

Designing lattice proteins with variational quantum algorithms

Hanna Linn^{1,*}, Lucas Knuthson², Anders Irbäck², Sandipan Mohanty³,
Laura García-Álvarez¹, and Göran Johansson¹

¹*Department of Microtechnology and Nanoscience (MC2), Chalmers University of Technology,
SE-412 96 Göteborg, Sweden*

²*Computational Science for Health and Environment (COSHE), Lund University,
SE-223 62 Lund, Sweden*

³*Jülich Supercomputing Centre, Forschungszentrum Jülich, D-52425 Jülich, Germany*



(Received 6 August 2025; revised 16 April 2026; accepted 28 May 2026; published 28 September 2026)

Quantum heuristics have shown promise in solving various optimization problems, including lattice protein folding. Equally relevant is the inverse problem, protein design, where one seeks sequences that fold to a given target structure. The latter problem is often split into two steps: (i) searching for sequences that minimize the energy in the target structure, and (ii) testing whether the generated sequences fold to the desired structure. Here, we investigate the utility of variational quantum algorithms for the first of these two steps on today's noisy intermediate-scale quantum devices. We focus on the sequence optimization task, which is less resource-demanding than folding computations. We test the quantum approximate optimization algorithm and variants of it, with problem-informed quantum circuits, as well as the hardware-efficient ansatz, with problem-agnostic quantum circuits. While the former approach, with a careful choice of mixer, yields good results in noiseless simulations (success probability ≥ 0.95 in all instances), its performance drops drastically under noise. With the problem-agnostic circuits, which are more compatible with hardware constraints, improved performance is observed in noisy simulations, compared to their problem-informed counterparts. When running the problem-agnostic circuits on a real quantum device, with parameters taken from simulations, we obtain a significant success probability (≥ 0.2) in most instances.

DOI: [10.1103/kpf7-fx7t](https://doi.org/10.1103/kpf7-fx7t)

I. INTRODUCTION

The field of quantum computing is undergoing rapid change, with significant advances made in recent years [1–4]. Intense efforts are being directed toward building both analog quantum annealers and gate-based quantum hardware, with gate-based systems as the dominant model, because of its flexibility and universality. As gate-based quantum hardware continues to evolve, variational quantum algorithms (VQAs) are gaining traction in fields ranging from materials science to machine learning [5]. Algorithms such as the quantum approximate optimization algorithm (QAOA) and its variants [6,7], the Hamiltonian variational ansatz [8], and the hardware-efficient ansatz (HEA) [9] have emerged as strong candidates for solving discrete optimization problems on noisy intermediate-scale quantum

(NISQ) devices [10]. Despite the potential of quantum computing, real quantum devices are currently limited by noise and imperfections. Therefore, to make the most of the available quantum resources, these algorithms leverage parameterized quantum circuits, with parameters optimized iteratively by classical computations, to find the solution. This hybrid quantum-classical approach is key for the feasibility of these algorithms on NISQ devices; however, their practical success depends heavily on effective initialization heuristics, robust parameter optimization, and strategies for mitigating noise-induced errors [11].

Among these challenges, parameter optimization stands out as a critical bottleneck. Variational circuits often suffer from barren plateaus and poor convergence in high-dimensional parameter landscapes [11]. To address this hurdle, transfer learning techniques—such as parameter donation between related problem instances or distributed initialization strategies—have been proposed to accelerate convergence and enhance robustness. In particular, parameter donation has shown promise in QAOA [12–15] where reusing optimized parameters from smaller or similar instances can guide the optimization process more effectively. These strategies are especially relevant in the context of current quantum hardware, where limited

*Contact author: hannlinn@chalmers.se

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI. Funded by [Bibsam](https://www.bibsam.com/).

coherence times and gate fidelities impose strict constraints on circuit depth and runtime [10,16].

A biophysically relevant optimization problem that has been explored using digital VQAs [17–20] as well as analog quantum annealing [21–24] is the folding of lattice proteins, where the task is to find the minimum energy structure(s) for a given amino acid sequence. In Ref. [23], the authors found that a field-like representation in conjunction with the ready availability of a large number of qubits in D-Wave quantum annealers allows hybrid quantum-classical sampling to compete favorably with established classical methods. However, the number of qubits and gates required to implement digital quantum approaches to this problem makes their implementation on current NISQ devices impractical beyond proof-of-concept problem sizes [25]. Another important biophysical challenge is the inverse problem [26–28], known as protein design, where one looks for sequences that fold into a given target structure. Recent years have seen great advances in protein design methods, in part based on machine learning techniques [29,30]. However, computational analysis of the biophysics of protein design remains a challenge. The possibility of using quantum optimization to speed up such computations has recently been addressed, focusing on quantum annealing [31–33] and Grover’s algorithm [34].

In this paper, we explore the utility of VQAs for the design of lattice proteins, through classical simulations of quantum circuits and, in selected cases, tests on quantum hardware. We are not aware of any previous work that explores the use of such quantum circuits for the protein design problem. Specifically, we consider the problem of determining amino acid sequences that minimize the energy in a given target structure, using the 2D hydrophobic/polar (HP) model of Lau and Dill [35,36] as a test bed. In this problem, the degrees of freedom are types rather than positions of the amino acids, which makes this task less resource-demanding than the folding problem. However, whether or not the generated sequences actually fold to the desired structure generally needs to be checked. The problem instances examined in this article were chosen so that this verification step is not required. Using exact results available for HP chains with lengths $N \leq 30$ [37,38], we choose instances such that the sequence optimization problem has a unique solution that is also known to fold to the target structure. Our choice of simplified yet nontrivial problems with *a priori* known exact solutions helps us evaluate the effectiveness of nascent computational techniques such as VQAs and identify the inherent challenges impeding their wider applicability.

To this end, we test and compare two types of VQAs: problem-informed QAOA variants, which incorporate the problem structure into the quantum circuit through parametrized gates derived from the objective function; and the

problem-agnostic HEA approach, for which the quantum circuit structure is problem-independent and tailored to the hardware capabilities. That is, the quantum circuit is formulated using the gate set and connectivity of the device at hand, to minimize circuit depth. While the best of the QAOA variants performs well in noise-free simulations (probability ≥ 0.95 to find the ground state), all variants tested prove impractical when adding noise due to the substantial circuit depths demanded. In contrast, when using HEA, our problems could also be solved with noise, especially when donating parameters between similar problem instances. When running on the IBM Torino quantum device, using predetermined parameters taken from the simulations, two-layer HEA had a success probability ≥ 0.2 for most instances studied, with N up to 29.

The structure of this paper is as follows. Section II reviews the formulation of the sequence optimization problem for quantum computing devices and details our methodology, including quantum circuit design and classical optimization strategies. Section III presents our numerical and experimental results, which are further discussed in Sec. IV. We conclude with a discussion and outlook in Sec. V.

II. METHODS

A. Protein design theory

In protein design, an amino acid sequence $s = (s_1, \dots, s_N)$ is sought that folds into a given target structure C_t . To this end, the goal is to maximize the probability of finding the chain in the state C_t , given by

$$P_\beta(s) = e^{-\beta E(C_t, s)} / \sum_C e^{-\beta E(C, s)}, \quad (1)$$

where $E(C_t, s)$ is the energy of the sequence s in conformation C_t , β is the inverse temperature, and the sum runs over all possible structures C . Although methods for this task have been developed [39,40], maximizing $P_\beta(s)$ involves a generally time-consuming search in both sequence and structure spaces. Therefore, a common approach is to ignore the s dependence of the sum in Eq. (1) and thus minimize $E(C_t, s)$ [31,32], or introduce an approximation $e^{-\beta \tilde{F}(s)}$ of the sum and minimize $E(C_t, s) - \tilde{F}(s)$ [33]. In either case, an additional filtering step is needed to reject candidate sequences that have a higher probability for some other structure $C \neq C_t$.

In this work, we focus on the problem of minimizing $E(C_t, s)$ over s , for brevity referred to as sequence optimization. In all the instances studied below, the sequences generated in this way fold to the desired structure C_t [37,38].

B. HP lattice proteins

We consider the minimal two-dimensional lattice-based HP model of proteins [35], in which the protein is

represented by a self-avoiding chain of N hydrophobic (H) or polar (P) beads that interact through a pairwise contact potential. A contact between two beads is said to occur if they are nearest neighbors on the lattice but not along the chain. The energy function is defined as $E_{\text{HP}} = -N_{\text{HH}}$, with N_{HH} the number of HH contacts [35]. This definition renders the formation of a hydrophobic core energetically favorable.

The ground state, i.e., the state of minimum energy, may be degenerate or unique. On a two-dimensional square lattice, it is known from exhaustive enumerations that about 2% of all HP sequences with length $N \leq 30$ have a unique ground state [37,38]. A sequence whose ground state is unique is said to design that structure. The designability of a structure is the number of sequences that design it. High designability implies mutation-tolerance and is a characteristic of real proteins. In our computations, we focus on one target structure for each chain length N , chosen as the most designable structure for that N .

Despite their simplicity, coarse-grained HP models are still relevant for qualitative insights into computationally challenging problems like liquid-liquid phase separation of intrinsically disordered proteins [41,42] and protein evolution modeling [43,44].

C. HP sequence optimization in QUBO form

Given a target structure C_t , we want to find sequences s that minimize the energy $E_{\text{HP}}(C_t, s)$, using variational quantum circuits. To this end, we recast the problem in quadratic unconstrained binary optimization (QUBO) form. Furthermore, we introduce a penalty term to control the total number of H beads, N_{H} ; since an unbiased minimization of $E_{\text{HP}}(C_t, s)$ has the homopolymer sequence of all H as a trivial solution.

As in any biophysical model based on pairwise contact interactions, the only structural information required to calculate $E_{\text{HP}}(C_t, s)$ is the contact matrix w_{ij} , which indicates whether two arbitrary beads i and j are in contact ($w_{ij} = 1$) or not ($w_{ij} = 0$). When using the HP model, a suitable choice of total energy $E(s)$ to minimize is given by

$$E(s) = - \sum_{1 \leq i < j \leq N} w_{ij} s_i s_j + \lambda \left(\sum_{i=1}^N s_i - N_{\text{H}} \right)^2, \quad (2)$$

where s_i describes whether bead i is of type P ($s_i = 0$) or H ($s_i = 1$). In Eq. (2), the first term represents the HP interaction energy $E_{\text{HP}}(C_t, s)$, while the second term biases the total number of H-type beads toward a preset value, N_{H} . The balance between the two terms is set by the Lagrange multiplier λ . Figure 1 shows an example of the sequence optimization problem for $(N, N_{\text{H}}) = (10, 4)$. Minimizing $E(s)$ in Eq. (2) can be seen as a graph bisection problem. It is a fully connected problem, but

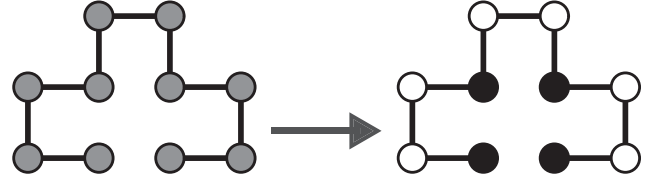


FIG. 1. Illustration of HP sequence optimization for $N = 10$. Given a structure (left) and an N_{H} value, in this example 4, the task is to minimize the energy $E(s)$ in Eq. (2). The solution is a sequence of H (●) and P (○) beads (right).

since the structure remains fixed for the sequence optimization problem, its quantum formulation requires many fewer qubits than the HP folding problem [23,25] since only the bead types—not their location—need to be encoded.

The parameter λ must be large enough for the generated sequences to acquire the desired composition, as set by N_{H} . Once above this threshold, the method's performance becomes robust to small changes in λ [32]. In the calculations presented below, we used $\lambda = 1.1$, which worked well for all instances studied.

Below, we select one N_{H} value for each target structure used. For all problem instances studied, it is possible to infer the minimum HP energy E_{HP} by inspecting the bead-bead contacts in the target structure. The specific N_{H} values used and the known minimum E_{HP} values can be found in Appendix A.

D. Problem-informed quantum circuits: QAOA

In the standard QAOA, the quantum circuit $U(\theta)$ consists of an alternating sequence of p mixer and p problem unitaries with the respective forms $U_M(\beta) = e^{-i\beta M}$ and $U_C(\gamma) = e^{-i\gamma C}$. Here, C encodes the cost function to be optimized, and the mixer Hamiltonian M is a sum of Pauli-X matrices. Given an initial state $|\psi_0\rangle$, an approximate solution to the optimization problem is generated by maximizing $\langle \psi_0 | U^\dagger(\theta) C U(\theta) | \psi_0 \rangle$ over the variational parameters $\theta = (\beta_1, \dots, \beta_p, \gamma_1, \dots, \gamma_p)$. For large p and suitably chosen parameters θ , QAOA can be seen as a discrete version of the analog quantum annealing method [45,46], which has been used to tackle a variety of optimization problems, including protein design [31–33].

In the standard algorithm, constraints are enforced by adding soft penalty terms to the cost Hamiltonian C . In some cases, it is possible to restrict the quantum evolution of the system such that one or more penalty terms can be dropped. This variant, called the quantum alternating operator ansatz [7], requires that the initial state, $|\psi_0\rangle$, belongs to the subspace of feasible states, and that $U_M(\beta)$ does not generate transitions from feasible to unfeasible states.

TABLE I. The QAOA variants studied. We consider two XY-mixers with which the qubits are either fully connected (XY-FC) or connected in a ring (XY-ring). This initial state can be a uniform superposition of all computational basis states (UI), a uniform superposition of all such states with the desired Hamming weight N_H (Dicke state; DI), or a single such state with Hamming weight N_H ($|1 \dots 10 \dots 0\rangle$; BI).

#	Mixer, M	Initial state, $ \psi_0\rangle$
I	X	UI
II	XY-FC	BI
III	XY-FC	DI
IV	XY-ring	BI
V	XY-ring	DI

To convert Eq. (2) to a cost Hamiltonian C that can be implemented on a quantum computer, s_i is replaced by $(I_i - Z_i)/2$, where I_i is the identity. The state of qubit i therefore decides the letter (H or P) of amino acid i . As such, for a chain with N amino acids, only N qubits are needed.

The problem that we wish to solve involves the constraint that the total Hamming weight of the bitstring $\mathbf{s}$ —the number of ones in the bitstring—should equal the preset composition parameter N_H , see Eq. (2). In this case, the penalty term becomes unnecessary if we use XY-mixers [7,47,48]. Specifically, we consider two different XY-mixers, namely a fully connected one (called XY-FC below) given by $\sum_{i < j} (X_i X_j + Y_i Y_j)$ and one (called XY-ring) where the qubits are connected in a ring, $\sum_{i=1}^N (X_i X_{i+1} + Y_i Y_{i+1})$, where $X_{N+1} = X_1$ and $Y_{N+1} = Y_1$. For each of the two XY-mixers, we explore two choices of the initial state $|\psi_0\rangle$. The first is to pick some computational basis state with the desired Hamming weight N_H ; we choose the one where the first N_H bits are set to one (called BI). The second choice considered is a uniform superposition of all computational basis states with the desired Hamming weight (called DI), which is often referred to as a Dicke state [49]. To prepare the Dicke states, we follow the method given in Ref. [49], for which the circuit depth scales as $\mathcal{O}(N)$ (but with $\mathcal{O}(N_H N)$ gates).

In the following, we investigate and compare the performance of these four variants of QAOA, all using XY-mixers, and standard QAOA, with its X-mixer. In the latter method, the initial state $|\psi_0\rangle$ is a uniform superposition of all computational basis states (called UI). A summary of the QAOA variants studied can be found in Table I.

E. Problem-agnostic quantum circuits: HEA

With some QAOAs, it is possible to restrict the search space to feasible solutions, but this comes at the cost of an increased circuit depth (see Sec. III A). Such a depth requirement poses a challenge for near-term quantum

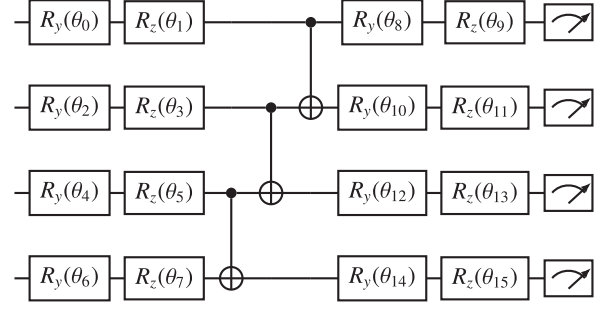


FIG. 2. The single-layer hardware-efficient SU(2) 2-local quantum circuit used in our HEA computations, for $N = 4$ qubits. It features an entangling layer of CNOT gates in a reverse linear pattern, sandwiched between two blocks of parameterized single-qubit R_y and R_z rotations. Final measurements are performed in the computational basis.

devices. Even standard QAOA with its simple X-mixer can reach circuit depths that are difficult to optimize on current quantum hardware.

To address this issue, we also consider the HEA approach [9], which leads to reduced circuit depths. In HEA, the quantum circuits typically consist of alternating layers of parametrized single-qubit rotations and entangling gates (e.g., CNOTS), arranged in a pattern that reflects the hardware topology. While this structure enhances compatibility with NISQ devices and allows for expressive quantum states, it lacks problem-specific encoding.

Our HEA implementations utilize single- or two-layer hardware-efficient SU(2) 2-local quantum circuits, tailored to IBM's Torino device. Figure 2 illustrates the single-layer circuit structure for the case of $N = 4$ qubits. In general, with N qubits, the single- and two-layer circuits comprise, respectively, $4N$ and $6N$ variational parameters.

F. Optimizing VQA parameters

The classical optimization of the quantum circuit parameters in VQAs is a challenging task, known to be NP-hard [50]. The presence of noise and barren plateaus [11], where the gradient effectively vanishes, makes it necessary to find the right trade-off between size and expressivity of the circuits. To mitigate these challenges, various optimization strategies are being employed. In this paper, we consider three such strategies.

In QAOA, we optimize the circuits by a deterministic iterative procedure [51]. We begin by optimizing a single-layer circuit ($p = 1$), with both angles initialized to π . From the optimized parameters obtained with $p - 1$ layers, an initial guess for the parameters of a p -layer circuit is created using interpolation [51].

In HEA, where the basic block (Fig. 2) has more parameters than a single QAOA layer has, we use parameter donation between different problem instances [12,15,52], the aim being to leverage similarities between problem instances to guide the optimizer toward promising

parameter regions. Moving step-by-step upward in problem size, we donate optimized parameters for one circuit as initial values for the optimization of the next. Specifically, we begin with the smallest instance ($N = 4$), the circuit of which is optimized with all parameters initialized randomly. These optimized parameters then serve as initial values for the parameters associated with the first four qubits in the next circuit ($N = 8$). The parameters associated with the remaining four qubits are randomly initialized. This process is repeated iteratively. Note that for every instance, the optimization involves some randomly initialized parameters. Therefore, we performed 20 runs for each instance, corresponding to different realizations of the random initial values. The optimized parameters from the best run (with the highest success probability) are transferred to the next problem instance. We do not transfer parameters between noiseless and noisy simulations to ensure that the optimization remains tailored to specific noise conditions.

Our hardware experiments with HEA are carried out using two sets of optimized parameters for a given problem instance. The sets represent warm starts, obtained as classically optimized parameters under noiseless and noisy conditions, respectively. In both cases, we take the parameters from the best performing of 20 runs.

G. Computational details

All calculations are performed in Python with NumPy [53] and SciPy [54], and all the plots are generated with Matplotlib [55]. Quantum circuit simulations are performed using Qiskit [56] with state vector simulators for noiseless runs and with a noise model derived from IBM's Torino backend for noisy simulations. This noise model incorporates gate errors, readout errors, and thermal relaxation effects based on the device's most recent calibration data. However, it does not account for non-Markovian effects nor crosstalk between qubits. All quantum circuits used in this work are 2-local, meaning each term in the Hamiltonian acts on at most two qubits. All quantum hardware executions are conducted on IBM's Torino device, which features a Heron r1 processor. Optimization is performed using the COBYLA algorithm [57], with a maximum of 10000 iterations. In practice, convergence is typically achieved earlier, resulting in early termination of the optimization process.

H. Analysis

In Sec. III below, we analyze the circuit depths and accuracy of the QAOA and HEA methods when applied to the HP sequence optimization problem. We consider problem instances with unique solutions (Appendix A), which are encoded in the ground state of the cost Hamiltonian C .

For QAOA, we limit ourselves to classical simulations of the quantum circuits. Since the parameter optimization

is deterministic (Sec. II F), we perform one run for each problem instance. As a measure of accuracy, we take the success probability, defined as the absolute square of the overlap between the ground state of C and $U(\theta)|\psi_0\rangle$.

For HEA, we perform both simulations and hardware experiments. In the simulations, using parameter donation and 20 different random number seeds (Sec. II F), we generate 20 sets of optimized parameters for each problem instance. As a measure of accuracy, we compute the average of the success probabilities obtained with these 20 sets of optimized parameters, which is referred to as the success rate. In the hardware experiments, we determine the success rate for each of two choices of optimized parameters (Sec. II F) by averaging over many shots.

III. RESULTS

We investigate the utility of VQAs for identifying HP sequences that minimize the energy E_{HP} for a given target structure and composition (N_{H}). We explore two approaches: QAOAs, with problem-informed quantum circuits (Sec. II D); and HEAs, with problem-agnostic quantum circuits (Sec. II E).

We evaluate the VQAs by classical simulations of the quantum circuits, both with and without noise. The former simulations use the noise model of IBM's Torino device. In the HEA case, we additionally carry out hardware experiments on the IBM Torino device.

In Sec. III A, we present the results obtained for five QAOA variants (Table I), which use the standard X-mixer or one of two XY-mixers. With XY-mixers, it is possible to attain a high success probability for small instances ($N \leq 16$) in noiseless simulations. However, all five QAOA variants suffer from excessive circuit depth, which leads to a sharp decline in success probability when noise is included. These findings suggest that QAOAs, in their current form, are not suitable for implementation on NISQ hardware for this problem.

In Sec. III B, we turn to the HEA approach, which is more compatible with hardware constraints due to its shallower circuit structure. In noiseless simulations, the success rates obtained with HEA are worse than those obtained with the best QAOA variant. However, HEA has the advantage over QAOA of being less sensitive to noise, especially when using a simple single-layer circuit structure. Therefore, our hardware experiments focus entirely on HEA.

A. Problem-informed quantum circuits: QAOA

We evaluate the performance of five QAOA variants (Table I), corresponding to different choices of the mixer Hamiltonian M and the initial state $|\psi_0\rangle$, using as test bed the problem instances in Appendix A with chain lengths $N \leq 16$. The upper limit on problem size was needed due to rapidly growing circuit depths (see below).

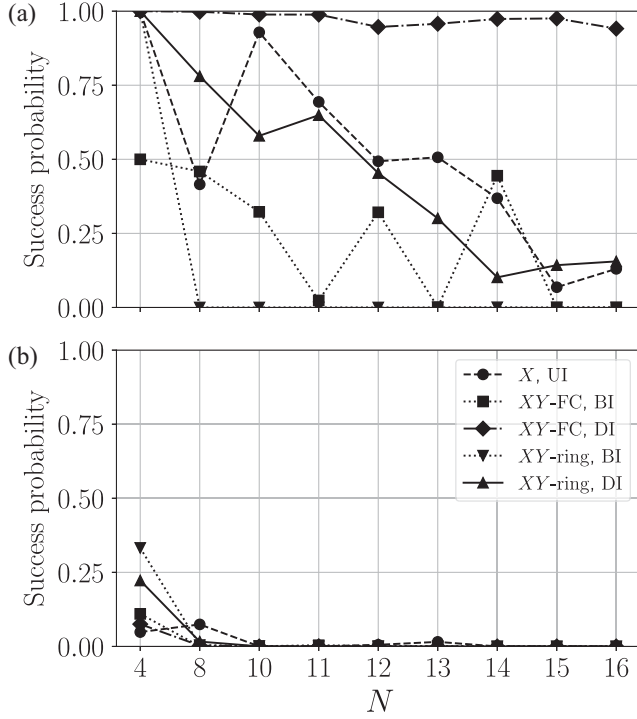


FIG. 3. Simulated success probability as a function of problem size (N) for HP sequence optimization with QAOAs. We consider the five QAOA variants in Table I, and the problem instances with $N \leq 16$ amino acids in Appendix A. All circuits have $p = 15$ layers. The training scheme is deterministic, and the success probability is defined as the ground state probability in the final state. (a) Noiseless simulations. (b) Simulations using the noise model of IBM's Torino device.

The QAOA computations reported below use quantum circuits with $p = 15$ layers, unless otherwise stated. The variational parameters ($\beta_1, \dots, \beta_p, \gamma_1, \dots, \gamma_p$) were determined using an iterative procedure (Sec. II F).

We aim to understand how these circuit design choices affect the algorithm's effectiveness in idealized noiseless simulations and under a hardware-specific noise model. Our analysis highlights how circuit depth, mixer structure, and noise impact the success probability across different problem sizes.

Figure 3 shows the problem size dependence of the simulated success probabilities. In the absence of noise [Fig. 3(a)], the best variant (XY-FC, DI) yields success probabilities ≥ 0.95 for all problem sizes. With the four other variants, the success probability is < 0.5 for $N > 13$. The N -dependence is somewhat irregular, probably in part due to the fact that we consider a single instance for each N . However, two trends can be seen when comparing the data for the four variants with XY-mixers. First, for a given choice of XY-mixer, the Dicke choice of initial state gives the best results. Second, for a given initial state, the fully connected XY-mixer gives the best results. Unfortunately, when adding noise [Fig. 3(b)], the success probability drastically drops for all five variants of QAOA studied.

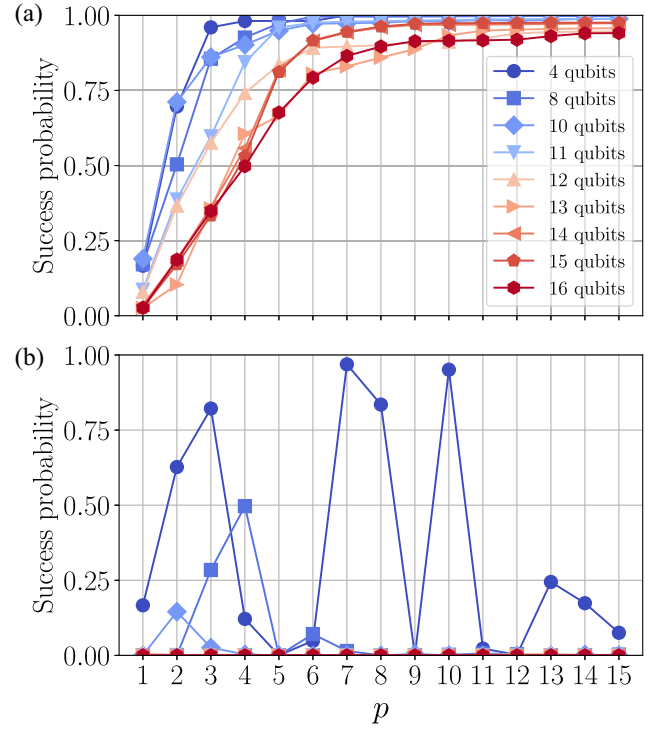


FIG. 4. Simulated success probability as a function of the number of QAOA layers, p . Here, we focus on QAOA variant III in Table I, with a fully connected XY-mixer and a Dicke initial state. We consider the problem instances with $N \leq 16$ in Appendix A. The training scheme is deterministic, and the success probability is defined as the ground state probability in the final state. (a) Noiseless simulations. (b) Simulations using the noise model of IBM's Torino device.

To investigate whether this noise-induced drop in success probability can be avoided by reducing the number of layers, p , we repeated the same calculations for $1 \leq p < 15$, focusing on the QAOA variant found best in Fig. 3(a) (XY-FC, DI). In the absence of noise [Fig. 4(a)], the success probability increases with p , as expected with adequate parameter optimization. We note that in all problem instances, it is possible to reduce p from 15 to ≈ 8 without any major loss in success probability. However, when adding noise [Fig. 4(b)], we again observe a sharp decline in success probability, which for all problem instances with $N > 10$ becomes tiny for all p . For the smallest instances ($N = 4, 8, 10$), there are some values $p < 15$ for which the success probability exceeds its value for $p = 15$.

We attribute the poor performance of the noisy simulations to the considerable circuit depth. Figure 5 shows how the circuit depth increases with problem size for the five QAOA variants. The variant with a fully connected XY-mixer and a Dicke initial state, which performs best in terms of success probability [Fig. 3(a)], is also the one demanding most resources, with a circuit depth of > 2000 for $N = 16$. The overhead arises because both the mixer

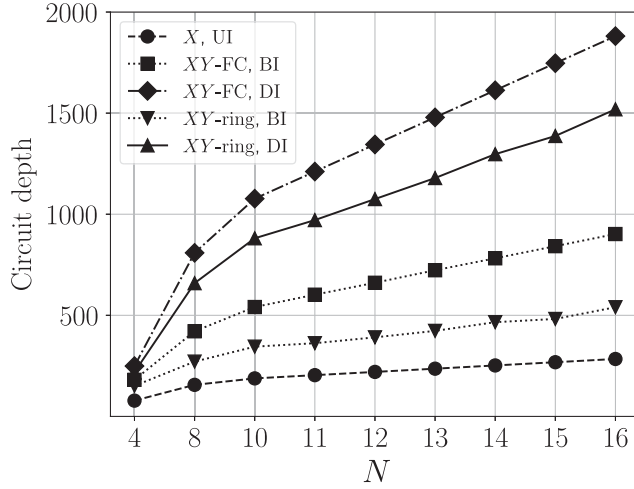


FIG. 5. Circuit depth as a function of problem size (N) for HP sequence optimization with QAOAs. We consider the five QAOA variants in Table I, and the problem instances with $N \leq 16$ in Appendix A. All circuits use $p = 15$ layers.

and the initial state are relatively complex. In contrast, the X -mixer yields shallower circuits but lower success probabilities [Fig. 3(a)].

Throughout the above calculations, the variational parameters were determined through an iterative procedure in p , where the first optimization ($p = 1$) was started with both angles set to π (Sec. II F). To test whether the poor performance of the noisy simulations could be due to the choice of these initial values, we also explored using parameter transfer between problem instances. Here, the $p = 1$ optimization was started from optimized values for a smaller instance. Despite significant overlap between low-energy regions of different $p = 1$ energy landscapes (Appendix B), this parameter transfer did not yield notable improvements. This outcome suggests that the circuit depth is the dominant limiting factor.

B. Problem-agnostic quantum circuits: HEA

To avoid the large circuit depths required by the QAOAs, we also consider the HEA approach, with its problem-agnostic and shallower circuits. Specifically, we investigate the hardware-efficient $SU(2)$ 2-local ansatz, using one- and two-layer circuits (Sec. II E). Figure 6 shows how the circuit depth grows with problem size with this approach. The circuit depths are indeed significantly smaller than they are even with the most shallow QAOA implementation using the standard X -mixer (Fig. 5).

In the HEA computations, we use all the problem instances in Appendix A, with chain lengths up to $N = 29$.

The HEA quantum circuits generate parameterized distributions of bitstrings. The parameters are determined to optimize the objective function, in our case, the average of the energy $E(s)$ in Eq. (2). To estimate this quantity, $E(s)$ is computed classically for a set of bitstrings s

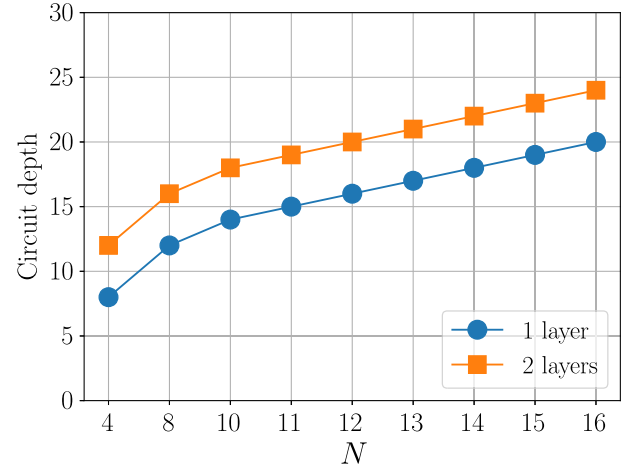


FIG. 6. Circuit depth as a function of problem size (N) for HP sequence optimization with HEAs. We consider one- and two-layer hardware-efficient $SU(2)$ 2-local circuits (Sec. II E), and the problem instances with $N \leq 16$ in Appendix A.

generated by the quantum circuit. We first tried optimizing the variational parameters starting from random initial values between 0 and 2π , however, with poor results for large systems (data not shown). Therefore, we adopted the parameter donation scheme described in Sec. II F.

Figure 7 shows simulated success rates for the one- and two-layer HEAs with and without noise, as obtained using parameter donation from smaller to larger problem instances (Sec. II E). The success rate is defined as the average success probability over 20 independent runs. Additionally, Fig. 7 shows the success probability in the best of the 20 runs.

The results for $N \leq 16$ in Fig. 7 can be compared with those obtained for QAOA (Fig. 3). In the noiseless case, the best QAOA variant (XY-FC, DI) outperforms both one- and two-layer HEA, whose performance, however, is better than or comparable to that of the other four QAOA variants.

With noise, the one-layer HEA [Fig. 7(a)] performs better than the two-layer HEA [Fig. 7(b)], which in turn performs better than any of the QAOA variants [Fig. 3(b)].

For the larger problem sizes ($16 < N \leq 29$), the average success probabilities are in many cases low, especially for the two-layer circuits in the presence of noise [Fig. 7(b)]. However, the run-to-run variations are large, as indicated by significant statistical errors on the mean.

In fact, in the noiseless case, the success probability was often close to one in at least one of the 20 runs, while vanishing in others. Therefore, the best run can have a success probability that is far above the mean (Fig. 7). This effect is particularly pronounced in the two-layer case. While the average success probability is generally lower with two layers instead of one, the number of instances for which the best run has a success probability > 0.75 is higher in the two-layer case. We believe this is because the

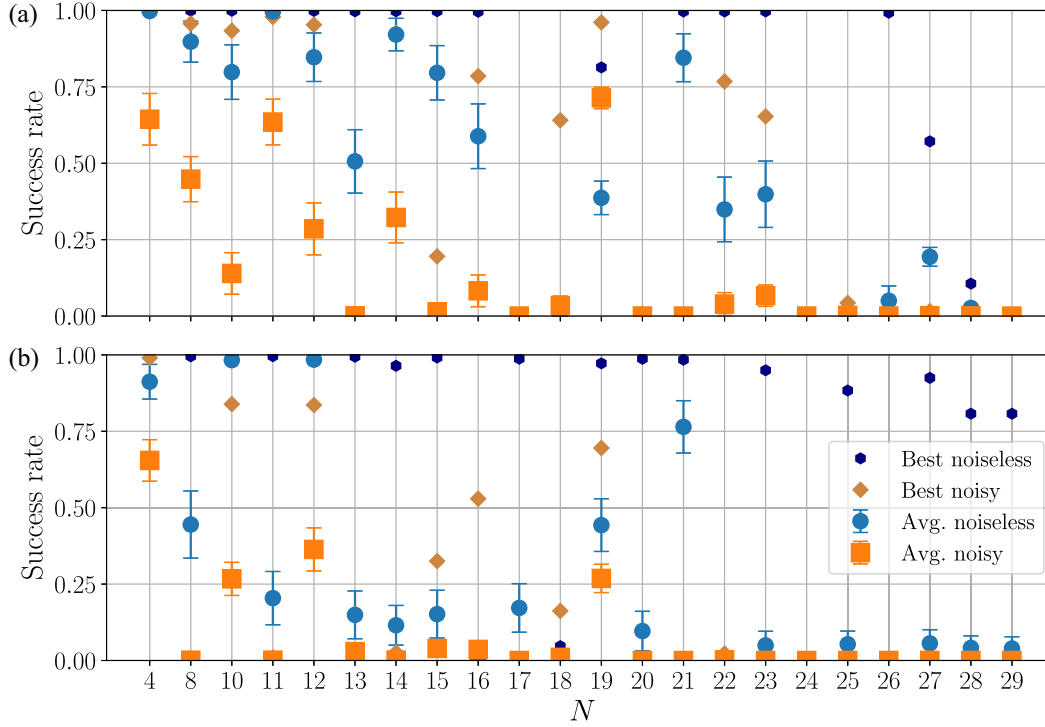


FIG. 7. Simulated success rates as a function of problem size (N) for HP sequence optimization with HEAs, with and without noise. We consider the problem instances in Appendix A. The noisy simulations used IBM's Torino noise model. For each problem instance, we generated 20 noiseless and 20 noisy runs, using different random number seeds for the parameter optimization. We show the average success probabilities over the two sets of runs as well as the highest success probabilities in the two cases. Errors bars indicate one standard error. (a) One-layer HEA. (b) Two-layer HEA.

two-layer circuit, with more parameters, is more expressive, but also more challenging to train.

The success rates in Fig. 7 show a decreasing trend with increasing N , but the decrease is not monotonic. It is important to note monotonicity is not expected. The hardness of a problem does not depend only on N , but also on the choice of target structure and composition (N_H). The results for the $N = 17$ problem illustrate this. This is the smallest N for which the success rate without noise is close to zero for both one- and two-layer HEA (Fig. 7). Interestingly, the same problem also proved hard for its size in a previous quantum annealing study [Fig. 4(b) in Ref. [32]].

Summarizing the simulation results obtained with noise, we find that both HEAs, and especially the single-layer one, perform better than any of the QAOAs studied. However, problem instances with $N > 16$ are difficult to solve in the presence of noise, even with a single-layer HEA.

Because of their better performance in noisy simulations, we also conducted hardware experiments for HEAs, on IBM's Torino device. These experiments were carried out using two warm-start variants, in which the variational parameters were taken from the best simulation runs with and without noise, respectively (see Fig. 7).

Figure 8 shows the results of the hardware experiments for both one- and two-layer HEA. In both cases, the observed success rates tend to be higher when using parameters taken from the noiseless rather than noisy simulations. This indicates that the poorer performance of the noisy simulations (Fig. 7) may in part be due to imperfectly optimized parameters.

When using two layers and parameters from the noiseless simulations, we observe success rates $> 10\%$ for 17 of 22 problem instances, including the three largest systems [Fig. 8(b)]. We note that the systems for which the success rate is below this threshold are precisely those for which the results of the simulations are poor [Fig. 7(b)]. The same pattern can be seen in the one-layer results as well [Figs. 7(a) and 8(a)], although poor success rates occur somewhat more frequently in the one-layer case (7 of 22 problem instances). Therefore, whenever we take the parameters from a noiseless simulation that gave a high success probability, we obtain a significant success rate ($> 10\%$) in the hardware experiments.

When using parameters from the best noisy simulation runs, the hardware experiments yield success rates somewhat lower than those in the simulations, possibly indicating incompleteness of the error model. Still, at a semi-quantitative level, the simulations capture the trends seen in the hardware results.

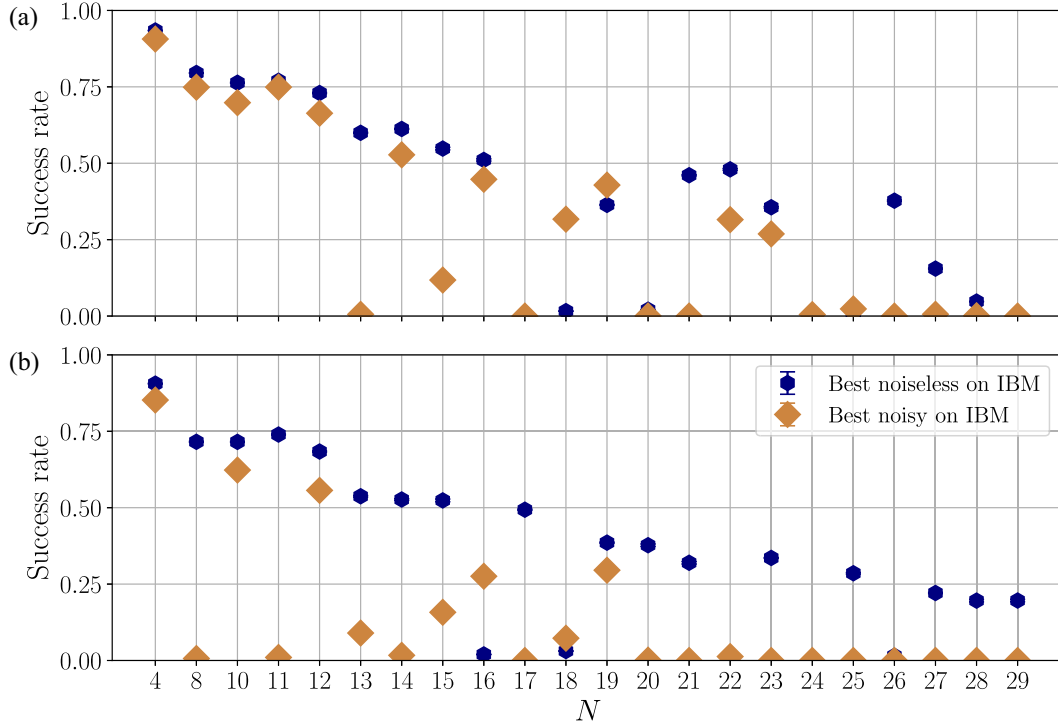


FIG. 8. Success rate of HEA on IBM's Torino device, as a function of problem size (N). We use the hardware-efficient SU(2) 2-local circuit ansatz (Sec. II E) and warm starts, where the parameters are taken from the best simulation runs with and without noise (Fig. 7). We consider the problem instances in Appendix A. Each data point represents an average over 20 experiments, using a fixed set of optimized parameters and 10 000 shots per experiment. Statistical standard errors (1σ) are smaller than the plot symbols. (a) One-layer HEA. (b) Two-layer HEA.

IV. DISCUSSION

Using QAOAs and HEAs with, respectively, problem-informed and problem-agnostic quantum circuits, we have explored the protein sequence optimization problem, with the minimal HP model as a test bed. To this end, we first simulated classically the quantum circuits with and without noise.

With QAOA, it was possible to obtain high success rates under noiseless conditions, especially when using the fully connected XY-mixer and Dicke initial states. However, when adding noise, the performance of all five QAOA variants studied deteriorated. We attribute this noise sensitivity to large circuit depths (Fig. 5).

Note that in the circuit depth analysis for QAOA (Fig. 5), we assumed a method for the preparation of Dicke states that is based on a linear nearest-neighbor topology. In practice, we have access to higher connectivity, which could allow for a more efficient preparation of Dicke states. However, since noise significantly degrades performance even when starting from the other initial states, it is unlikely that an improved Dicke state preparation would yield practical benefits under current hardware limitations.

The shallower one- and two-layer HEAs studied, both based on the hardware-efficient SU(2) 2-local ansatz, showed better noise tolerance. The most noise-tolerant

among the VQAs studied was the minimal one-layer HEA.

Their higher noise tolerance motivated us to conduct hardware experiments with the HEAs, using the IBM Torino device. For both one- and two-layer HEA, the experimental success rates are somewhat lower than the simulated ones with the same parameters, indicating that non-negligible error sources may be missing in the error model. The latter included noise only from gate errors, gate lengths, thermal errors, and readout errors on each qubit, while neglecting temporal and correlated multiqubit errors. Still, overall, the noisy simulations provide a semi-quantitative description of the data from the hardware experiments.

When simulating HEAs, we found that parameter transfer between different problem instances, rather than random parameter initialization for each instance, improved the success probability for large systems. For the problem-informed QAOA circuits, parameter transfer across instances is likely less useful. For QAOA, we instead transferred parameters between same-instance circuits with different numbers of layers (p), following the iterative procedure of Ref. [51]. In the HEA hardware experiments, we found two warm-start approaches to be useful, in which the parameters were directly taken from simulations with or without noise. The same HP sequence

optimization problem was recently addressed using analog quantum computing on a D-Wave annealer [32]. A roughly exponential decay in success rate with problem size was observed when using the purely quantum solver, with a decay rate consistent with control error estimates by D-Wave. For chain length $N = 20$, success rates of approximately 1%–10% were obtained [32]. As in the case of the VQAs studied in the present paper, error mitigation seems essential in order to compete with classical optimization. The sharp deterioration in success rate with problem size is avoided with D-Wave’s hybrid quantum-classical solver, whose classical part divides the problem into smaller sub-problems that are sent as queries to the QPU. With this solver, it was possible to reliably both sequence optimize and fold chains with lengths up to $N = 64$ [32], competing favorably with classical Monte Carlo methods.

Compared to D-Wave’s annealers, the QAOA approach has the advantage of being flexible when it comes to the choice of mixer Hamiltonian and allows for a direct evaluation of the potential quantum advantage, as the characterization of the usage of quantum resources is straightforward. Indeed, under noiseless conditions, our best QAOA results were obtained using an XY -mixer rather than the standard X -mixer (Fig. 3). Unfortunately, the same QAOA variant was also the one with largest circuit depth (Fig. 5), which underscores the need for hardware improvements in order to unlock the potential of QAOA.

V. CONCLUSION AND OUTLOOK

VQAs offer a promising approach to discrete optimization. We have implemented and tested two types of VQAs, QAOAs and HEAs, for sequence optimization in the HP protein model. We find that the more advanced problem-informed approach QAOA suffers from low noise tolerance due to large circuit depths. The problem-agnostic approach HEA, with shallower circuits tailored to the hardware, has better noise tolerance. Using HEA and parameters taken from the best noiseless simulation runs, it was possible to solve the sequence optimization problem on IBM’s Torino device for 17 of the 22 problem instances studied, including the largest system with $N = 29$. With access to an effective parameter determination method, it would likely be possible to obtain significant overlaps with the desired ground state for even larger problems.

Moreover, the success of D-Wave’s hybrid solver with related problems suggests that gate-based methods could benefit from similar problem partitioning approaches and integration with classical heuristics. Therefore, the development and combination of current approaches with more sophisticated classical pre- and postprocessing techniques could help extend the reach of gate-based algorithms to larger problem instances.

ACKNOWLEDGMENTS

We acknowledge support from the Knut and Alice Wallenberg Foundation through the Wallenberg Center for Quantum Technology (WACQT). L. G.-Á. further acknowledges support from the Swedish Foundation for Strategic Research (Grant No. FUS21-0063) and OpenSuperQ-Plus100 (101113946).

DATA AVAILABILITY

The data that support the findings of this study were generated by numerical simulations. The source code and parameters used to generate the simulations are publicly available [58].

CODE AVAILABILITY

The code used to analyze the data is available from Ref. [58].

APPENDIX A: PROBLEM INSTANCES

We solve the HP sequence optimization problem for chain lengths $4 \leq N \leq 29$. Given N , we select one target structure and one value for the number of H beads, N_H . The choice of target structure and N_H is such that (i) the problem of minimizing E_{HP} , given the target structure and N_H , has a unique solution, and (ii) this sequence solution

TABLE II. The number of amino acids, N , the number of H amino acids, N_H , and the known minimum energy, $E_{HP}^{\min}$, for the sequence optimization instances studied.

N	N_H	$E_{HP}^{\min}$
4	2	−1
8	4	−3
10	4	−4
11	5	−4
12	4	−4
13	8	−6
14	8	−7
15	8	−7
16	6	−6
17	6	−6
18	8	−8
19	8	−8
20	8	−8
21	10	−10
22	10	−11
23	10	−10
24	10	−11
25	13	−13
26	14	−14
27	13	−13
28	13	−13
29	15	−15

has the target structure as its unique minimum E_{HP} structure. Property (i) can be checked by visual inspection of the target structures, while property (ii) can be inferred from exhaustive enumerations [37,38].

A list of the problem instances studied can be found in Table II, which also shows the known minimum energies, $E_{\text{HP}}^{\text{min}}$.

APPENDIX B: ENERGY LANDSCAPES FOR QAOA WITH ONE LAYER

To investigate the potential for parameter reuse across problem instances, we examine the QAOA energy

landscapes for depth $p = 1$, i.e., one layer of the algorithm. Figures 9 and 10 show the energy as a function of the variational parameters γ connected with the cost Hamiltonian and β connected with the mixer Hamiltonian for two different protein instances. The similarity in landscape structure suggests that optimal parameters are not highly instance-specific, supporting the viability of parameter donation strategies. This observation motivates the use of previously optimized parameters as warm starts for related problem instances, potentially improving convergence in larger or more complex systems.

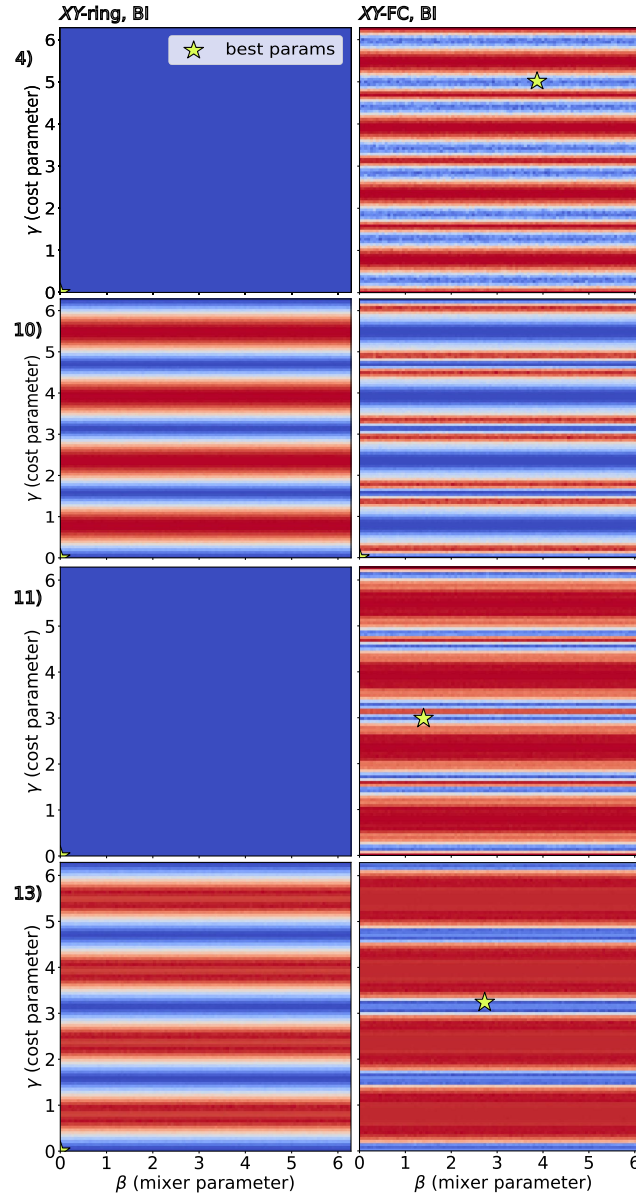


FIG. 9. QAOA energy landscapes for $p = 1$ across different problem instances and circuit structures. Each column corresponds to a different QAOA circuit structure, while each row represents a distinct protein instance. The energy is plotted as a function of the variational parameters γ (cost Hamiltonian) and β (mixer Hamiltonian), with blue indicating low energy and red indicating high energy.

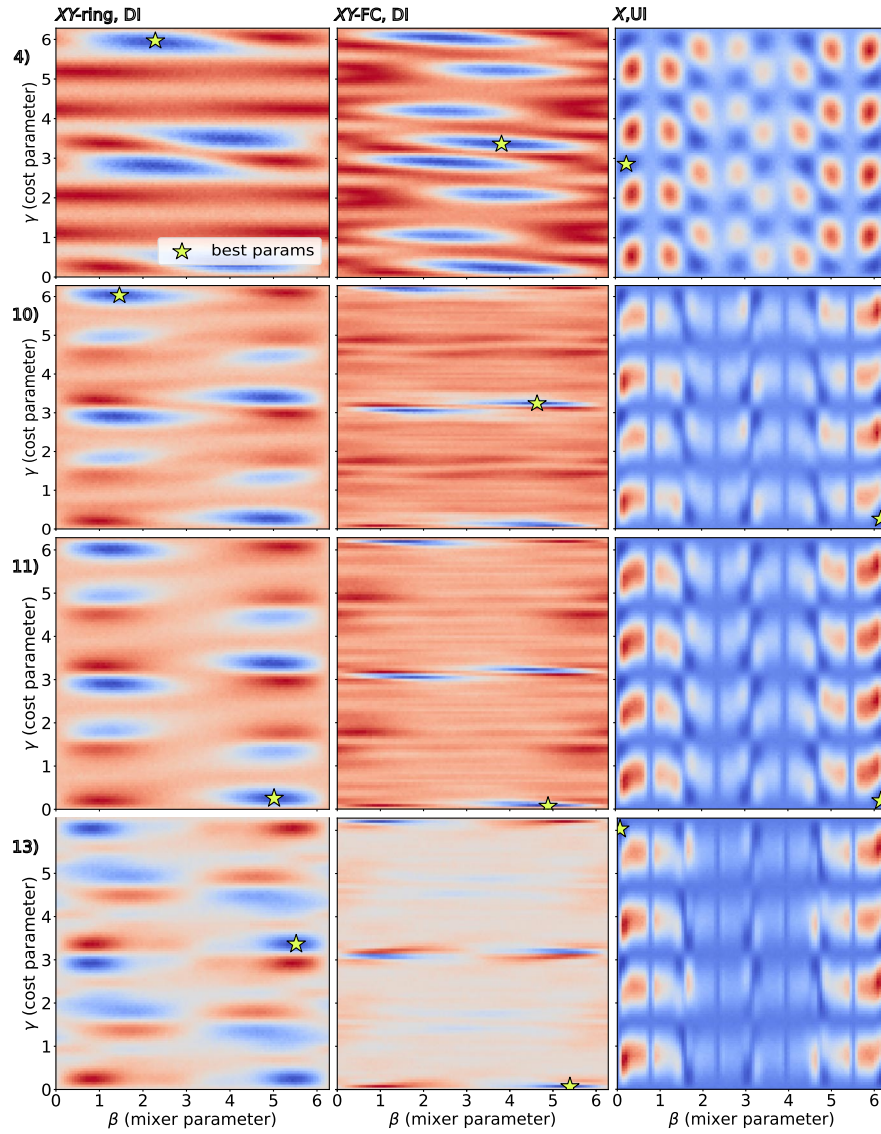


FIG. 10. QAOA energy landscapes for $p = 1$ across different problem instances and circuit structures. Each column corresponds to a different QAOA circuit structure, while each row represents a distinct protein instance. The energy is plotted as a function of the variational parameters γ (cost Hamiltonian) and β (mixer Hamiltonian), with blue indicating low energy and red indicating high energy. The visual similarity between landscapes across instances suggests that optimal parameters are transferable, supporting the use of parameter donation strategies.

-
- [1] K. Wang, Z. Lu, C. Zhang, G. Liu, J. Chen, Y. Wang, Y. Wu, S. Xu, X. Zhu, F. Jin *et al.*, Demonstration of low-overhead quantum error correction codes, *Nat. Phys.* **22**, 308 (2026).
 - [2] S. Krinner, N. Lacroix, A. Remm, A. Di Paolo, E. Genois, C. Leroux, C. Hellings, S. Lazar, F. Swiadek, J. Herrmann *et al.*, Realizing repeated quantum error correction in a distance-three surface code, *Nature (London)* **605**, 669 (2022).
 - [3] R. Acharya, I. Aleiner, R. Allen, T. I. Andersen, M. Ansmann, F. Arute, K. Arya, A. Asfaw, J. Atalaya, R. Babbush *et al.*, Suppressing quantum errors by scaling a surface code logical qubit, *Nature (London)* **614**, 676 (2023).
 - [4] R. Acharya, D. A. Abanin, L. Aghababaie-Beni, I. Aleiner, T. I. Andersen, M. Ansmann, F. Arute, K. Arya, A. Asfaw, N. Astrakhantsev *et al.*, Quantum error correction below the surface code threshold, *Nature (London)* **638**, 920 (2024).
 - [5] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio *et al.*, Variational quantum algorithms, *Nat. Rev. Phys.* **3**, 625 (2021).

- [6] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, [arXiv:1411.4028](#).
- [7] S. Hadfield, Z. Wang, B. O’Gorman, E. G. Rieffel, D. Venturelli, and R. Biswas, From the quantum approximate optimization algorithm to a quantum alternating operator ansatz, *Algorithms* **12**, 34 (2019).
- [8] D. Wecker, M. B. Hastings, and M. Troyer, Progress towards practical quantum variational algorithms, *Phys. Rev. A* **92**, 042303 (2015).
- [9] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets, *Nature (London)* **549**, 242 (2017).
- [10] J. Preskill, Quantum computing in the NISQ era and beyond, *Quantum* **2**, 79 (2018).
- [11] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, Barren plateaus in quantum neural network training landscapes, *Nat. Commun.* **9**, 4812 (2018).
- [12] Q. Langfitt, J. Falla, I. Safro, and Y. Alexeev, Parameter transferability in QAOA under noisy conditions, in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)* (2023), Vol. 02, pp. 300–301.
- [13] J. A. Montañez-Barrera, D. Willsch, A. Maldonado-Romo, and K. Michielsen, Unbalanced penalization: A new approach to encode inequality constraints of combinatorial problems for quantum optimization algorithms, *Quantum Sci. Technol.* **9**, 025022 (2024).
- [14] A. Galda, E. Gupta, J. Falla, X. Liu, D. Lykov, Y. Alexeev, and I. Safro, Similarity-based parameter transferability in the quantum approximate optimization algorithm, *Front. Quantum Sci. Technol.* **2**, 1200975 (2023).
- [15] I. Lyngfelt and L. García-Álvarez, Symmetry-informed transferability of optimal parameters in the quantum approximate optimization algorithm, *Phys. Rev. A* **111**, 022418 (2025).
- [16] S. Kosen, H.-X. Li, M. Rommel, D. Shiri, C. Warren, L. Grönberg, J. Salonen, T. Abad, J. Biznárová, M. Caputo *et al.*, Building blocks of a flip-chip integrated superconducting quantum processor, *Quantum Sci. Technol.* **7**, 035018 (2022).
- [17] M. Fingerhuth, T. Babej, and C. Ing, A quantum alternating operator ansatz with hard and soft constraints for lattice protein folding, [arXiv:1810.13411](#).
- [18] A. Robert, P. K. Barkoutsos, S. Woerner, and I. Tavernelli, Resource-efficient quantum algorithm for protein folding, *npj Quantum Inf.* **7**, 38 (2021).
- [19] S. Boulebnane, X. Lucas, A. Meyder, S. Adaszewski, and A. Montanaro, Peptide conformational sampling using the quantum approximate optimization algorithm, *npj Quantum Inf.* **9**, 70 (2023).
- [20] H. Linn, R.-H. Li, A. Holden, A. A. Saki, F. DiFilippo, T. Radivoyevitch, D. Blankenberg, L. García-Álvarez, and G. Johansson, Efficient quantum protein structure prediction with problem-agnostic ansatzes, [arXiv:2509.18263](#).
- [21] A. Perdomo-Ortiz, N. Dickson, M. Drew-Brook, G. Rose, and A. Aspuru-Guzik, Finding low-energy conformations of lattice protein models by quantum annealing, *Sci. Rep.* **2**, 248 (2012).
- [22] C. Outeiral, G. M. Morris, J. Shi, M. Strahm, S. C. Benjamin, and C. M. Deane, Investigating the potential for a limited quantum speedup on protein lattice problems, *New J. Phys.* **23**, 103030 (2021).
- [23] A. Irbäck, L. Knuthson, S. Mohanty, and C. Peterson, Folding lattice proteins with quantum annealing, *Phys. Rev. Res.* **4**, 043013 (2022).
- [24] A. Irbäck, L. Knuthson, and S. Mohanty, Folding lattice proteins confined on minimal grids using a quantum-inspired encoding, *Phys. Rev. E* **112**, 045302 (2025).
- [25] H. Linn, I. Lyngfelt, L. García-Álvarez, and Göran Johansson, Resource analysis of quantum algorithms for coarse-grained protein folding models, *Phys. Rev. Res.* **6**, 033112 (2024).
- [26] B. Kuhlman, G. Dantas, G. C. Ireton, G. Varani, B. L. Stoddard, and D. Baker, Design of a novel globular protein fold with atomic-level accuracy, *Science* **302**, 1364 (2003).
- [27] G. Bhardwaj, V. K. Mulligan, C. D. Bahl, J. M. Gilmore, P. J. Harvey, O. Cheneval, G. W. Buchko, S. V. S. R. K. Pulavarti, Q. Kaas, A. Eletsky *et al.*, Accurate de novo design of hyperstable constrained peptides, *Nature (London)* **538**, 329 (2016).
- [28] K. K. Yang, Z. Wu, and F. H. Arnold, Machine-learning-guided directed evolution for protein engineering, *Nat. Methods* **16**, 687 (2019).
- [29] B. Kuhlman and P. Bradley, Advances in protein structure prediction and design, *Nat. Rev. Mol. Cell Biol.* **20**, 681 (2019).
- [30] L. Cao, I. Goreschnik, B. Coventry, J. B. Case, L. Miller, L. Kozodoy, R. E. Chen, L. Carter, A. C. Walls, Y.-J. Park *et al.*, De novo design of picomolar SARS-CoV-2 mini-protein inhibitors, *Science* **370**, 426 (2020).
- [31] V. K. Mulligan, H. Melo, H. I. Merritt, S. Slocum, B. D. Weitzner, A. M. Watkins, P. D. Renfrew, C. Pelissier, P. S. Arora, and R. Bonneau, Designing peptides on a quantum computer, [10.1101/752485](#).
- [32] A. Irbäck, L. Knuthson, S. Mohanty, and C. Peterson, Using quantum annealing to design lattice proteins, *Phys. Rev. Res.* **6**, 013162 (2024).
- [33] V. Panizza, P. Hauke, C. Micheletti, and P. Faccioli, Protein design by integrating machine learning and quantum-encoded optimization, *PRX Life* **2**, 043012 (2024).
- [34] M. H. Khatami, U. C. Mendes, N. Wiebe, P. M. Kim, and N. Ben-Tal, Gate-based quantum computing for protein design, *PLOS Comput. Biol.* **19**, 1 (2023).
- [35] K. F. Lau and K. A. Dill, A lattice statistical mechanics model of the conformational and sequence spaces of proteins, *Macromolecules* **22**, 3986 (1989).
- [36] K. Yue, K. M. Fiebig, P. D. Thomas, H. S. Chan, E. I. Shakhnovich, and K. A. Dill, A test of lattice protein folding algorithms, *Proc. Natl. Acad. Sci.* **92**, 325 (1995).
- [37] A. Irbäck and C. Troein, Enumerating designing sequences in the HP model, *J. Biol. Phys.* **28**, 1 (2002).
- [38] C. Holzgräfe, A. Irbäck, and C. Troein, Mutation-induced fold switching among lattice proteins, *J. Chem. Phys.* **135**, 195101 (2011).
- [39] A. Irbäck, C. Peterson, F. Potthast, and E. Sandelin, Design of sequences with good folding properties in coarse-grained protein models, *Structure* **7**, 347 (1999).
- [40] A. Aina and S. Wallin, Multisequence algorithm for coarse-grained biomolecular simulations: Exploring the sequence-structure relationship of proteins, *J. Chem. Phys.* **147**, 095102 (2017).

- [41] D. Nilsson and A. Irbäck, Finite-size scaling analysis of protein droplet formation, *Phys. Rev. E* **101**, 022413 (2020).
- [42] A. Statt, H. Casademunt, C. P. Brangwynne, and A. Z. Panagiotopoulos, Model for disordered proteins with strongly sequence-dependent liquid phase behavior, *J. Chem. Phys.* **152**, 075101 (2020).
- [43] E. Bornberg-Bauer and H. S. Chan, Modeling evolutionary landscapes: Mutational stability, topology, and superfunnels in sequence space, *Proc. Natl. Acad. Sci. USA* **96**, 10689 (1999).
- [44] J. Aguirre, P. Catalán, J. A. Cuesta, and S. Manrubia, On the networked architecture of genotype spaces and its critical effects on molecular evolution, *Open Biol.* **8**, 180069 (2018).
- [45] T. Kadowaki and H. Nishimori, Quantum annealing in the transverse Ising model, *Phys. Rev. E* **58**, 5355 (1998).
- [46] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem, *Science* **292**, 472 (2001).
- [47] I. Hen and F. M. Spedalieri, Quantum annealing for constrained optimization, *Phys. Rev. Appl.* **5**, 034007 (2016).
- [48] Z. Wang, N. C. Rubin, J. M. Dominy, and E. G. Rieffel, XY mixers: Analytical and numerical results for the quantum alternating operator ansatz, *Phys. Rev. A* **101**, 012320 (2020).
- [49] A. Bärttschi and S. Eidenbenz, Deterministic preparation of Dicke states, *Fundamentals of Computation Theory*, edited by L. A. Gąsieniec, J. Jansson, and C. Levcopoulos (Springer International Publishing, Cham, 2019), pp. 126–139.
- [50] L. Bittel and M. Kliesch, Training variational quantum algorithms is NP-hard, *Phys. Rev. Lett.* **127**, 120502 (2021).
- [51] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices, *Phys. Rev. X* **10**, 021067 (2020).
- [52] J. A. Montañez-Barrera, D. Willsch, and K. Michielsen, Transfer learning of optimal QAOA parameters in combinatorial optimization, *Quantum Inf. Process.* **24**, 129 (2025).
- [53] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith *et al.*, Array programming with NumPy, *Nature (London)* **585**, 357 (2020).
- [54] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright *et al.*, SciPy 1.0: Fundamental algorithms for scientific computing in python, *Nat. Methods* **17**, 261 (2020).
- [55] J. D. Hunter, Matplotlib: A 2D graphics environment, *Comput. Sci. Eng.* **9**, 90 (2007).
- [56] A. Javadi-Abhari, M. Treinish, K. Krsulich, C. J. Wood, J. Lishman, J. Gacon, S. Martiel, P. D. Nation, L. S. Bishop, A. W. Cross *et al.*, Quantum computing with Qiskit, [arXiv:2405.08810](https://arxiv.org/abs/2405.08810).
- [57] M. J. D. Powell, A direct search optimization method that models the objective and constraint functions by linear interpolation, *Advances in Optimization and Numerical Analysis*, edited by S. Gomez and J.-P. Hennart (Springer Netherlands, Dordrecht, 1994), pp. 51–67.
- [58] H. Linn, L. Knuthson, and L. García-Álvarez, Github repository name: Designing lattice proteins with variational quantum algorithms (2025).