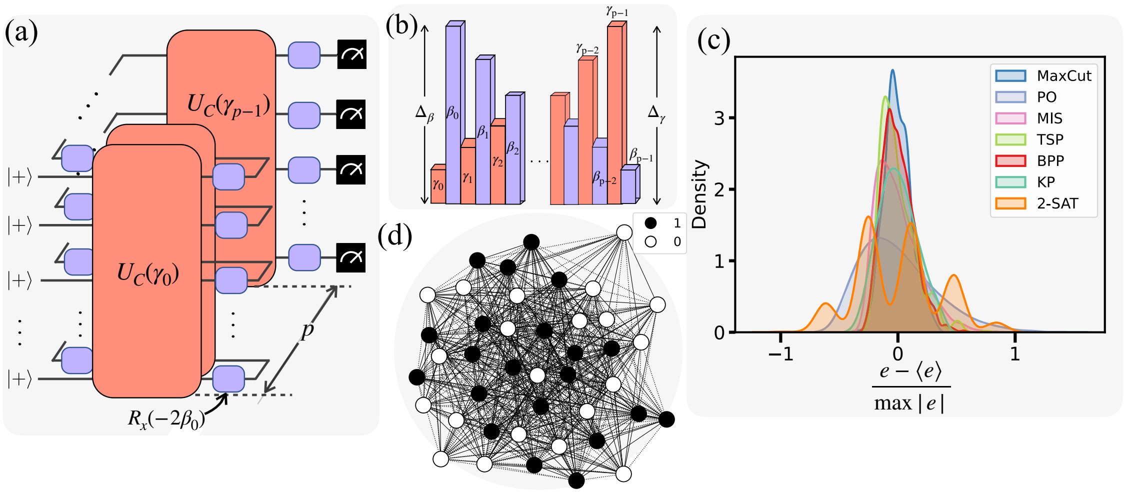

Fig. 7 | The LR-QAOA protocol. **a** Quantum circuit of the QAOA algorithm, **b** LR-QAOA schedule, **c** density vs the normalized eigenvalue distribution for the different COPs with e representing the eigenvalue. All the distributions are for 10-qubit problems except BPP and TSP, with 12-qubit and 9-qubit problems, respectively. **d** An optimal solution for one of the 42-qubit WMaxcut problems using $p = 50$ LR-QAOA. Dashed lines represent cuts, black (white) vertices qubits in 1 (0) state. At the end of the algorithm, the probability of finding the maximum cut is 32%.

We find that there is an effective number of layers for which the real device shows the best performance. We call it the p_{eff}, this parameter can be used to measure the progress of quantum technology for combinatorial optimization. For IBM Eagle devices and *ionq_aria*$p_{\text{eff}} = 10$ and for *quantinuum_H2* is $p_{\text{eff}} = 50$ for a 1D topology problem. We expect the p_{eff} decreases for a fully connected graph problem because the number of two-qubit gates per layer grows by $O(N_q^2)$ compared with the 1D case, $O(N_q)$.

The experimental results make us optimistic that LR-QAOA can keep high performance even in the presence of noise. For example, *quantinuum_H2* already shows its peak performance at $p = 50$ and loses little performance at $p = 100$. At the peak point, the device reaches the best approximation ratio of $r = 0.95$ and probability $(x^*) = 0.08$ for a 25-qubit problem. On the other hand, the inaccuracy of *ibm_brisbane* is still compensated by its sample rate for the same problem.

In a recent study[37], 10 hard COPs for classical solvers were introduced, many of which are sparsely connected; this characteristic can make them suitable to be tackled by LR-QAOA. Between these problems is the MIS, and hard instances show up at sizes with a few hundred qubits. For instance, there is a case with 500 qubits and 6256 edges (see Table 7, R 500 005 1 in ref. 37) for which the optimal solution is not known. For a $p = 200$ LR-QAOA and based in our perceived scale, we need 1'250,200 2-qubit gates, the noise in this case to reach an overlap of 10% is $\lambda = 1.45 \times 10^{-6}$ and based on the scaling of Fig. 1 the number of samples needed is around 25,000 to observed the optimal solution. Currently, the noise in the QPUs is 2 orders of magnitude above the level of error needed, and the number of qubits is at most 156.

It might be possible that the noise level required of a fault-tolerant quantum computer (FTQC). A recent effort to estimate the overhead scenario of FTQC has been presented for the 8-SAT problem[38], finding that at some point, QAOA combined with amplitude amplification with the FTQC overhead can still outperform the best classical solver for that problem.

Methods

In this section, we describe the LR-QAOA, some properties of LR-QAOA, the combinatorial optimization problems used, the classical solvers used to compare scaling properties, and experimental details on real quantum hardware.

LR-QAOA

QAOA consists of alternating layers that encode the problem of interest along with a mixer element in charge of amplifying solutions with low energy. In this case, the COP cost Hamiltonian is given by

$$H_C = \sum_i h_i \sigma_z^i + \sum_{i,j>i} J_{ij} \sigma_z^i \sigma_z^j, \tag{2}$$

where σ_z^i is the Pauli-z term of qubit i, and h_i and J_{ij} are coefficients associated with the problem. Usually, H_C is derived from the quadratic unconstrained binary optimization (QUBO) formulation[2,20,39]. The QUBO to H_C transformation usually includes a constant term that does not affect the QAOA formulation and is left out for simplicity. H_C is encoded into a parametric unitary gate given by

$$U_C(H_C, \gamma_i) = e^{-j\gamma_i H_C}, \tag{3}$$

where γ_i is a parameter that in our case comes from the linear ramp schedule. Following this, in every second part of a layer, a unitary operator is applied, given by

$$U(H_B, \beta_i) = e^{j\beta_i H_B}, \tag{4}$$

where β_i is taken from the linear ramp schedule and $H_B = \sum_{i=0}^{N_q-1} \sigma_i^x$ with σ_i^x the Pauli-x term of qubit *i*. The general QAOA circuit is shown in Fig. 7-(a). Here, $R_X(-2\beta_i) = e^{j\beta_i \sigma^x}$, p is the number of repetitions of the unitary gates of Eqs. (3) and (4), and the initial state is a superposition state $|+\rangle^{\otimes N_q}$. Repeated preparation and measurement of the final QAOA state yields a set of candidate solution samples, which are expected to give the optimal solution or some low-energy solution.

In Fig. 7b, we show the LR-QAOA protocol. It is characterized by three parameters Δ_β, Δ_γ, and the number of layers p. The β_i and γ_i parameters are given by

$$\beta_i = \left(1 - \frac{i}{p}\right)\Delta_\beta \text{ and } \gamma_i = \frac{i+1}{p}\Delta_\gamma, \tag{5}$$

for $i = 0, \ldots, p-1$. For our simulations, we scan over a set of Δ_γ and Δ_β from one problem at each problem size and use the best value over the remaining cases. For the experimental results, we use $\Delta_\beta = 0.3$ and $\Delta_\gamma = 0.6$.

Properties of LR-QAOA

The QAOA evolution is usually presented from the point of view of the expectation value of the cost Hamiltonian[1,21,40]. In this section, we present a

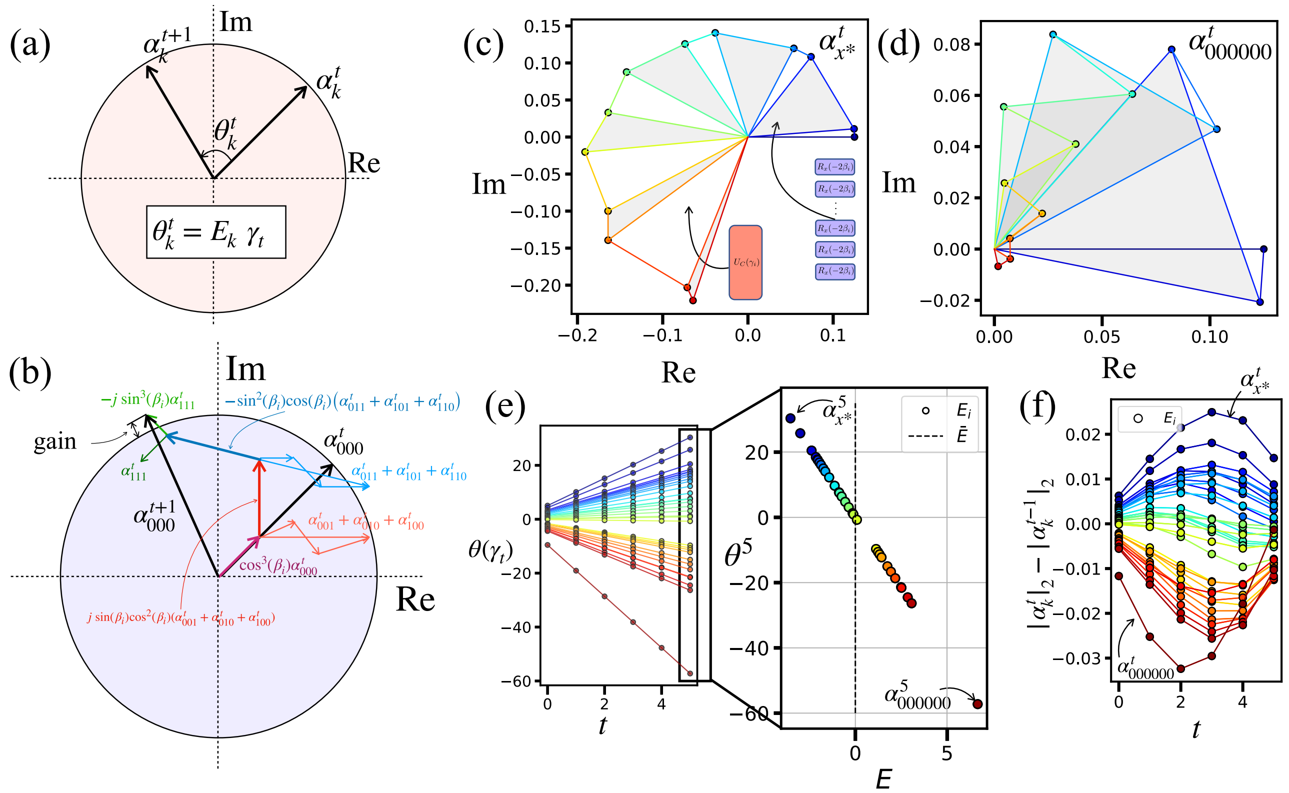

Fig. 8 | LR-QAOA from the point of view of individual time steps. **a** The action of the $U_C(\gamma_t)$ gate on the state k at time step t, **b** evolution of the $|000\rangle$ amplitude after the application of $U_B(\beta_t)$ for a 3-qubit system, **c** evolution of the optimal solution, x^*, in a 6-qubit WMaxcut problem. The gray (white) triangles are a time-step evolution due to $U_B(\gamma_t)$ ($U_C(\gamma_t)$). Line colors represent the time steps being blue (red) $t = 0$ ($t = 5$) step. **d** Evolution of the worst solution for the 6-qubit WMaxcut problem. **e**, left LR-QAOA γ rotations at each layer for each state. Positive angles refer to counterclockwise rotations. Colors represent the energy of the state, with darker blue (red) closer to the optimal (worst) solution of the problem. **e**, Right last layer rotation in LR-QAOA vs. the energy, following Eq (8). **f** Amplitude gain evolution of the states after each $U_B(\beta_t)$ for the 6-qubit WMaxcut problem.

framework where the evolution under LR-QAOA is seen from the point of view of the individual amplitudes of all possible states in a COP. The state vector that describes the evolution of probability (x^*) of a COP is given by

$$|\psi_t\rangle = \sum_{k=0}^{2^{N_q}-1} \alpha_k^t |k\rangle, \tag{6}$$

where t is some step in the QAOA algorithm, k is the state in the computational basis, and α_k^t the amplitude of $|k\rangle$ at time t.

The unitary transformation induced by $U_C(\gamma_t)$, $|\psi_{t+1}\rangle = U_C(\gamma_t)|\psi_t\rangle$, produces a rotation in the complex plane for every state given by

$$\alpha_k^{t+1} = e^{j\theta_k^t}\alpha_k^t, \tag{7}$$

$$\theta_k^t = E_k\gamma_t, \tag{8}$$

where $E_k = \langle k|H_c|k\rangle$. This evolution is shown in Fig. 8a. Eq. (8) explains why the amplitude amplification is proportional to the energy of a given solution. Negative energies are rotated counterclockwise with the rotation proportional to their energies. This can be seen in Fig. 8e.

The change by $U_B(\beta_t)$, $|\psi_{t+1}\rangle = U_B(\beta_t)|\psi_t\rangle$, is more complex and depends on the Hamming distance of the given state to the other states. This operator is responsible for the change in energy and produces an interference pattern that exploits the $U_C(\gamma_t)$ effect. It is described by

$$\alpha_k^{t+1} = \sum_{l=0}^{2^{N_q-1}} \left(\cos(\beta_t)\right)^{N_q - k\cdot l}\left(j\sin(\beta_t)\right)^{k\cdot l}\alpha_l^t, \tag{9}$$

with

$$k\cdot l = \sum_{m=0}^{N_q-1}(k_m \oplus l_m), \tag{10}$$

where k and l are states in the computational basis. Equation (10) gives the Hamming distance between the states k and l. See Supplementary Note 5 for a detailed derivation of Eq. (9). For example, in a 3-qubit system, the evolution of α_{000}^t is given by

$$\begin{aligned}\alpha_{000}^{t+1} &= \langle 000|U_B(\beta_t)|\psi_t\rangle\\ &= \cos^3(\beta_t)\alpha_{000}^t\\ &+j\sin(\beta_t)\cos^2(\beta_t)\left(\alpha_{001}^t+\alpha_{010}^t+\alpha_{100}^t\right)\\ &-\sin^2(\beta_t)\cos(\beta_t)\left(\alpha_{011}^t+\alpha_{101}^t+\alpha_{110}^t\right)\\ &-j\sin^3(\beta_t)\alpha_{111}^t.\end{aligned}$$

A schematic representation of how the $U_B(\beta_t)$ induces an evolution of α_{000} is shown in Fig. 8b. Here, $U_B(\beta_t)$ changes the amplitude and direction of α_{000} using the information of α_{000} and the other states. The Hamming distance indicates how the amplitudes are grouped. For example, the effective vector $r_1 = (\alpha_{001} + \alpha_{010} + \alpha_{100})$ of states with Hamming distance 1, contribute to α_{000} after a $\pi/2$ rotation and a rescaling given by $\sin(\beta_t)\cos^2(\beta_t)$.

In Fig. 8c is shown the evolution of the optimal solution, x^*, of a 6-qubit WMaxcut problem for the $U_C(\gamma_t)$ and $U_B(\beta_t)$ steps for $t \in \{0, \ldots, Nq - 1\}$. In Fig. 8d shows the same evolution but for the state with the lowest energy. In this case, the evolution of the eigenvalue due to

$U_C(\gamma_t)$ goes in the opposite direction to the evolution of $U_B(\beta_t)$ producing the desired effect of interference. Figure 8e shows the angle of rotation of the white triangles, i.e., the rotation due to $\theta(\gamma_t)$. The gain in the amplitude of the α_k^t at each time step after the unitary evolution $U_B(\beta_t)$ is shown in Fig. 8f.

COPs

A detailed description of some COPs used in this work can be found in the appendix of ref. 20, and for the Max-3-SAT is presented in the Supplementary Note 6. For them, we use a normalization technique described in Sec. IV D. We pick 5 random instances for different problem sizes. For the TSP, we use instances with 3, 4, 5, and 6 cities (9, 16, 25, and 36 qubits), where the distances between cities are symmetric and randomly chosen from a uniform distribution with values between 0.1 and 1.1. In the BPP, we consider scenarios involving 3, 4, 5, and 6 items (12, 20, 30, and 42 qubits). The weight of each item is randomly chosen from 1 to 10, and 20 is the maximum weight of the bins. The WMaxcut, 3-Maxcut, MIS, and PO problem sizes are given by the number of qubits and chosen to be 20, 25, 30, 35, and 40.

For WMaxcut problem simulations, we use randomly weighted edges with weights chosen uniformly between 0 and 1 and edge density, $E_d = 0.7$. One of these cases with its optimal solution is presented in Fig. 7d. To test the scaling of LR-QAOA, we use a fully connected random WMaxcut with weights chosen from a uniform discrete distribution from 0 to 1000 in steps of 1. For MIS, edges between nodes are randomly selected with $E_d = 0.4$. For KP problems, item values range from 5 to 63, weights from 1 to 20, and the maximum weight is set to half of the sum of all weights. Finally, for PO, correlation matrix values are chosen from $[-0.1, 0, 0.1, 0.2]$, asset costs varying between 0.5 and 1.5, and the budget is set to half of the total asset cost.

For the inequality constraints in the KP, PO, and BPP, we use the unbalanced penalization approach[39,41]. In this approach, two penalty terms in the QUBO are tuned following the characteristics of the inequality constraints and the objective function. Consequently, any variation in the parameter range necessitates a re-tuning of the penalty terms to maintain performance. For the probability (x^*) using unbalanced penalization, our focus is on finding the ground state of the cost Hamiltonian, since we are interested in knowing the LR-QAOA performance in finding the ground state of the Hamiltonian and there is no guarantee that the optimal solution of the original problem is encoded in the ground state of the Hamiltonian (see also the discussion in ref. 39).

From the problems tested, MIS, BPP, TSP, Maxcut, WMaxcut, KP, PO, and Max-3-SAT are NP-hard[2,25], with varying structural properties and practical solution approaches. Some of them admit effective approximation schemes and are commonly addressed using heuristics or dynamic programming in restricted cases. In particular, MIS and PO have been included in a list of 10 classical hard problems that might benefit from quantum algorithms[37].

We use the probability (x^*) as a metric of the performance for the different COPs. Here, x^* represents the set of optimal bitstrings of the problem's Hamiltonian. Additionally, we use the approximation ratio for the Maxcut and its variations. The approximation ratio is given by

$$r = \frac{\sum_{i=1}^{n} C(x_i)/n}{C(x^*)}, \tag{11}$$

$$C(x) = \sum_{k,l>k}^{N_q} w_{kl}(x_k + x_l - 2x_k x_l), \tag{12}$$

where n is the number of samples, x_i the *ith* bitstring obtained from LR-QAOA, and $C(x)$ is the cost function of WMaxcut, x^* is the optimal bitstring, $C(x^*)$ is the maximum cut, w_{kl} is the weight of the edge between nodes k and l, and x_k is the kth position of the x bitstring.

Figure 7c presents examples of the eigenvalue distribution of the Hamiltonian for different COPs. In the scenario of large-scale problems, the distribution of eigenvalues tends to converge to a normal distribution[42].

Hamiltonian normalization

The Hamiltonian normalization is one important step in LR-QAOA. As we show, every eigenvalue rotates accordingly to Eq. (8), which means that the normalization limits the rotation angle, fixing the *ridge region* to a specific location in the performance diagram[11] (see Supplementary Note 7). The general form of the COP's Ising Hamiltonian is given by

$$H_c(z) = \frac{1}{\max\{|J_{ij}|\}}\left(\sum_{i=0}^{n-1}\sum_{j>i}^{n-1} J_{ij} z_i z_j + \sum_{i=0}^{n-1} h_i z_i + \mathrm{O}\right), \tag{13}$$

where J_{ij} and h_i are real coefficients that represent the COP, and the offset, O, is a constant value. Since O does not affect the location of the optimal solution, it can be left out for the sake of simplicity. There are different ways of normalizing the Hamiltonian, we identify two, normalizing by $\max\{|J_{ij}|\}$ or $\max\{|h_i, J_{ij}|\}$, and use them on each problem. We select the one with the best results in terms of probability (x^*). We find that the $\max\{|J_{ij}|\}$ strategy improves faster the probability (x^*) while $\max\{|h_i, J_{ij}|\}$ improves optimal and suboptimal energies. For the results presented, we choose to normalize the Hamiltonian by $\max\{|J_{ij}|\}\ \forall\, i > j \in 0, .., n-1$ for almost all the cases except MIS where we use $\max\{|h_i|\}\ \forall\, i \in 0, .., n-1$.

Classical solvers

To assess the performance of LR-QAOA, we compare its scalability to simulated annealing (SA)[43], TABU search[44], and CPLEX's spatial B&B[45]. We selected TABU search because it has been shown to outperform other solvers in finding optimal solutions to Maxcut problems[46]. The improved performance in the TABU search can be attributed to a TABU list that prevents revisiting previous solutions and therefore mitigates local minimum problems. We use time-to-solution (TTS) as a metric to compare the resources needed to find the optimal solutions to fully connected WMaxcut. The TTS is given by

$$TTS_{p_d} = T\frac{\ln(1-p_d)}{\ln(1-\mathit{probability}\,(x^*))}, \tag{14}$$

where T is the time needed to get one sample, $p_d = 0.99$ is the target probability, i.e., the confidence level that the optimal solution is sampled at least once with 99% certainty. T in SA and TABU depend on the number of sweeps, with 1 sweep representing a full update cycle over all variables. In experiments, we vary the number of sweeps from 50 to 500.

For SA, we use the `dwave-neal`[47] and for TABU we use `dwave-tabu`[48], both performant C++-based libraries that use Python as an interface. In the case of the CPLEX solver, we use `docplex`[49] Python interface of CPLEX. All the algorithms run on a MacBook Pro equipped with an Apple M1 chip. The code used to run the given cases can be found at[50].

The case of LR-QAOA, the $T = t_{2q}(2N_q + 2)p$ with t_{2q} is the 2-qubit gate time, and the time to execute one layer of QAOA scales as $O(2N_q + 2)$ based on a flexible scheme that can be run in a 1D chain of qubits[31]. The t_g for most superconducting-based QPUs is on the order of nanoseconds.

Noise model

At the instruction level, the main source of noise in digital quantum computers comes from the 2-qubit entangling gates[51]. Thus, we use a depolarizing noise channel in the 2-qubit gates of the LR-QAOA protocol. This channel is given by

$$\mathcal{E}[\rho] = (1-\lambda)\rho + \lambda\frac{I}{4}, \tag{15}$$

where λ is the depolarizing error parameter, I is a 4×4 identity matrix, and ρ is the density matrix of the 2-qubit system. In general, the action of a 2-qubit gate on a general density matrix can be expressed by

$$\mathcal{E}_{ij}[\rho] = (1-\lambda)U^{ij}_{2Q}\rho U^{ij}_{2Q} + \frac{\lambda}{4}\mathrm{Tr}_{ij}(\rho)\otimes I, \tag{16}$$

where $\mathcal{E}_{ij}$ is the channel acting on ρ, Tr_{ij} is the partial trace over qubits i and j, and U^{ij}_{2Q} is the 2-qubit unitary gate. For simplicity, we assume λ is the same for all the 2-qubit gates.

To test how noise affects the LR-QAOA solution for a given problem, we use the following relation,

$$p_{ovl} = \frac{probability\,(x^*)^{QPU} - probability\,(x^*)^{r}}{probability\,(x^*) - probability\,(x^*)^{r}}, \tag{17}$$

where p_{ovl} is the overlap between the ideal success probability probability (x^*) and the one obtained in the real device, probability $(x^*)^{\mathrm{QPU}}$, normalized by the random sampler success probability, probability $(x^*)^r$. Additionally, we define the accumulated error in the circuit using

$$\varepsilon_{acc} = N_g\lambda \tag{18}$$

where N_g is the number of 2-qubit gates involved in the circuit. Using this relation, we find that a model that describes p_{ovl} is

$$p_{ovl} = 2^{-k_0\varepsilon_{acc}}, \tag{19}$$

where k_0 is a fitting parameter that depends on the problem. In results, we show that this approach can be applied to superconductive and trapped ion-based QPUs, obtaining a good match of experimental results for both devices.

Mitigation: Hamming distance 1

In ref. 20, we introduce the mitigation technique used here. This involves applying a bitflip to each position within the output bitstring of samples from a quantum computer, to mitigate single-qubit bitflips errors. The computational overhead of this postprocessing method is $O(NN_q)$, where N represents the number of samples and N_q is the number of qubits. While this mitigation technique can correct errors coming from the readout of the quantum device, it is also an optimization step that can completely obscure the optimization coming from the LR-QAOA algorithm. Therefore, it is important to compare the results against those obtained from a random sampler using the same mitigation technique, which is included in our main results. The details of our proposed approach are described in Algorithm 1.

Algorithm 1. **Sampler mitigation**

```
Data: bitstring samples S = [$s_0$, ..., $s_{N-1}$]
Result: Samples corrected $S_{mitig}$
Initialization;
for $i$=0; $i$++; $i < N$ do
    $E_{best}$ = Energy($S[i]$);
    $s_{best}$ = $S[i]$;
    for $j$=0; $j$++; $j < N_q$ do
        $s_{new}$ = $S[i]$;
        if $s_{new}[j]$ == 1 then
            $s_{new}$[j] = 0
        else
            $s_{new}$[j] = 1
        $E_{new}$ = Energy($s_{new}$);
        if $E_{new} < E_{best}$ then
            $E_{best}$ = $E_{new}$;
            $s_{best}$ = $s_{new}$;
        $S_{mitig} \leftarrow s_{best}$
return $S_{mitig}$;
```

Experimental details

We use random fully connected WMaxcut from 5 to 15 qubits. We run experiments on ibm_fez and H1-1E. For the case of ibm_fez, we use the parity twine chains (PTC)[30,31] strategy to encode the LR-QAOA quantum circuit into a 1D-chain of qubits of the QPU. H1-1E is a 20-qubit emulator of the Quantinuum H1-1 QPU. In these experiments, $\Delta_\gamma = \Delta_\beta = 0.6$, the number of samples is 1000.

We implement WMaxcut problems using LR-QAOA with $\Delta_\gamma = 0.6$ and $\Delta_\beta = 0.3$ on three quantum computing technologies: IonQ Aria a fully connected 25-qubit device based on trapped ions with 2-qubit gate error of 0.4% and 2-qubit gate speed of $t_{2q} = 600$ μs[52], labeled *ionq_aria*, Quantinuum H2-1 (a fully connected 32-qubit device based on trapped ions with a 2-qubit error rate of 0.2%[28], labeled *quantinuum_H2*), and three IBM Eagle superconducting processors[53], 127 transmon qubits with heavy-hex connectivity and 2-qubit median gate error between 0.74 and 0.95%, error per layered gate (EPLG)[54] between 1.9% and 3.6%, and 2-qubit gate speed of $t_{2q} = 0.66$ μs, labeled *ibm_brisbane*, *ibm_kyoto* and *ibm_osaka*).

We perform different experiments to assess the practical performance of quantum technology to solve COPs using LR-QAOA. First, an experiment on *ionq_aria* for a 10-qubit WMaxcut with 70% of random connections as described in Section IV C, this helps for the sake of comparison with a depolarizing noise model. Additionally, different problems from 5 to 109 qubits were tested on *ibm_brisbane* using a WMaxcut problem with a 1D-chain topology shown in Fig. 6a. We opt for a simple graph due to constraints posed by noise. Additionally, we provide an experimental comparison across three distinct IBM devices for a 109-qubit WMaxcut problem. Finally, a comparison between *ionq_aria*, *ibm_brisbane*, and *quantinuum_H2* is shown for a 25-qubit WMaxcut problem, Fig. 6b.

The time of execution t_e for the 1D chain topology LR-QAOA protocol can be approximated to that of the 2-qubit gates. This is because single-qubit operations are a minority and their execution time is generally faster than 2-qubit gates. In *ionq_aria* the 2-qubit gates are executed sequentially so $t_e = t_{2q}N_{2q}p$ where N_{2q} is the number of 2-qubit terms in the cost Hamiltonian. For the case of *ibm_brisbane*, the time of execution is $t_e = 2t_{2q}p$. *quantinuum_H2* can execute 4 2-qubit gates in parallel, hence, the execution time is $t_e = t_{2q}(N_{2q}/4)p$. The time per 2-qubit gate is 600μs on *ionq_aria*, and 660ns on *ibm_brisbane*. We could not find information about t_{2q} for *quantinuum_H2*, but we assume it is similar to *ionq_aria*. Therefore, a 25-qubit WMaxcut with 1D topology requires 14.4 ms, 3.6 ms, and 1.32 μs for each layer using *ionq_aria*, *quantinuum_H2*, and *ibm_brisbane*, respectively. For each experiment on IBM devices, *ionq_aria*, and *quantinuum_H2*, we use 10000, 1000, and 50 samples, respectively.

Data availability

All study data are included in this article and the Supplementary Materials. The datasets for the problems used and/or analyzed during the current study are available from the following publicly accessible repository https://github.com/alejomonbar/LR-QAOA.

Code availability

The code is available from the following publicly accessible repository https://github.com/alejomonbar/LR-QAOA.

Received: 28 April 2025; Accepted: 22 July 2025;
Published online: 04 August 2025

References

1. Farhi, E., Goldstone, J. & Gutmann, S. A quantum approximate optimization algorithm. Preprint at *arXiv* https://doi.org/10.48550/arXiv.1411.4028 (2014).
2. Lucas, A. Ising formulations of many NP problems. *Front. Phys.* **2**, 1–14 (2014).
3. Kochenberger, G. et al. The unconstrained binary quadratic programming problem: a survey. *J. Comb. Optim.* **28**, 58–81 (2014).
4. Farhi, E., Goldstone, J., Gutmann, S. & Sipser, M. Quantum computation by adiabatic evolution. Preprint at *arXiv* https://arxiv.org/abs/quant-ph/0001106 (2000).
5. Harrigan, M. P. et al. Quantum approximate optimization of non-planar graph problems on a planar superconducting processor. *Nat. Phys.* **17**, 332–336 (2021).
6. Niroula, P. et al. Constrained quantum optimization for extractive summarization on a trapped-ion quantum computer. *Sci. Rep.* https://doi.org/10.1038/s41598-022-20853-w (2022).
7. Shaydulin, R., Lotshaw, P. C., Larson, J., Ostrowski, J. & Humble, T. S. Parameter transfer for quantum approximate optimization of weighted MaxCut. *ACM Trans. Quantum Comput.* **4**, 1–15 (2023).
8. Ohzeki, M. Breaking limitation of quantum annealer in solving optimization problems under constraints. *Sci. Rep.* **10**, 1–12 (2020).
9. Cerezo, M. et al. Variational quantum algorithms. *Nat. Rev. Phys.* **3**, 625–644 (2021).
10. Bittel, L. & Kliesch, M. Training variational quantum algorithms Is NP-Hard. *Phys. Rev. Lett.* **127**, 120502 (2021).
11. Kremenetski, V., Hogg, T., Hadfield, S., Cotton, S. J. & Tubman, N. M. Quantum alternating operator ansatz (qaoa) phase diagrams and applications for quantum chemistry. Preprint at *arXiv* https://doi.org/10.48550/arXiv.2108.13056 (2021).
12. Larocca, M. et al. Diagnosing barren plateaus with tools from quantum optimal control. *Quantum* **6**, 824 (2022).
13. Koßmann, G., Binkowski, L., van Luijk, L., Ziegler, T. & Schwonnek, R. Deep-circuit qaoa. Preprint at *arXiv* https://doi.org/10.48550/arXiv.2210.12406 (2022).
14. Zhou, L., Wang, S. T., Choi, S., Pichler, H. & Lukin, M. D. Quantum approximate optimization algorithm: performance, mechanism, and implementation on near-term devices. *Phys. Rev. X* **10**, 1–23 (2020).
15. Apolloni, B., Carvalho, C. & de Falco, D. Quantum stochastic optimization. *Stoch. Process. Appl.* **33**, 233–244 (1989).
16. De Falco, D. & Tamascelli, D. An introduction to quantum annealing. *RAIRO Theor. Inform. Appl.* **45**, 99–116 (2011).
17. Brandao, F. G. S. L., Broughton, M., Farhi, E., Gutmann, S. & Neven, H. For fixed control parameters the quantum approximate optimization algorithm's objective function value concentrates for typical instances. Preprint at *arXiv* https://doi.org/10.48550/arXiv.1812.04170 (2018).
18. Willsch, D., Willsch, M., Jin, F., Michielsen, K. & De Raedt, H. Gpu-accelerated simulations of quantum annealing and the quantum approximate optimization algorithm. *Comput. Phys. Commun.* **278**, 108411 (2022).
19. Hess, M., Palackal, L., Awasthi, A. & Wintersperger, K. Effective Embedding of Integer Linear Inequalities for Variational Quantum Algorithms. *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Montreal, QC, Canada (2024).
20. Montañez-Barrera, J. A., Willsch, D. & Michielsen, K. Transfer learning of optimal QAOA parameters in combinatorial optimization. *Quantum Inf. Process* **24**, 129 (2025).
21. Kremenetski, V., Apte, A., Hogg, T., Hadfield, S. & Tubman, N. M. Quantum alternating operator ansatz (qaoa) beyond low depth with gradually changing unitaries. Preprint at *arXiv* https://doi.org/10.48550/arXiv.2305.04455 (2023).
22. Krause, D. JUWELS: Modular Tier-0/1 supercomputer at the Jülich supercomputing centre. *J. Large-Scale Res. Facil.* **5**, A135 (2019).
23. Alvarez, D. JUWELS cluster and booster: exascale pathfinder with modular supercomputing architecture at Juelich supercomputing centre. *J. Large-Scale Res. Facil.* **7**, A183 (2021).
24. Gutin, G. & Yeo, A. Lower bounds for maximum weighted cut. *SIAM J. Discret. Math.* **37**, 1142–1161 (2023).
25. Paradimitriou, C. & Yannakakis, M. Optimization, approximation, and complexity classes. *J. Comput. Syst. Sci.* **43**, 425–440 (1991).
26. Shaydulin, R. et al. Evidence of scaling advantage for the quantum approximate optimization algorithm on a classically intractable problem. *Sci. Adv.* https://doi.org/10.1126/sciadv.adm6761 (2024).
27. Boulebnane, S. & Montanaro, A. Solving boolean satisfiability problems with the quantum approximate optimization algorithm. *PRX Quantum* **5**, 030348 (2024).
28. Moses, S. A. et al. A race-track trapped-ion quantum processor. *Phys. Rev. X* **13**, 41052 (2023).
29. Kirch, W. (ed). *Pearson's Correlation Coefficient*, 1090–1091 (Springer Netherlands, Dordrecht, 2008). https://doi.org/10.1007/978-1-4020-5614-7_2569.
30. Dreier, F. et al. Connectivity-aware synthesis of quantum algorithms. Preprint at *arXiv* https://doi.org/10.48550/arXiv.2501.14020 (2025).
31. Klaver, B. et al. Swap-less implementation of quantum algorithms. Preprint at *arXiv* https://arxiv.org/abs/2408.10907 (2024). 2408.10907.
32. Knill, E. et al. Randomized benchmarking of quantum gates. *Phys. Rev.* https://doi.org/10.1103/PhysRevA.77.012307 (2008).
33. Arute, F. et al. Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505–510 (2019).
34. Kim, Y. et al. Evidence for the utility of quantum computing before fault tolerance. *Nature* **618**, 500–505 (2023).
35. Pednault, E. et al. Pareto-efficient quantum circuit simulation using tensor contraction deferral. Preprint at *arXiv* https://doi.org/10.48550/arXiv.1710.05867 (2017).
36. Begušić, T., Gray, J. & Chan, G. K. L. Fast and converged classical simulations of evidence for the utility of quantum computing before fault tolerance. *Sci. Adv.* **10**, 1–4 (2024).
37. Koch, T. et al. Quantum optimization benchmark library—the intractable decathlon. Preprint at *arXiv* https://arxiv.org/abs/2504.03832 (2025).
38. Omanakuttan, S. et al. Threshold for fault-tolerant quantum advantage with the quantum approximate optimization algorithm. Preprint at *arXiv* https://doi.org/10.48550/arXiv.2504.01897 (2025).
39. Montanez-Barrera, A., Willsch, D., Maldonado-Romo, A. & Michielsen, K. Unbalanced penalization: a new approach to encode inequality constraints of combinatorial problems for quantum optimization algorithms. *Quantum Sci. Technol.* https://doi.org/10.1088/2058-9565/ad35e4 (2024).
40. Hadfield, S., Hogg, T. & Rieffel, E. G. Analytical framework for quantum alternating operator ansätze. *Quantum Sci. Technol.* **8**, 1–65 (2023).
41. Montanez-Barrera, J. A., van den Heuvel, P., Willsch, D. & Michielsen, K. Improving performance in combinatorial optimization problems with inequality constraints: an evaluation of the unbalanced

penalization method on D-wave advantage. *2023 IEEE Int. Conf. Quantum Comput. Eng. (QCE)* **01**, 535–542 (2023).

42. Wald, A. & Wolfowitz, J. Statistical tests based on permutations of the observations. *Ann. Math. Stat.* **15**, 358–372 (1944).
43. Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. Optimization by simulated annealing. *Science* **220**, 671–680 (1983).
44. Palubeckis, G. Multistart tabu search strategies for the unconstrained binary quadratic optimization problem. *Ann. Oper. Res.* **131**, 259–282 (2004).
45. Bliek, C., Bonami, P. & Lodi, A. Solving mixed-integer quadratic programming problems with ibm-cplex: a progress report https://api.semanticscholar.org/CorpusID:16208906 (2014).
46. Dunning, I., Gupta, S. & Silberholz, J. What works best when? A systematic evaluation of heuristics for Max-Cut and QUBO. *INFORMS J. Comput.* **30**, 608–624 (2018).
47. D-Wave Systems. Simulated annealing sampler—dwave-neal 0.5.9 documentation. https://docs.ocean.dwavesys.com/projects/neal/en/latest/reference/sampler.html (2023).
48. D-Wave Systems. D-wave tabu—d-wave tabu 0.4.2 documentation. https://docs.ocean.dwavesys.com/projects/tabu/en/latest/ (2021).
49. IBM Decision Optimization on Cloud team. DOcplex: IBM Decision Optimization CPLEX Modeling for Python. https://pypi.org/project/docplex/ (2024).
50. Montañez-Barrera, A. Lr-qaoa: fixed linear ramp schedules in qaoa. https://github.com/alejomonbar/LR-QAOA (2024). Accessed: 2025-04-25.
51. Pascuzzi, V. R., He, A., Bauer, C. W., de Jong, W. A. & Nachman, B. Computationally efficient zero-noise extrapolation for quantum-gate-error mitigation. *Phys. Rev. A* **105**, 042406 (2022).
52. IonQ Aria Quantum System. https://ionq.com/quantum-systems/aria.
53. IBM Quantum Blog. Eagle Quantum Processor. https://www.ibm.com/quantum/blog/eagle-quantum-processor-performance (2022).
54. McKay, D. C. et al. Benchmarking quantum processor performance at scale. Preprint at *arXiv* https://doi.org/10.48550/arXiv.2311.05933 (2023).

Acknowledgements

The authors thank Dennis Willsch, Vrinda Mehta, Hans De Raedt, and Fengping Jin for the insightful discussions and suggestions made for the present work. J. A. Montanez-Barrera acknowledges support from the German Federal Ministry of Education and Research (BMBF), funding program Quantum Technologies—from basic research to market, project QSolid (Grant No. 13N16149).The authors gratefully acknowledge the Gauss Center for Supercomputing e.V. (https://www.gauss-center.eu/) for funding this project by providing computing time on the GCS Supercomputer JUWELS at Jülich Supercomputing Center (JSC).This research used resources of the Oak Ridge Leadership Computing Facility for the experiments on quantinuum_H2, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725.

Author contributions

J.A.M.B. wrote the main paper, J.A.M.B. prepared all the figures, and K.M. provided guidance during the project. All the authors reviewed the paper.

Funding

Open Access funding enabled and organized by Projekt DEAL.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41534-025-01082-1.

Correspondence and requests for materials should be addressed to J. A. Montañez-Barrera.