ARTICLE OPEN

Check for updates

Space-efficient binary optimization for variational quantum computing

Adam Glos [1,✉], Aleksandra Krawiec[1] and Zoltán Zimborás [2,3]

In the era of Noisy Intermediate-Scale Quantum (NISQ) computers it is crucial to design quantum algorithms which do not require many qubits or deep circuits. Unfortunately, most of the well-known quantum algorithms are too demanding to be run on currently available quantum devices. Moreover, even the state-of-the-art algorithms developed for the NISQ era often suffer from high space complexity requirements for particular problem classes. In this paper, we show that it is possible to greatly reduce the number of qubits needed for the Travelling Salesman Problem (TSP), a paradigmatic optimization task, at the cost of having deeper variational circuits. While the focus is on this particular problem, we claim that the approach can be generalized for other problems where the standard bit-encoding is highly inefficient. Finally, we also propose encoding schemes which smoothly interpolate between the qubit-efficient and the circuit depth-efficient models. All the proposed encodings have the same volume up to polylogarithmic factors and remain efficient to implement within the Quantum Approximate Optimization Algorithm framework.

INTRODUCTION

During the past few decades, a wealth of quantum algorithms have been designed for various problems, many of which offered a speedup over their classical equivalents[1–4]. The theoretical developments have also been complemented by progress on the experimental side. Indeed, the demonstration of quantum supremacy by Google[5] indicates that in the near future useful quantum technology may be available. However, current Noisy Intermediate-Scale Quantum (NISQ) devices are too small and too noisy to implement complicated algorithms like Shor's factorization algorithm[2] or Grover-search-based optimizers[6,7]. This resulted in a new field of quantum computation that focuses on designing new algorithms requiring significantly less noisy qubits.

Optimization is a problem that seems to be particularly suitable for current NISQ devices. In particular, the Variational Quantum Eigensolver (VQE)[8–10] seems to be the state-of-the-art algorithm for solving molecule Hamiltonians. Although it can solve optimization problems defined over discrete spaces, so-called combinatorial optimization problems, quantum annealing and Quantum Approximate Optimization Algorithm (QAOA)[11] are considered to be more suitable.

For all of these algorithms, the original combinatorial optimization problem has to be transformed into the Ising model. Typically, one starts with a high-level description, like the Max-Cut problem, where nodes in the graph $G = (V, E)$ have to be colored either red or black, so that certain function is minimized. Then, one has to transform it to a pseudo-Boolean polynomial $\sum_{\{u,v\}\in E}(b_u - b_v)^2$. Each binary variable (bit) b_v denotes the color of the node v in the graph. For example we can choose $b_v = 1$ for red color, and $b_v = 0$ for black color. For quantum algorithms it is convenient to change the representation into Ising model via transformation $b_v \leftarrow (1 - Z_v)/2$. Here Z_v is a Pauli operator acting on a qubit corresponding to the node v. By transforming the original objective function into Hamiltonian, we also change the domain of the problem into the space of quantum states.

Quantum optimization algorithms differ in the way how they solve the problem. Variational Quantum Eigensolver (VQE) is a heuristic algorithm in which the quantum circuit is optimized using classical procedure. More precisely, we are given an *ansatz* $U(\theta)$ which, after fixing the parameter θ, produces a state $|\theta\rangle = \prod_{i=0}^{k-1} U_i(\theta_i)|0\rangle$. The vector θ is optimized using classical optimization procedure like gradient descent, simultaneous perturbation stochastic approximation (SPSA), and other[12,13], so that $|\theta\rangle$ will be localized at high-quality answer. Due to its generality, VQE is commonly used for molecule Hamiltonians, however its usability to the classical optimization problems may be limited.

Quantum annealing theoretically can also be applied for chemistry Hamiltonians, however current machines restrict their usability to combinatorial optimization problems. The algorithm turns the ground state of initial Hamiltonian $H_{mix} = \sum_i X_i$ into a ground state of objective Hamiltonian H through adiabatic evolution $g(t)H_{mix} + (1 - g(t))H$. Adiabatic theorem provides a good premise for high-quality solutions of the problem. Furthermore, while available D-Wave's annealers have thousands of qubits, the topology restrictions may limit the size of tractable problems to cases solvable by classical procedures[14,15].

Quantum Approximate Optimization Algorithm (QAOA) is a mixture of the methods described above[11]. While quantum annealing is a continuous process, QAOA interchangeably applies both H_{mix} and H for some time. The evolution time is a parameter of the evolution, and as it was in the case of VQE, they are adjusted by external classical procedure. Here the resulting state takes the form

$$|\theta\rangle = \prod_{i=0}^{r-1} \exp(-i\theta_{mix,i}H_{mix}) \exp(-i\theta_{obj,i}H)|+\rangle, \tag{1}$$

where r is the number of levels. The algorithm can be implemented as long as both mixing and objective Hamiltonians can be implemented, which in particular allows for applying it to combinatorial problems. Many studies have been performed to

characterize properties of QAOA algorithms, including both rigorous proofs of computational power and reachability properties[16–19] as well as characterization through heuristics and numerical experiments and extensions of the algorithm[20–24].

While the proposed quantum algorithms can theoretically solve arbitrary combinatorial optimization problems, not all pseudo-Boolean polynomials can be natively considered for current quantum devices. A general pseudo-Boolean polynomial takes the form

$$H(b) = \sum_{I \subseteq \{0, \ldots, n-1\}} a_I \prod_{i \in I} b_i,$$

where $a_I \in \mathbb{R}$ defines the optimization problem. The problem of finding optimum of such function is called Higher-Order Binary Optimization (HOBO). An order of such Hamiltonian is a maximum size of S for which a_S is nonzero. Current D-Wave machines are restricted to polynomials of order 2, hence if one would like to solve Hamiltonians of higher order, first a quadratization procedure has to be applied, in general at cost of extra qubits[25]. Note that optimization of quadratic polynomials, Quadratic Unconstrained Binary Optimization (QUBO) is NP-complete, hence it encapsulates most of the relevant problems.

The objective Hamiltonian for Max-Cut requires n qubits for graph of order n. Hence it can encode 2^n solutions, which is equal to the number of all possible colorings. However, this is not the case in general. For example for Traveling Salesman Problem (TSP) over N cities, the QUBO requires N^2 bits[26]. However, to store $N!$ permutations only $\lceil \log_2(N!) \rceil \approx N \log N$ bits are needed. We consider this as a waste of computational resources. Unfortunately, in general polynomials with optimal number of qubits have order larger than two, thus we are actually dealing with higher-order binary optimization, which is currently not possible using D-Wave machines.

The idea of using higher-order terms is not new. In fact, in the original work of Farhi et al.[11], the authors have not restricted the model to two-local model. Furthermore, Hamiltonian of order 4 was used for variational quantum factoring[27], while Hamiltonian of order l was constructed for Max-l-SAT problem[20]. Since the terms of arbitrary order can be implemented efficiently, QAOA for the problem can reduce the number of required logical qubits. In general, if objective polynomial is of constant order a, then the circuit of depth $\mathcal{O}\left(\binom{n}{a} \log a\right)$ implements the objective Hamiltonian exactly. While the number may be large, it is still polynomial, which makes the implementation tractable. However, even for slowly growing a (say $\mathcal{O}(\log n)$), in general the number of terms grows exponentially, which could be the case for $l \to \infty$ in Max-l-SAT. Note that even an encoding that requires only logarithmic number of qubits has been introduced[28], however, the minimizer of this encoding does not necessarily map to the minimizer of the original problem.

Furthermore, when dealing with unbounded order, one has to be careful when transforming QUBO into the Ising model. Let us consider a polynomial $2^n \prod_{i=0}^{n-1} b_i$. Default transformation $b_i \leftarrow (1 - Z_i)/2$ will produce Hamiltonian $\sum_{I \subseteq \{0, \ldots, n-1\}} (-1)^{|I|} \prod_{i \in I} Z_i$, which consists of 2^n terms[29]. For this example, one can easily find a better Hamiltonian $-\sum_{i=0}^{n-1} Z_i$, that shares the same global minimizer, however, in general finding such a transformation requires a higher-level understanding of the problem. Note that this is not an issue for constant-order polynomials, as the number of terms is guaranteed to be polynomial even in the worst case scenario.

Despite the potential issues coming from utilizing unbounded-order polynomials, we present a polynomial (encoding) for TSP problem with unbounded order, which can be efficiently implemented using approximately optimal number of qubits. Furthermore, our model requires fewer measurements for

estimating energy. We also developed a transition encoding, where one can adjust the improvement in the required number of qubits and circuit's depth. Finally, QAOA optimizes our encoding with similar or better efficiency compared to the state-of-the-art QUBO encoding of TSP.

Current state-of-the-art encoding of TSP problem can be found in the paper by Lucas[26]. Let us consider the Traveling Salesman Problem (TSP) over N cities. Let W be a cost matrix, and b_{ti} be a binary variable such that $b_{ti} = 1$ iff the i-th city is visited at time t. The QUBO encoding takes the form[26]

$$H^{\text{QUBO}}(b) = A_1 \sum_{t=0}^{N-1} \left(1 - \sum_{i=0}^{N-1} b_{ti}\right)^2 + A_2 \sum_{i=0}^{N-1} \left(1 - \sum_{t=0}^{N-1} b_{ti}\right)^2$$
$$+ B \sum_{\substack{i,j=0 \\ i \neq j}}^{N-1} W_{ij} \sum_{t=0}^{N-1} b_{ti} b_{t+1,j}.$$

Here $A_1, A_2 > B \max_{i \neq j} W_{ij}$ are parameters that have to be adjusted during the optimization. We also assume $N \to 0$ simplification for the indices. Note that any route can be represented in N different ways, depending on which city is visited at time $t = 0$. Such redundancy can be solved by fixing that the first city should be visited at time $t = 0$. Thanks to that, $n = (N-1)^2$ (qu)bits in total are required.

In the scope of this paper we will take advantage of various quality measures of encodings. First, since the Hamiltonian has to be implemented directly, we prefer encodings with possibly small depth. In this manner, QUBO can be simulated with a circuit of depth $\mathcal{O}(n)$ using round-robin scheduling, see Fig. 2.

However, QUBO encoding for TSP is inefficient in the number of qubits. Using Stirling formula one can show that $\lceil \log(N!) \rceil = N \log(N) - N \log(e) + \Theta(\log(N))$ qubits are sufficient to encode all possible routes, which is significantly smaller than N^2. Note that the number of qubits also has an impact on the volume of the circuit, defined as a product of the number of qubits and the circuit's depth. In case of this encoding, the volume is of order $\mathcal{O}(N^3)$.

In the paper we also consider the required number of measurements to estimate the energy within constant additive error. Instead of estimating each term of the Hamiltonian independently, which has to be done for VQE, we consider the measurement's output as a single sample. This way, using Hoeffding's theorem, QUBO encoding requires $\mathcal{O}(N^3)$ measurements.

Finally, we would also like to mention some other approaches of reducing the number of qubits required for optimization problems. The authors of[30] proposed a similar approach of changing the numeric system, however, they have not considered its applications to combinatorial problems, and particularly focused on qudits. The authors of Fuchs et al.[31] independently considered Max-K-Cut, which is adealing with unbounded order, one has to be generalization of Max-Cut with K-partitions of the graph. They obtained a similar reduction from NK to $\sim N \log K$ qubits, however, they did not provide a transition model which provides a trade-off between the circuit depth and the required number of qubits. Finally, there are known QAOA versions in which one reduces the size of infeasible space for TSP problems based on the generalization of QAOA[20], however, they still require N^2 number of qubits, which is the same as for the state-of-the-art encoding[32].

RESULTS

Preliminaries

Travelling Salesman Problem is natively defined over the permutations of $\{0, \ldots, N-1\}$. A simple encoding can be defined as follows. We make a partition of all bits into N collections b_t, where each collection encodes a city visited in a particular time. Then, for each collection we choose a number encoding that represents the city.
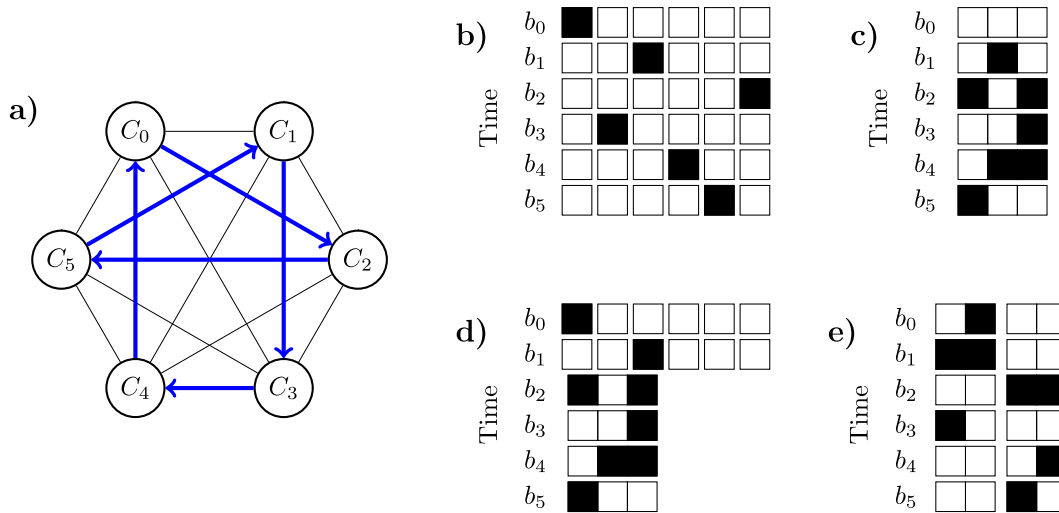
Fig. 1 Visualization of QUBO encoding and encodings introduced in the paper for TSP problem. a exemplary solution for TSP problem. On the right there are assignments of the exemplary solution using respectively **b** QUBO, **c** HOBO, **d** naïve mixed, and **e** mixed encodings.

QUBO is an example of such an encoding, where each city is represented by an one-hot vector, see Fig. 1. Instead, each city can be encoded as a number using binary numbering system. Using binary numbering system is a state-of-the-art way to encode inequalities[26], however, it is new in the context of encoding elements of a feasible space.

The Hamiltonian takes the form

$$
H(b) = A_1 \sum_{t=0}^{N-1} H_{\text{valid}}(b_t) + A_2 \sum_{t=0}^{N-1} \sum_{t'=t+1}^{N-1} H_{\neq}(b_t, b_{t'})
$$
$$
+ B \sum_{\substack{i,j=0 \\ i \neq j}}^{N-1} W_{ij} \sum_{t=0}^{N-1} H_\delta(b_t, i) H_\delta(b_{t+1}, j). \quad (2)
$$

Hamiltonian H_{valid} checks whether a vector of bits b_t encodes a valid city. For example, for QUBO it checks whether exactly one bit is equal to 1.

Hamiltonian $H_{\neq}$ verifies whether two collections encode the same city. Note that QUBO encoding falls into this representation by choosing

$$
H_{\neq}^{\text{QUBO}}(b_t, b_{t'}) = 2 \sum_{i=0}^{N-1} b_{ti} b_{t'i} - \frac{1}{N-1} \left(\sum_{i=0}^{N-1} b_{ti} + \sum_{i=0}^{N-1} b_{t'i} + 2 \right)
$$

Hamiltonian H_δ plays a similar role as $H_{\neq}$. If the inputs are different, then both H_δ and $H_{\neq}$ give zeros. If the inputs are the same, then the outputs are nonzero and moreover we expect that the Hamiltonian H_δ outputs 1. This is in order to preserve the costs of routes. In case of QUBO we took $H_\delta(b_t, i) = b_{ti}$. Note that in particular, $H_\delta = H_{\neq}$ may be a good choice, however, later we will show that choosing different $H_{\neq}$ may be beneficial.

Simple HOBO encoding

The simplest encoding is the one in which each collection b_t encodes a city in a binary system, see Fig. 1. In this case, for each time we need $K := \lceil \log N \rceil$ qubits. In total we need $\sim N \log(N)$ qubits, which match the lower bound. Moreover, we have to design H_{valid} in such a way that b_t represents the number at most $N - 1$.

Following Eq. (2), it is easy to note that HOBO defined in a way described above, is of polynomial size. Note that the sum of

Hamiltonians $H_{\neq}$ produces at most $\binom{N}{2} 2^{2K}$ elements.

Furthermore, the terms introduced by $H_{\text{valid}}^{\text{HOBO}}$ and H_δ^{HOBO} are already present in $H_{\neq}^{\text{HOBO}}$. Hence in total, we have $\mathcal{O}(N^4)$ terms, which implies the polynomial size and depth, and thus volume.

Let us now present an exemplary encoding. Suppose $\tilde{b}_{K-1} \ldots \tilde{b}_0$ is a binary representation of $N - 1$. Suppose $k_0 \in K_0$ are such indices that $\tilde{b}_{k_0} = 0$. Then one can show that

$$
H_{\text{valid}}^{\text{HOBO}}(b_t) := \sum_{k_0 \in K_0} b_{t,k_0} \prod_{k=k_0+1}^{K-1} \left(1 - (b_{t,k} - \tilde{b}_k)^2 \right)
$$

validates whether the encoding number is at most $N - 1$. A detailed proof can be found in Supplementary Methods, here let us consider an example. Suppose $N - 1 = 100101_2$. All the numbers larger than $N - 1$ are of the form $11????_2$, $101???_2$ or $10011?_2$, where '?' denotes an arbitrary bit value. The polynomial

$$
b_{t5} b_{t4} + b_{t5}(1 - b_{t4}) b_{t3} + b_{t5}(1 - b_{t4})(1 - b_{t3}) b_{t2} b_{t1} \quad (3)
$$

punishes all these forms. At the same time, one can verify that numbers smaller than $N - 1$ are not punished by the Hamiltonian.

Here, we will consider $H_{\neq}^{\text{HOBO}} \equiv H_\delta^{\text{HOBO}}$, hence it is enough to define the latter only. Hamiltonian H_δ^{HOBO} can be defined as

$$
H_\delta^{\text{HOBO}}(b, b') := \prod_{k=0}^{K-1} \left(1 - (b_k - b'_k)^2 \right).
$$

Note that if b' is a fixed number like it is in the case of objective function implementation in Eq. (2), then we simply take consecutive bits from binary representation. After transforming the polynomial above into Ising model, one can see that $(b_k - b'_k)^2 = \frac{1}{4}(Z'_k - Z_k)^2 = \frac{1}{2}(1 - Z'_k Z_k)$. This means that all the Pauli terms are of the form where spins are in pairs with the same index k. This allows reducing the initial estimate $\mathcal{O}(N^4)$ of number of Pauli terms into $\mathcal{O}(N^3)$.

Let us estimate the cost of this encoding. As it was previously stated, the number of factors is of order $\mathcal{O}(N^3)$. Using round-robin technique for distributing gates and Gray code for ordering the applications of Pauli $ZZ \ldots Z$ operators, see Figs. 2 and 3, one can show that the depth of the circuit is $\mathcal{O}(N^2)$, which gives us the volume $\mathcal{O}(N^3 \log(N))$, which is almost the same as it was for QUBO. Note that the Gray code scheduling requires additional $\lfloor N/2 \rfloor$ qubits, which does not change the final result. Finally, in order to achieve a similar quality of energy measurements, we need $\mathcal{O}(N^2)$ measurements.
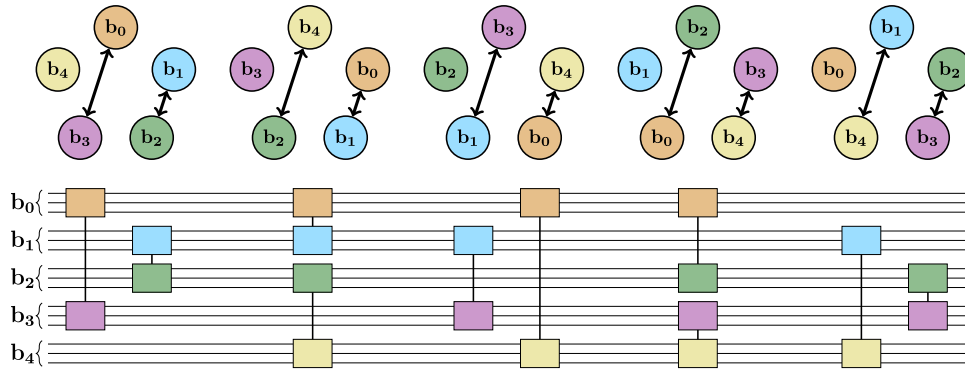
Fig. 2   Round-robin schedule for binary vectors $b_0, \ldots, b_4$. A description of the schedule can be found in[40]. We assumed that each b_i is defined over 3 qubits. Each gate defined over a pair b_i, b_j is an implementation of the Hamiltonian defined over these variables.
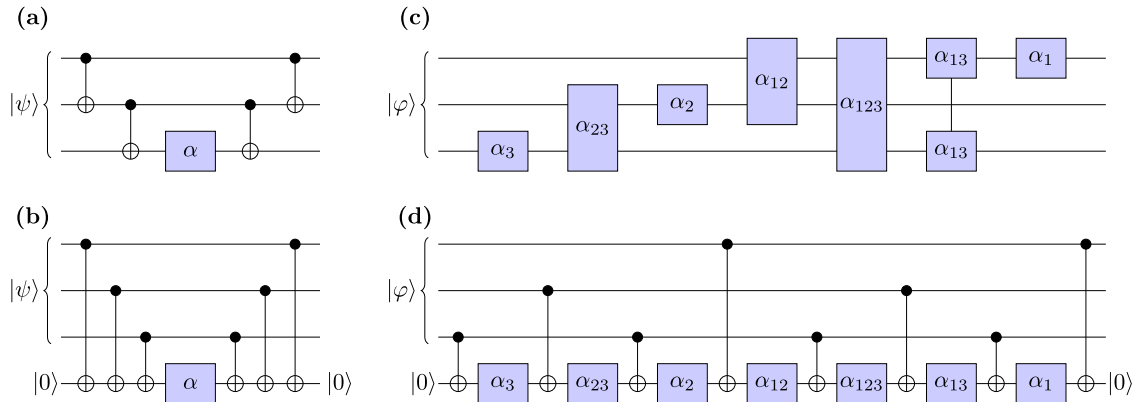

Fig. 3   Decompositions of 3-local Pauli operator ZZZ and higher-local Ising model with and without ancilla. On the left, two decompositions of $\exp(itZ_1Z_2Z_3)$, **a** without auxiliary qubit[41] (p.30), and **b** with auxiliary qubit[42] (p.210). On the right, an example of simplifying circuit for $\sum_{I \subseteq \{1,2,3\}} a_I \prod_{i \in I} Z_i$ using Gray codes ordering for applications of $\prod_{i \in I} Z_i$ gates, **c** using Z operations only, and **d** with CNOTs and single-qubit gates only. In all the figures blue gates are a k-local Z operations.

One can expect that Higher-Order Binary Optimization may lead to difficult landscapes, harder to investigate for optimization algorithm. We have investigated TSP encodings with $W \equiv 0$ and random W matrices. The results are presented in Fig. 4. One can see that with the same number of Hamiltonians applied, the results are either similar or in favor for higher-order encodings.

Mixed QUBO-HOBO approach

While QUBO encoding requires significantly more qubits compared to HOBO, the latter produces much deeper circuits. It is not clear whether the number of qubits or the depth of the circuit will be more challenging, and in fact we claim that both may produce significant difficulties when designing quantum computers.

One can consider a simple mixing of the proposed QUBO and HOBO approaches in the following way: let $R \in \{1, \ldots, N-1\}$ be a free parameter of our model. Exactly R of collections b_t will be encoded as one-hot vectors (in QUBO's fashion), while the remaining $N - R$ collections will be encoded using the binary representation, see Fig. 1d).

Unfortunately, this approach combines flaws of both models introduced before. For $R = \Omega(N)$, the mixed approach requires $\Theta(N^2)$ qubits. On the other hand, for $R = \mathcal{O}(N)$ the depth of the circuit is the same as in the HOBO approach due to numerous HOBO-encoded b_t.

Instead, we propose another encoding. Suppose $N = (2^K - 1)L$ for suitable integers K and L. Each b_t consists of KL qubits of the form b_{tlk}. The cities are encoded as follows. First K qubits (first bunch) decodes numbers $0, \ldots, 2^K - 2$. The second bunch decodes $2^K - 1, \ldots, 2 \cdot 2^K - 3$, and so on. Note that QUBO and HOBO

encodings introduced before are special instances with $L = N$ and $L = 1$ respectively.

We add the following assumptions, which also define H_{valid}. All bits being zero is an invalid assignment, which is equivalent to $\sum_{k,l} b_{tlk} \geq 1$. This can be forced by using standard techniques for transforming inequalities to QUBO[26]. Secondly, if in some bunch there are nonzero bits, then in all other bunches bits have to be zeros. Note that this assumption is equivalent to the fact that for all $l_0 = 0, \ldots, L-1$ we have that either $\sum_k b_{tl_0k}$ is zero or $\sum_{l \neq l_0} \sum_k b_{tlk}$ is zero. The Hamiltonian H_{valid}^{MIX} takes the form

$$H_{valid}^{MIX}(b_t) := \left(-\sum_{l=0}^{L-1}\sum_{k=0}^{K-1} b_{tlk} + 1 + \sum_{i=0}^{\lceil \log(KL) \rceil} 2^i \xi_{t,i} \right)^2$$

$$+ \sum_{l=0}^{L-1} \left(\sum_{k=0}^{K-1} b_{tlk} \right) \left(\sum_{\substack{l'=0 \\ l' \neq l}}^{L-1} \sum_{k=0}^{K-1} b_{tl'k} \right).$$

Here ξ_i are additional bits for encoding the first assumption. In total, there will be additional $N\log(KL) \leq N\log(N)$ qubits.

Now let us define $H_{\neq}^{MIX}$ Hamiltonian. Since due to H_{valid}^{MIX} there exist two indices l_0, l_0' such that b_{tl_0} and $b_{t'l_0'}$ are nonzero, we only have to check for consecutive bunches $l = 0, \ldots, L-1$ if there exists l such that $b_{t,l}$ are nonzero and identical. The Hamiltonian $H_{\neq}^{MIX}$ takes the form

$$H_{\neq}^{MIX}(b_t, b_{t'}) := \sum_{l=0}^{L-1} \left(\sum_{k=0}^{K-1} (b_{tlk} + b_{t',l,k}) \right) \prod_{k=0}^{K-1} \left(1 - (b_{t,l,k} - b_{t',l,k})^2 \right).$$

Note that the first factor checks whether the bunches are nonzero, while the latter is the Kronecker delta implementation as before.
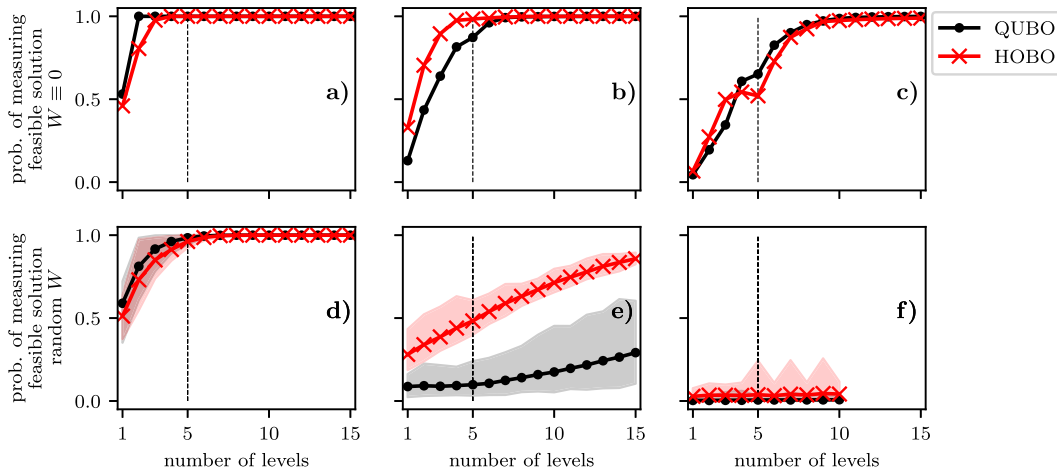
Fig. 4 The dependence between the probability of measuring the state in feasible space and the number of levels for Travelling Salesman Problem. Analysis was done for **a** 3 cities, **b** 4 cities, **c** 5 cities for $W \equiv 0$, and **d** 3 cities, **e** 4 cities, **f** 5 cities for randomly chosen W. For most cases HOBO and QUBO present similar quality, while HOBO clearly outperforms QUBO approach for random W for 4 cities. Vertical line denotes the change of optimization method. For 5 cities with random W we were only capable of estimating up to 10 levels applied due to convergence issues. The area describes the range between the worst and the best cases of the best probabilities over all TSP instances.

Finally, let us define H_δ^{MIX}. Let $\bar{l} : \{0, \dots, N-1\} \to \{0, \dots, L-1\}$ be a function that outputs a bunch index denoting the i-th city. Then it is enough to apply the Kronecker delta on $\bar{l}(i)$-th bunch. Hence H_δ^{MIX} will take the form

$$H_\delta^{\mathrm{MIX}}(b_t, i) := \prod_{k=0}^{K-1} \left(1 - (b_{t', \bar{l}(i), k} - b_k^i)^2 \right),$$

where b_k^i is k-th bit of binary representation of i.

Let us now calculate the efficiency of the encoding. We will consider only the scenario $k = a \log(N)$ for $a \in (0, 1)$. First we note that $L = \frac{N}{2^K - 1} = \Theta(N^{1-a})$. For the proposed mixed encoding, we need $NKL + N\log(KL) = \Theta(N^{2-a} \log(N))$ qubits. Hamiltonian can be encoded in a circuits of depth $\mathcal{O}(N^{1+a} \log^2(N))$. This finally gives us the volume $\mathcal{O}(N^3 \log^3(N))$. All these parameters show a perfect, up to poly-log factors, transition between HOBO and QUBO approaches. Finally, to achieve constant error of energy estimation, we require $\mathcal{O}(N^{3-a} \log N) \max_{i \neq j} W_{ij}$ runs of the circuit.

Optimal encoding

So far we assumed that all binary variables are split into collections of binary variables, such that each collection defines a particular time point. We heavily used this assumption, so that the encoding was particularly simple. Therefore, it was implementable on a quantum computer, which is necessary for QAOA. Nevertheless, dropping this assumption can save us from even more qubits.

Let H be a diagonal Hamiltonian. Then $\langle \psi | H | \psi \rangle = \sum_{b \in \{0,1\}^n} E_b |\langle b | \psi \rangle|^2$, where E_i is the energy value corresponding to the bit string i. Hence, the statistics from the measurements are sufficient to estimate the energy.

Suppose we are given a general combinatorial optimization problem of function $f : \mathcal{A} \to \mathbb{R}$, where $\mathcal{A}$ is a natural feasible space for the problem. In the case of TSP, $\mathcal{A}$ would be a collection of all permutations of some fixed order. Let $g : \mathcal{A} \to \mathcal{B} \subseteq \{0,1\}^n$ be a bijection function where $n = \lceil \log(|\mathcal{A}|) \rceil$. Let $\tilde{g}^{\mathrm{inv}}(b)$ be an extension of g^{-1} such that it maps some penalty energy $E_{\mathrm{pen}} > \min_{a \in \mathcal{A}} f(a)$, i.e. $g^{\mathrm{inv}}(b) = g^{-1}(b) \delta_{b \in \mathcal{B}} + E_{\mathrm{pen}} \delta_{b \notin \mathcal{B}}$. Then provided that g^{inv} can be efficiently computed, we can use it to estimate the expected energy directly from the measurement's statistics. Since converting binary representation into numbers takes negligible time, it is enough to provide a procedure for numbering elements of $\mathcal{A}$.

We can incorporate this technique to TSP problem as well. In this case, the simplest way is to use a factorial numbering system in which i-th digit starting from the least significant one can be any number between 0 and $i - 1$. In general $(d_k \dots d_0)_! \equiv \sum_{i=0}^{k} d_0 \cdot i!$. The opposite transformation can be done by computing the modulo by consecutive natural numbers. Then such representation can be transformed to permutation via Lehmer codes which, starting with the most significant factoradic digit, takes $(k + 1)$-th digit of the sequence $(0, 1, \dots, k)$. The used digit is removed and the procedure repeats for next digit. The taken digits in given order directly encodes a permutation.

Since the procedure described above maps consecutive natural numbers to routes, we require only $\lceil \log(N!) \rceil$ qubits, which is optimal for each N. Since arbitrary pseudo-Boolean function can be transformed to pseudo-Boolean polynomial, it is as well the case for $f \circ g^{\mathrm{inv}}$. Hence there exists a diagonal Hamiltonian representing the same optimization problem. However, in general such encoding may require exponential number of terms, which makes it intractable for QAOA. Hence, for such an approach VQE is at the moment the preferable quantum algorithm. The numbers of required measurements will in general depend on the choice of E_{pen}, however, they can be equal to the length of any route, or $N \max_{i \neq j} W_{ij}$. By this we can show that the number of measurements is approximately $\mathcal{O}(N) \operatorname{range}(W)$, which is significantly smaller than for any encoding described before.

DISCUSSION

The presented results, summarized in Table 1, show that it is possible to significantly reduce the number of required qubits at the cost of having deeper circuits. Since reducing both the depth and the number of qubits are challenging tasks, we claim that it is necessary to provide alternative representation allowing trade-offs between the different measures. Our numerical results hint that the increase of the depth might not be that significant for larger system sizes, as one needs fewer levels in the space-efficient embedded version. Thus, it would be interesting to investigate how many fewer levels one needs in the space-efficient encoding scheme.

Note that the approach cannot be applied for general problems. For example, the state-of-the-art representation of the Max-Cut problem over a graph of order N requires exactly N qubits. Since the natural space in general is of order 2^N, it seems unlikely to further reduce the number of qubits. However, one can expect

Table 1. Upperbounds on resources required for various Hamiltonian encodings.

	QUBO H^{QUBO}	HOBO H^{HOBO}	Mixed H^{MIX}	Enumerating H^{VQE}
No. of qubits	N^2	$N \log N$	$\frac{a}{C} N^{2-a} \log N$	$N \lceil \log(N!) \rceil$
No. of terms	$2N^3$	$(C^2 - C/2)N^3$	$\frac{a}{2} N^3 \log N$	Exponential
Circuit depth	$12N$	$(4C^2 + 3C)N^2$	$2a^2 C N^{1+a} \log^2 N$	Exponential
Circuit volume	$12N^3$	$(4C^2 + 3C)N^3 \log N$	$2a^3 N^3 \log^3 N$	Exponential
No. of measurements	$\mathcal{O}(N^3) \max_{i \neq j} W_{ij}$	$\mathcal{O}(N^2) \max_{i \neq j} W_{ij}$	$\mathcal{O}(N^{3-a} \log N) \max_{i \neq j} W_{ij}$	$\mathcal{O}(N)$range(W)

Only leading terms are written, for more exact bounds see Supplementary Methods. The $\mathcal{O}(\cdot)$ does not depend on the choice of W. We assumed $B = 1$ and $A_1, A_2 = \mathcal{O}(\max_{i \neq j} W_{ij})$. Note that H^{MIX} scales up to poly-logarithmic factor between H^{QUBO} and H^{HOBO}. The constant $C \in (1, 2)$ for HOBO and mixed Hamiltonians depends on a and N. For depth, only terms for even N are present, for odd N terms can be found in the Supplementary Methods.

similar improvements for other permutation-based problems like max-k-coloring problem.

On top of that, while arbitrary HOBO can be turned into QUBO by automatic quadratization techniques, it remains an open question whether there are simple techniques that reduce the number of qubits at the cost of additional Pauli terms. This is due to the fact that quadratization is well-defined: if $H : \mathcal{B}_X \to \mathbb{R}$ is a general pseudo-Boolean function, then quadratic pseudo-Boolean function $H' : \mathcal{B}_X \times \mathcal{B}_Y \to \mathbb{R}$ is its quadratization iff for all $x \in \mathcal{B}_X$

$$H(x) = \min_{y \in \mathcal{B}_Y} H'(x, y). \tag{4}$$

Note that y does not have any meaning in the context of original problem H. However, when removing qubits from binary function, we may not be able to reproduce the original solution. Thus, such (automatic) procedure requires context of the problem being solved.

One could consider recent results showing that QAOA with polynomial depth is exposed to the negative side-effect induced by the noisy system[33,34]. This threatens the possible application of QAOA not only to this model, but also to the state-of-the-art model. However, if we want to compare the efficiency of QAOA with QUBO against HOBO or mixed encoding, we should take into account that smaller systems would likely give more faithful results. This effect is in fact already observed in the current quantum devices: e.g., when comparing superconducting qubits and trapped ion technologies, one can note that while for the former it is much easier to construct larger computers, the latter technology allows us to run deep quantum algorithms with much higher fidelity as long as the memory resources are significantly smaller. Thus, the applicability of our results against the state-of-the-art model may depend on how the current technologies will evolve in the nearest future, and it is difficult to estimate whether small-depth large-width computation will be more plausible than large-depth short-width ones.

Finally, using 2-complete linear swap network presented in O'Gorman et al.[35] we can implement HOBO on Linear Nearest Neighbour (LNN) topology with the same depth up to polylog overhead, by swapping whole registers b_t instead of single qubits. This comes from the fact that swapping two neighboring registers of size $\sim \log N$ requires only $\sim \mathcal{O}(\log N)$ depth. Furthermore, by using decomposition from Fig. 3a and proper reordering of qubits, one can implement up to $\sim 2 \log N$-local Pauli operators with only extra $\mathcal{O}(\log N)$ depth. On the other hand, swap network presented in[35] guarantees only depth proportional to the number of qubits for QUBO. For this reason for the QUBO formulation considered in this paper, it gives only $\mathcal{O}(N^2)$ depth. This in fact is optimal, as one cannot construct swap network achieving depth $N^{2-\varepsilon}$ for any $\varepsilon > 0$. One can show it as follows: if we take any initial arrangement on the LNN network for QUBO, then all of the spins which interact should be with distance $N^{2-\varepsilon}$. However, *all* spins presented in the formulation have to interact with the spins interacting with the first spin (in other words, graph of spin interactions has radius 2

for any node). This would mean that all spins have to be within distance $N^{2-\varepsilon}$ from the first spin, which is not possible since there are N^2 of them. Hence we see that $\Theta(N^2)$ depth is the best achievable for TSP. Hence for such LNN topology HOBO depth is larger only by poly-logarithmic factors, while having significant reduction on qubits.

METHODS

The analysis of circuits' depths

Let us begin with HOBO and mixed approaches. According to Eq. (2) we can split all the terms into those defined over pairs $(b_t, b_{t'})$ for $t \neq t'$. For pairwise different t_0, t_1, t_2, t_3, if we have polynomials defined over b_{t_0}, b_{t_1}, and b_{t_2}, b_{t_3}, then we can implement them independently. Using round-robin schedule, we can implement those $\binom{N}{2}$ polynomials in $N - 1$ (N) steps for even (odd) N, as it is described in Fig. 2.

Let H be a general Hamiltonian defined over K bits. If we implemented each term independently, then it would require $\sum_{k=1}^{K} 2i \binom{K}{i} \Theta(2^K K) = 2^K K - 1$ controlled NOTs according to the decomposition presented in Fig. 3a. Adding single auxiliary qubit and using the decomposition from Fig. 3b, and ordering terms according to Gray code, we can do it using 2^K qubits. Following the reasoning from previous paragraph we can apply only $\lceil \frac{N}{2} \rceil$ Hamiltonians at once. We have an additional cost of $\lceil \frac{N}{2} \rceil$ qubits, however reducing the depth cost by $K = \Theta(\log N)$ for both mixed and simple HOBO approaches.

For $H^{HOBO}_{\neq}$ all terms are formed of pair of spins $Z_{t,k} Z_{t',k}$. Here we can again use the Gray code schedule, however each wire in Fig. 3c has a meaning of the presented pair of spins. This means that each application of Pauli term will results in 2 new controlled NOTs, instead of a single one as in Fig. 3d. The situation for mixed formulation is much more complicated, as there are terms which consist of the mentioned pairs of spins plus one spin w/o match, hence the application of Gray code is not straightforward. Hence we did not assume any CNOT simplification.

As far as QUBO is concerned, we have to make separate analysis, since only 2-local terms are present. Note that 1-local terms Z_{ti} can be implemented with circuit of depth 1. Moreover, for each $0 \le t < N$, terms $\{Z_{ti} Z_{tj} : 0 \le i < j < N\}$ can be implemented with a circuit of depth $\approx N$ using round-robin schedule. We can similarly implement $\{Z_{ti} Z_{t'i} : 0 \le t < t' < N\}$, which implement first two addends of H^{QUBO}. For the last addend we have to implement for each $0 \le t < N$ terms $\mathcal{Z}_t = \{Z_{ti} Z_{t+1j} : i \neq j\}$. For even N, note that we can first implement them for even t, then for odd t, which doubles the depth of the circuit for single t. For odd N we have to include extra layer for $t = N - 1$, which will only increase the depth by 3/2. Finally, note that $\mathcal{Z}_t = \bigcup_{k=0}^{N-1} \mathcal{Z}_{t,k}$, where each $\mathcal{Z}_{t,k} = \{Z_{ti} Z_{t+1,i+k} : 0 \le i < N\}$ can be implemented with circuit of depth 1. Eventually, the depth of Hamiltonian H^{QUBO} is of order $\Theta(N)$.

The detailed analysis for each encoding can be found in Supplementary Methods.

Numerical analysis

Below we describe the optimization algorithm used to generate the result presented in Fig. 4. We have emulated the quantum evolution and take an exact expectation energy of the state during the optimization. As a classical subroutine we used a L-BFGS algorithm implemented in Julia's

Optim package[36]. Independently of the instance of algorithm, we assume that the parameters θ_{mix} for mixing Hamiltonian could be from the interval $[0, \pi]$. For objective Hamiltonian we assumed the parameter θ_{obj} will be from $[0, R]$. For both we assume periodic domain, mainly if $\pi + \varepsilon$ or respectively $R + \varepsilon$ was encountered, the parameter was changed to ε, which changed the hypercube domain to hypertorus.

Let r be the number of levels of the circuit. For $r < 5$, each run was started from randomly chosen vector within range of the parameters. For $r \geq 5$, we used a trajectories method similar to the one proposed in[37]. First, we optimized the algorithm for $r = 5$, as described in previous paragraph. Then, for each $r \geq 5$ we took the optimized parameter vectors $\overrightarrow{\theta}_{mix}^{(r)}$, $\overrightarrow{\theta}_{obj}^{(r)}$ of length r, and constructed new vectors $\overrightarrow{\theta}_{mix}^{(r+1)}$, $\overrightarrow{\theta}_{obj}^{(r+1)}$ of length $r + 1$ by copying the last element at the end. We took these vectors as initial points for $r + 1$. Therefore, we obtained a trajectory of length 11 (6 for Fig. 4f) of locally optimal parameter vectors, one for each $r \geq 5$.

Sometimes the algorithm has not converged to the local optimum in reasonable time. We claim that the reason came from periodicity of the domain, which for general TSP breaks the smoothness of the Hamiltonian landscape. We only accepted runs for which: for $r < 5$ the gradient was below 10^{-5}; for $r \geq 5$, for each parameters vector from the trajectory, the gradient was supposed to be below 10^{-5}.

Since the energy values for both QUBO and HOBO are incomparable, we decided to present the probability of measuring the feasible solution, i.e. the solution describing a valid route.

Figures a, b, c from Fig. 4 were generated as follows. We took QUBO and HOBO encodings of TSP with $W \equiv 0$ for QAOA algorithm. One can consider it as a Hamiltonian problem on a complete graph. We took $A_1 = A_2 = 1$ for both encodings, and $R = \pi$ ($R = 2\pi$) for QUBO (HOBO). For each $r = 1, \ldots, 15$ we generated 100 locally optimal parameter vectors, and for each r we chose the maximum probability.

Subfigures d,e,f from Fig. 4 were generated as follows. We generated 100 matrices W to be $W = X + X^\top$, where X is a random matrix with i. i. d. elements from the uniform distribution over $[0, 1]$. For each TSP instance we repeated the procedure as in $W \equiv 0$ case, however, generating 40 samples for each r. For each TSP instance we chose the maximal probability of measuring the state in the feasible space. The line describes the mean of the best probabilities over all TSP instances. The area describes the range between the worst and the best cases of the best probabilities over all TSP instances.

The number of measurements
For estimating the number of measurements we applied the Hoeffding's inequality. Let $\overline{X} = \frac{1}{M}\sum_{i=1}^{M} X_i$ be the mean of i.i.d. random variables such that $X_i \in [a, b]$. Then

$$\mathbb{P}(|\overline{X} - \mathbb{E}\overline{X}| \geq t) \leq 2\exp\left(-\frac{2Mt^2}{b-a}\right). \tag{5}$$

In our case $\overline{X}$ is the energy estimation of the energy samples X_i. Provided that we expect both probability error and estimation error to be constant, we require $M = \Omega(b - a)$.

The values of a, b depend not only on the cost matrix W, but also on the form of the encoding and the values of constants A_1, A_2, B in Eq. (2). For simplicity, we take the following assumptions when estimating the samples number. First, w.l.o.g. we assume $B = 1$. Furthermore, we assumed $C \max_{i \neq j} W_{ij} \leq A_i \leq C' \max_{i \neq j} W_{ij}$ where C, C' do not depend on N and W. This matches the minimal requirement for QUBO. While various measures on W could be considered, we presented the results in the form $f(N)\max_{i \neq j} W_{i,j}$, where f does not depend on W. Furthermore, our analysis for each model is tight in N assuming that $\max_{i \neq j} W_{ij}, \min_{i \neq j} W_{ij} = \Theta(1)$ independently on N. Note that using this assumption, $a \geq N \min_{i \neq j} W_{ij}$ is valid for any correctly chosen A_1, A_2.

Furthermore, Hamiltonians $H_{\neq}$ and H_{valid} are integer-valued, and the spectral gap is of constant order. For QUBO, the spectral gap is at most two, which can be generated by adding superfluous one-bit to any valid encoding. For HOBO, it can be generated by choosing the same number for different $b_t, b_{t'}$. Finally, for the mixed approach we can generate incorrect assignments to slack $\xi_{t,i}$ variables. Theoretically, there is no upper-bound for A_1, A_2. However, in general it is not encouraged to make them too large, as classical optimization algorithm may focus too much on pushing the quantum state to feasible state instead of optimizing over feasible space. For this reason and to make the presentation of our results simpler, we assumed that A_i are of order $\max_{i \neq j} W_{i,j}$.

During the optimization, one could expect that the quantum states will finally have large, expectedly $1 - o(1)$ overlap with the feasible space. Thus, one could expect that the estimated energy will be of typical, and later close to minimal, route. Thus, for these points one could expect that $\mathcal{O}(N)$ range(W) samples would be enough to estimate the energy accurately. We agree that it is a valid approach, especially when the gradient is calculated using $(f(\theta_0 + \varepsilon) - f(\theta_0 - \varepsilon))/(2\varepsilon)$ formula. However, recently a huge and justified effort has been made on analytical gradient estimation, which is calculated based on θ parameters that are far from the current θ_0 point[38,39]. In this scenario, we can no longer assume that the energy will be of the order of the typical route. Thus we believe that our approach for number of measurements estimation is justified.

The detailed analysis for each encoding can be found in the Supplementary Methods.

REFERENCES
1. Grover, L. K. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, 212-219 (ACM Digital Library, 1996).
2. Shor, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.* **41**, 303–332 (1999).
3. Montanaro, A. Quantum algorithms: an overview. *npj Quant. Inf.* **2**, 1–8 (2016).
4. Jordan, S. Quantum Algorithm Zoo. quantumalgorithmzoo.org/ https://quantumalgorithmzoo.org/.
5. Arute, F. et al. Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505–510 (2019).
6. Montanaro, A. Quantum-walk speedup of backtracking algorithms. *Theory Comput.* **14**, 1–24 (2018).
7. Ambainis, A. et al. Quantum speedups for exponential-time dynamic programming algorithms. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, 1783-1793 (SIAM, 2019).
8. Peruzzo, A. et al. A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **5**, 4213 (2014).
9. McClean, J. R., Romero, J., Babbush, R. & Aspuru-Guzik, A. The theory of variational hybrid quantum-classical algorithms. *New J. Phys.* **18**, 023023 (2016).
10. Moll, N. et al. Quantum optimization using variational algorithms on near-term quantum devices. *Quant. Sci. Technol.* **3**, 030503 (2018).
11. Farhi, E., Goldstone, J. & Gutmann, S. A quantum approximate optimization algorithm. Preprint at http://arXiv.org/abs/1411.4028 (2014).
12. Ostaszewski, M., Grant, E. & Benedetti, M. Structure optimization for parameterized quantum circuits. *Quantum* **5**, 391 (2021).
13. Kübler, J. M., Arrasmith, A., Cincio, L. & Coles, P. J. An adaptive optimizer for measurement-frugal variational algorithms. *Quantum* **4**, 263 (2020).
14. Choi, V. Minor-embedding in adiabatic quantum computation: I. the parameter setting problem. *Quant. Inf. Process.* **7**, 193–209 (2008).
15. Choi, V. Minor-embedding in adiabatic quantum computation: Ii. minor-universal graph design. *Quant. Inf. Process.* **10**, 343–353 (2011).
16. Morales, M. E., Biamonte, J. & Zimborás, Z. On the universality of the Quantum Approximate Optimization Algorithm. *Quant. Inf. Process.* **19**, 1–26 (2020).
17. Lloyd, S. Quantum approximate optimization is computationally universal. Preprint at http://arXiv.org/abs/1812.11075 (2018).
18. Hastings, M. B. Classical and quantum bounded depth approximation algorithms. Preprint at http://arXiv.org/abs/1905.07047 (2019).
19. Farhi, E., Gamarnik, D. & Gutmann, S. The quantum approximate optimization algorithm needs to see the whole graph: Worst case examples. Preprint at http://arXiv.org/abs/2005.08747 (2020).
20. Hadfield, S. et al. From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms* **12**, 34 (2019).

21. Akshay, V., Philathong, H., Morales, M. E. & Biamonte, J. D. Reachability deficits in quantum approximate optimization. *Phy. Rev. Lett.* **124**, 090504 (2020).

22. Zhu, L. et al. An adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer. Preprint at http://arXiv.org/abs/2005.10258 (2020).

23. Zhao, T., Carleo, G., Stokes, J. & Veerapaneni, S. Natural evolution strategies and variational Monte Carlo. *Mach. Learn. Sci. Technol.* **2**, 02LT01 (2021).

24. Wierichs, D., Gogolin, C. & Kastoryano, M. Avoiding local minima in variational quantum eigensolvers with the natural gradient optimizer. *Phys. Rev. Res.* **2**, 043246 (2020).

25. Boros, E. & Gruber, A. On quadratization of pseudo-Boolean functions. In *International Symposium on Artificial Intelligence and Mathematics, ISAIM 2012* (2012).

26. Lucas, A. Ising formulations of many NP problems. *Front. Phys.* **2**, 5 (2014).

27. Anschuetz, E., Olson, J., Aspuru-Guzik, A. & Cao, Y. Variational quantum factoring. In *International Workshop on Quantum Technology and Optimization Problems*, 74–85 (Springer, 2019).

28. Tan, B., Lemonde, M.-A., Thanasilp, S., Tangpanitanon, J. & Angelakis, D. G. Qubit-efficient encoding schemes for binary optimisation problems. *Quantum* **5**, 454 (2021).

29. Mandal, A., Roy, A., Upadhyay, S. & Ushijima-Mwesigwa, H. Compressed quadratization of higher order binary optimization problems. In *Proceedings of the 17th ACM International Conference on Computing Frontiers*, 126–131 (2020).

30. Sawaya, N. P. et al. Resource-efficient digital quantum simulation of *d*-level systems for photonic, vibrational, and spin-*s* Hamiltonians. *npj Quant. Inf.* **6**, 1–13 (2020).

31. Fuchs, F. G., Kolden, H. Ø., Aase, N. H. & Sartor, G. Efficient encoding of the weighted MAX-*k*-CUT on a quantum computer using QAOA. *SN Comput. Sci.* **2**, 1–14 (2021).

32. Bärtschi, A. & Eidenbenz, S. Grover mixers for QAOA: Shifting complexity from mixer design to state preparation. In *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 72–82 (IEEE, 2020).

33. Wang, S. et al. Noise-induced barren plateaus in variational quantum algorithms. *Nat. Commun.* **12**, 1–11 (2021).

34. Stilck França, D. & García-Patrón, R. Limitations of optimization algorithms on noisy quantum devices. *Nat. Phys.* **17**, 1221–1227 (2021).

35. O'Gorman, B., Huggins, W. J., Rieffel, E. G. & Whaley, K. B. Generalized swap networks for near-term quantum computing. Preprint at http://arXiv.org/abs/1905.05118 (2019).

36. Mogensen, P. K. & Riseth, A. N. Optim: a mathematical optimization package for Julia. *J. Open Source Softw.* **3**, 615 (2018).

37. 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**, 021067 (2020).

38. Schuld, M., Bergholm, V., Gogolin, C., Izaac, J. & Killoran, N. Evaluating analytic gradients on quantum hardware. *Phys. Rev. A* **99**, 032331 (2019).

39. Sweke, R. et al. Stochastic gradient descent for hybrid quantum-classical optimization. *Quantum* **4**, 314 (2020).

40. Rasmussen, R. V. & Trick, M. A. Round robin scheduling – a survey. *Eur. J. Oper. Res.* **188**, 617–636 (2008).

41. Seeley, J. T., Richard, M. J. & Love, P. J. The Bravyi-Kitaev transformation for quantum computation of electronic structure. *J. Chem. Phys.* **137**, 224109 (2012).

42. Nielsen, M. A. & Chuang, I. L. Quantum computation and quantum information. *Phys. Today* **54**, 60–2 (2001).

AUTHOR CONTRIBUTIONS

All the authors have equal contribution to the preparation of the paper. A.G. proposed the main idea, made the analytical investigations presented in Supplementary Methods and performed the numerical analysis. A.K. prepared the figures, studied the problem of effective implementation on quantum computers and designed and implemented the first mixed encoding. Z.Z. proposed and investigated the second mixed encoding and quantified the impact of the paper.

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/s41534-022-00546-y.

Correspondence and requests for materials should be addressed to Adam Glos.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.