A Gradient-Cost Multiobjective Alternate Framework for Variational Quantum Eigensolver with Variable Ansatz

Ze-Tong Li, Fan-Xu Meng, Han Zeng, Zhai-Rui Gong, Zai-Chen Zhang, and Xu-Tao Yu*

Variational quantum eigensolver (VQE), aiming at determining the ground state energy of a quantum system described by a Hamiltonian on noisy intermediate scale quantum (NISQ) devices. However, the accuracy and trainability of the current VQE algorithm are significantly influenced due to the 

**barren plateau** (BP), the non-negligible gate error and limited coherence time in NISQ devices. To tackle these issues, a gradient-cost multiobjective alternate framework with variable ansatz is proposed. A theoretical framework is first proposed for VQE with variable ansatz (VA-VQE) via alternately solving a multiobjective optimization problem defined by cost function values and gradient magnitudes and the original VQE problem. Then, a novel implementation method based on the double $\epsilon$-greedy strategy with the candidate tree and modified multiobjective genetic algorithm is proposed. As a result, the local optima are avoided both in ansatz and parameter perspectives, the BP phenomenon is alleviated, and the stability of output ansatz is enhanced. The experimental results indicate that this framework shows considerably average improvement of the error and stability by 59.6% and 78.8% compared with the state-of-the-art cost-criterion-based VA-VQE implementation, and 54.1% and 73.5% compared with the gradient-criterion-based counterpart, respectively, with lower quantum costs.

1. Introduction

Quantum computing, a promising paradigm for solving many classically intractable problems, is confronted with the era of noisy intermediate scale quantum (NISQ). NISQ devices, characterized as tiny quantities and short coherent time of qubits and low fidelity of quantum gates, seem difficult to efficiently apply the extraordinary quantum algorithms.

A hybrid quantum-classical scheme kindles the hope of extracting the quantum advantages from NISQ devices. In this scheme, the quantum computer maintains a parameterized quantum circuit (aka. ansatz), and the classical computer runs an optimizer to find the parameters that minimize the cost function of the ansatz. This novel scheme was first proposed as the variational quantum eigensolver (VQE) to find the ground-state energy of a molecule. For this specific task, one commonly used ansatz is unitary coupled-cluster singles and doubles (UCCSD), derived from the coupled cluster theory in chemistry. However, UCCSD sometimes results in deep circuit and a large number of parameters to optimize, which depends on the task and the specifics of the implementation. Then, a kind of problem insensitive ansatz named hardware efficient ansatz is proposed. The elaborately engineered ansatz is implemented with fixed layer structure and adjustable cardinality of layers. Each layer of it is constructed by native implementable gates with short circuit depth. The hardware efficient ansatz have impressively exhibited high expressibility, the ability to express the extensive range of quantum states. Unraveling more potentiality of VQE, it accelerates the generalization of VQE for solving linear systems, simulating dynamics, decomposing matrix, reducing the dimensionality of data, and solving problems in the data science domain. In the following text, the term VQE is used to indicate all methods that adopt the aforementioned scheme.

The VQE has experimentally shown effectiveness in small-scale problems. However, the performance of VQE degrades significantly with the qubit number and circuit depth. The phenomena of the degradation are generally exhibited as the non-ignorable absolute error between the converged cost function value and the exact minimum. One reason for the degradation is that the VQE suffers from the so-called barren plateau that the gradient magnitudes vanish exponentially with the system...
scale. Then, the severity of the BP phenomenon is linked to the number of entanglements and the expressibility. Another reason is quantum hardware noise both in quantum gates and qubits. The influence of the noise accumulates with the number of quantum gates and the circuit depth and causes the accuracy reduction of the observable estimation. This significantly impedes the convergence of the cost function since the value of the cost function and its gradient are inaccurate, which is known as noise-induced barren plateaus.

Recently, the variable ansatz strategy has emerged as a promising technique to reserve the quantum advantage in large-scale situations. Unlike the structure fixed ansatz, the layer constraint is relaxed in variable ansatz, and gates can be added anywhere in the ansatz as required. Evolved from UCCSD, this scheme was first proposed for exact molecular simulations as ADAPT-VQE and qubit-ADAPT-VQE. Operators generated from the coupled cluster theory are inserted in the circuit based on the gradient magnitude. Then, several flexible strategies evolving from the hardware efficient ansatz are proposed for more general tasks. As a result, both the size and the depth of the constructed ansatz can be effectively reduced, and thus better solutions can be found. Nevertheless, implementing the VQE with variable ansatz (VA-VQE) exists a significant space overhead for the ansatz layout optimization procedure, and consequently requires substantially more computing resources than that with the structure fixed ansatz.

To further improve efficiency, several techniques from intelligent algorithms have also been introduced. Several methods based on the complete training are proposed. In these methods, the performance estimation of an ansatz requires VQE training until the cost function converges to obtain the lowest cost function value of the ansatz. As a result, high precision of the optimal selection among sampled ansätze is achieved but undesirably introduces tremendous cost function determinations which are time-consuming because each determination of cost function requires substantial quantum measurements. Therefore, the range of sampled ansätze is narrow when the quantity of quantum measurements is limited. This may result in local optimum from the ansatz perspective.

Attempting to overcome this hurdle, several methods based on the highly shared parameter pool are proposed. These methods construct ansätze layer-wisely and maintain parameter pools from which values of parameters are exploited to calculate the cost function value as the performance predictor. Introducing the weight-sharing policy, parameters are shared among layers with similar structures. Thus, the number of parameters in the parameter pool required to be trained is effectively reduced. These methods first train the parameter pools iteratively. In each iteration, several ansätze are sampled based on the sampling strategy. Then, the parameters corresponding to the ansätze are updated in specified steps (especially one step) by an optimizer. After training the parameter pools, the optimal ansatz selection procedure is processed. Quantities of ansätze are sampled and evaluated by the performance predictor. Finally, the ansatz with the best performance, generally the lowest cost function value, is selected as the optimal ansatz.

This kind of method gains the efficiency in finding optimal ansätze by avoiding complete training of ansätze. However, since the parameters are highly shared and the training step of each ansatz is insufficient to converge, the current cost function values may not express the actual performances of the ansätze, and hence the output suffers from high variance. This may result from two issues. The first is the training competitions that values of parameters in “good” ansätze with promising layouts may be disturbed by the training of “bad” ansätze incurred by the weight-sharing policy. The second issue is that the policy for ansatz layout optimization exploits the cost function value only as the benchmark to find the optimal ansatz structure. The mediocre ansatz whose cost function value is currently the lowest and cannot be further decreased may be selected as the optimum. Contrarily, the promising ansätze may be discarded because of their temporarily high cost function values. Consequently, the local optimum from the ansatz perspective still holds.

This challenge exists and can be restated to a completely opposite version by simply exploiting the criteria proposed in refs. where a layout is determined by the greedy selection based on the maximum gradient magnitude. It can be easily understood as that the methods always selects the layout of ansatz with the maximum gradient magnitude without considering cost function values. Ref. tried to implement the VA-VQE by solving a multiobjective optimization problem by non-dominated sorting genetic algorithm II (NSGA-II) to escape the local optimum. The objects are set as the cost function value and the circuit depth so that the outputs are generally with low cost function value and shallow depth. However, the transformed multiobjective optimization problem may not be equivalent to the original VA-VQE problem, because not all Pareto optimal ansätze are solutions of the original problem when the NSGA-II converged. For example, the ansatz with the prespecified lower-bound depth will always be one of the Pareto optimal ansätze.

In this paper, addressing these issues, we propose a gradient-cost multiobjective alternate framework (GCMA) for VA-VQE. The GCMA alternately optimizes the structure of the ansatz and correspondingParameters by comprehensively considering the cost function values and gradient magnitudes. Contributions are listed below:

i) We first propose a theoretical framework for VA-VQE via alternately solving a multiobjective optimization problem of the cost function and its gradient magnitudes and the original VQE problem, so that local optima can be avoided from the ansatz perspective and the stability is enhanced compared to other VA-VQE methods.

ii) To mitigate training competitions in training the parameter pool for the initialization of parameters, we first exploit the double $\epsilon$-greedy strategy based on the candidate tree to differentiate ansätze by their cost function values and the gradient magnitudes, so that local optima can be evaded from the parameter perspective.

iii) We further reduce the size of the search space of the ansatz via applying gate commutation rules and establishing an injection from the search space to practical implementations of ansätze to boost the time efficiency of the optimal ansatz and parameter determination.

iv) We adopt relatively fair criteria for measuring the performance of VA-VQE so that the transverse comparison among methods of VA-VQE can be conducted. In these results, our method shows conspicuously better performance in terms of...
of error, quantum cost (the invoking times of calculations of the cost function), and stability (mean square error) on average compared with the state-of-the-art cost-criterion-based method \cite{19} by 59.6%, 27.6%, and 78.8% improvement, respectively, and gradient-criterion-based method \cite{28} by 54.1%, 10.2%, and 73.5% improvement, respectively. Moreover, the GCMA exhibits mitigated BP phenomenon and relatively low expressibility with best accuracy.

This paper is structured as follows: In Section 2, we give a brief introduction to the basic knowledge. In Section 3, the theoretical gradient-cost multiobjective alternate framework is proposed. The details of the practical implementation of GCMA is proposed in Section 4. Subsequently, we conduct numerical simulations with relatively fair criteria in Section 5 to show the advantages of our proposed method. Finally, we conclude this work in Section 6. Note that the examples and pseudocodes of algorithms are summarized in Sections SA and SE, Supporting Information, respectively.

2. Background

2.1. Variational Quantum Eigensolver

In this work, we address the VQE tasks identified by sets of tuples \( T = \{(O_i, \rho_i, f_i)\} \) aiming at minimizing a cost function

\[
C(\theta) = \sum_i f_i \left( \text{Tr}[O_i U(\theta) \rho_i U^\dagger(\theta)] \right) \tag{1}
\]

where \( \{\rho_i\} \) is a set of \( n \)-qubit quantum states, \( U(\theta) \) is a specified parameterized quantum circuit (aka. ansatz) with parameters \( \theta \), \( O_i \) are observables and \( f_i \) are bounded second-order differentiable functions that encode the problem with respect to the operand observable \( O_i \) and state \( \rho_i \). Generally, while implementing the VQE, the quantum computer applies the ansatz \( U(\theta) \) and processes the quantum measurements to determine expectations \( \text{Tr}[O_i U(\theta) \rho_i U^\dagger(\theta)] \), \( V_i \). The classical computer computes the cost function value and runs an optimization algorithm to find the optimal parameters \( \theta^* \) that minimize the cost function. We provide an example finding the ground state energy of a quantum system described by a Hamiltonian in Example 1.

2.2. Quantum Gradient

In the optimization procedure of large-scale VQE, the gradient-based methods (e.g., gradient descent) are generally more preferred than gradient-free methods (e.g., Nelder-Mead) \cite{37}. In the gradient-based methods, the gradient of the cost function with respect to the parameters is essentially estimated in each optimization iteration.

Without loss of generality, an ansatz can be mathematically defined by

\[
U(\theta) := \prod_{l=1}^{N_l} U_l(\theta) W_l \tag{2}
\]

where \( U_l(\theta) = \exp(-i \theta_l V_l) \), \( V_l \) is a Hermitian operator, and \( W_l \) is a non-parametrized quantum gate. Then, the partial derivative of an expectation \( E(\theta) = \text{Tr}(\mathcal{O}_l U(\theta) \rho U^\dagger(\theta)) \) with respect to the \( k \)th parameter \( \theta_k \) is

\[
\partial_k E(\theta) \equiv \frac{\partial E(\theta)}{\partial \theta_k} = i \text{Tr} \left( \left[ V_k, U_l^\dagger O_l U_l \right] U_k \rho U_k^\dagger \right) \tag{3}
\]

where we use the notations

\[
U_k \equiv \prod_{l=1}^{k-1} U_l(\theta) W_l, \quad U_l \equiv \prod_{l=k}^{N_l} U_l(\theta) W_l \tag{4}
\]

In this paper, we apply the parameter-shift rule \cite{37} for gradient estimating. We assume that all parameterized quantum gates are Pauli rotations. Therefore, the partial derivative of \( E(\theta) \) with respect to \( \theta_k \) is

\[
\partial_k E(\theta) = \frac{1}{2} \left[ E(\theta + \frac{\pi}{2} \epsilon_k) - E(\theta - \frac{\pi}{2} \epsilon_k) \right] \tag{5}
\]

where \( \epsilon_k \) is a vector whose \( k \)th element is 1 and others are 0.

2.3. Gradient Descent

Gradient descent is literally a gradient-based optimizer that has been generally used in training VQE \cite{10, 11, 14, 25}. The kernel process of the gradient descent can be mathematically represented as

\[
\theta \leftarrow \theta - \alpha \nabla C(\theta) \tag{6}
\]

where \( \alpha \) is the step size.

In this paper, we exploit a line search to estimate the step size \( \alpha \) instead of a fixed one. In each optimization step, \( \alpha \) satisfies the Wolfe conditions \cite{38}

\[
C(\theta - \alpha g) \leq C(\theta) - c_1 \alpha \|g\|^2 \tag{7}
\]

\[
\nabla C(\theta - \alpha g)^T g \geq c_2 \|g\|^2 \tag{8}
\]

where \( g = \nabla C(\theta) \) and \( 0 < c_1 < c_2 < 1 \). To facilitate the \( \alpha \) determination, given a reference step size \( \alpha_0 \), we gradually decrease \( \alpha \) from \( \alpha_0 \) by repeating \( \alpha \leftarrow \rho \alpha \) until Equation (7) establishes, where \( \rho < 0 \). In this paper, we empirically set \( c_1 = 10^{-4} \) and \( \rho = 0.618 \).

2.4. Expressibility

In the absence of prior knowledge about the solution unitaries \( U_i \) of a VQE task, the ability of ansatz to generate a wide range of unitaries \( U \) to guarantee \( U_i \cap U \neq \emptyset \) is required. The expressibility of an ansatz describes the degree to which it uniformly explores the unitary group \( U(2^n) \), and can be simply considered as the range of unitaries the ansatz can generate. By comparing the uniform distribution of unitaries obtained from \( U \) to the Haar distribution of unitaries from \( U \), the expressibility of an ansatz can be defined by the superoperator \cite{39, 40, 41}

\[
A^{(1)}_U := \int_{U(2^n)} d\mu(V) V^{\otimes l} (\cdot) (V^{\dagger})^{\otimes l} - \int_{U} dU U^{\otimes l} (\cdot) (U^{\dagger})^{\otimes l}, \tag{9}
\]
where $d\mu(V)$ is the volume element of the Haar measure and $dU$ is the volume element corresponding to the uniform distribution over $U$. Here we are especially interested in the expressibility of the ansatz with respect to the input quantum state $\rho$ and the observable $O$

$$e_{U}^{\rho} := \left\| A_{U}^{(\rho)} - A_{U}^{(O)} \right\|_2, \quad e_{U}^{O} := \left\| A_{U}^{(O)} - A_{U}^{(O^2)} \right\|_2$$

Small values of $e_{U}^{\rho}$ and $e_{U}^{O}$ indicate the high expressibility of the ansatz.

### 2.5. Barren Plateau and Trainability

As one of the key challenges of VQE, the barren plateau phenomenon exhibited the exponential decrement of the gradient magnitudes with respect to the system size $n$\(^{[20]}\) and was generalized that the ansatzes' expressibility\(^{[25]}\) and the amount of entanglement\(^{[21–24]}\) play significant roles in leading to barren plateaus. We highlight the severity of the BP phenomenon with $n$ and the expressibility of the ansatz by the limited variance of gradient magnitudes

$$\text{Var}[\partial_\theta C(\theta)] \leq g(\rho, O, U) \frac{2n}{2^n - 1} + f\left(e_{U}^{\rho}, e_{U}^{O}\right)$$

where $g(\rho, O, U)$ is the prefactor in $O(2^n)$,

$$f(x, y) = 4xy + \frac{2^{n+1}(x\|O\|_2^2 + y\|\rho\|_2^2)}{2^n - 1}$$

and $U_1$ and $U_2$ are ensembles of $U_1$ and $U_2$, respectively.\(^{[25]}\)

The first term on the right in Equation (11) indicates the variance of 2-design ansatz and is in $O(1/2^n)$, and the second term is the expressibility-dependent correction. From the Chebyshev's inequality, the trainability of an ansatz can be described by

$$\text{Pr}\left[|\partial_\theta C(\theta)| \geq \delta\right] \leq \frac{\text{Var}[\partial_\theta C(\theta)]}{\delta^2}, \forall \delta > 0$$

When the ansatz exhibits the BP phenomenon, the probability decreases exponentially with respect to $n$, which indicates that the precision to determine a cost-minimizing direction is exponentially large to $n$.\(^{[12,41,42]}\) Notably, an ansatz with higher expressibility suffers lower trainability since it exhibits a more severe BP phenomenon.

Remarkably, there exists another kind of barren plateaus, the noise-induced barren plateaus\(^{[20]}\) caused by the imperfect quantum hardware. The cost function value concentrates exponentially around its average as the influence of noise accumulates, since the noise models acting throughout the ansatz map the input state toward the fixed point of the noise model.\(^{[26,43]}\)

This challenging phenomenon cannot simply be escaped by changing the optimizer.\(^{[42]}\) While several attempts have been made to mitigate the severity of the barren plateau,\(^{[44–48]}\) it is widely accepted that the variable ansatz strategy is promising to address this issue via automatically balancing the expressibility, the influence of noise, and the trainability.

### 2.6. Hardware Constraints

Generally, only a limited number of gates are available on a practical quantum computer and the two-qubit gates are only allowed to be applied on a specific set of qubit pairs. These available gates are known as native gates of the quantum hardware. In this paper, we assume the native gates on an $n$-qubit quantum system to be $R_y$, $R_z$, and CNOT mathematically represented as

$$R_y(\theta) = e^{-i\theta/2} \begin{bmatrix} \cos \theta/2 & -\sin \theta/2 \\ \sin \theta/2 & \cos \theta/2 \end{bmatrix}$$

$$R_z(\theta) = e^{-i\theta/2} \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}$$

$$\text{CNOT}^{n \times n} = \ket{0}\bra{0} \otimes I^{n} + \ket{1}\bra{1} \otimes X^{n}$$

where superscript $q$, $q_x$, and $q_y$ indicate the qubits on which the quantum gates act, $\sigma_i$ and $\sigma_z$ are Pauli operators mathematically represented as

$$\sigma_i = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$I$ is the identity quantum operation and $X$ is the quantum not gate mathematically represented as

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Moreover, $R_y^2$ and $R_z^2$ can be applied for all $q \in \{1, 2, \ldots, n\}$, while CNOT gates $\text{CNOT}^{n \times n}$ are unidirectionally available on adjacent qubit pairs, that is, $q_b = (q_a + 1) \mod n$. Note that the proposed framework can be easily adjusted for different native gate sets.

#### 3. Theoretical Framework

In this section, we introduce the theoretical framework derived from the problem of VQE. Then, we equivalently transform the solving procedure into solving a series of multiobjective optimization problems with respect to the cost function value and gradient magnitude.

Recall that a VQE task can be considered as minimizing a cost function, which is summarized in Problem 1.

**Problem 1 (VQE).** Given an ansatz $U(\theta)$ and a task $T$, the problem of VQE is to find parameters that minimize the cost function $C(\theta)$, that is,

$$\min_{\theta \in \mathbb{R}^{n_{\theta}}} C(\theta) = \sum_{i=1}^{D} f_i(\text{Tr}[O_i U(\theta) \rho, U_i(\theta)])$$

where $D \subseteq \mathbb{R}$, $n_{\theta}$ is the cardinality of trainable parameters $\theta$.

While introducing the VA-VQE framework as summarized in Problem 2, the structure of ansatz $U$ is treated as a variable that needs to be optimized in the cost function

$$C(U, \theta) = \sum_{i=1}^{D} f_i(\text{Tr}[O_i U(\theta) \rho, U_i(\theta)])$$
Problem 2 (VA-VQE). Given a search space of ansatz $S$ and a task $T$, the problem of VA-VQE is to find ansatzes and corresponding parameters that minimize the cost function $C(U, \theta)$, that is,

$$\min_{\theta \in \Theta} C(U, \theta)$$

where $C(U, \theta)$ forms as Equation (20), $\Theta \subset \mathbb{R}$, and $N_{\theta}$ is the cardinality of trainable parameters $\theta$.

Practically, the VA-VQE methods automatically construct ansatzes by quantum gates from a given gate set $G$, that is, $S = G^\infty$, and find ones with trained parameters that minimize the cost function. The search space of ansatz actually scales infinitely since the depth of ansatz circuits can be infinitely large. However, the trainability of the ansatz is substantially limited by the number of quantum gates because the expressibility and the impact of noise of the ansatz may increase with respect to the number of quantum gates, which aggravates the BP phenomenon. On the other hand, it is intractable to search for solutions with enormous (even infinite) search space of ansatz. Therefore, a fixed or gradually increased maximum number $n_g$ of quantum gates is introduced as $S = G^{n_g}$, where $N_{\theta}$ is sufficiently large that $\exists U \in G^{n_g}$ such that $U \cap U_i \neq \emptyset$. Nevertheless, determining exact solutions requires solving combinatorial optimization in an exponentially large search space, which conceals the efficiency of VQE. Most methods[19,30,34,35] attempt to provide approximate solutions in polynomial complexity with respect to $g$. Although the gradient is generally estimated for the optimization of parameters, its magnitude is generally neglected to provide a guideline for the optimization of the structure of ansatz. Therefore, several undesirable phenomena listed below may occur:

1) The ansatz which exhibits severe barren plateau, and thus the gradient magnitude is highly close to 0, is selected to test the cost function value multiple times.

2) The trivial ansatz whose parameters are completely trained, that is, the gradient magnitude is highly close to 0, is selected as the optimal ansatz since it temporarily cost less than potentially better ansatz with incompletely trained parameters, that is, the gradient magnitude is significantly larger than zero.

On the contrary, simply exploiting the greedy selection based on the maximum gradient magnitude[27,28] may result in that a trivial ansatz whose parameters are not completely trained and thus has the maximum gradient magnitude is selected to test the cost function value multiple times when the optimal ansatz has been completely trained with 0 gradient magnitude. This chosen challenge is especially severe and result in the termination difficulty in the hardware-efficient scheme since the search space is exponentially large (c.f. the layer-like distribution of results of ADA in Figure 8). These phenomena result in the local optimal of the VA-VQE from the ansatz perspective and instability of optimal ansatz and cost function value outputs.

Here comprehensively consider the cost function value and gradient magnitude in solving the VA-VQE problem by defining a multobjective problem. The cost function value reflects the local performance of an ansatz. The gradient magnitude reflects the decreasing potential of the cost and explicitly supervises the sufficiency of expressibility, the severity of the BP phenomenon, and the completeness of parameter training of ansatzes. We define a subproblem the gradient-related ansatz multiobjective optimization (GRAMO) in Problem 3.

Problem 3 (GRAMO). Given a search space of ansatzes $S$, a task $T$, and a set of parameters $\mathcal{P}$ such that $\exists \theta_j \in \mathbb{P}, \forall U \in S$, the problem of GRAMO is to find ansatzes that minimize the cost function $C(U, \theta_j)$ and maximize the gradient magnitude of the cost function, that is,

$$\min_{U \in S} C(U, \theta_j), \max_{U \in S} \frac{\|VC(U, \theta_j)\|_2}{|\theta_j|}$$

where $C(U, \theta_j)$ forms as Equation (20), $VC(U, \theta_j)$ is the gradient of $C(U, \theta_j)$ with respect to $\theta_j$. $U \in S$, and $|\theta_j|$ is the cardinality of parameters in $\theta_j$.

After estimating cost function values and gradients for each ansatz in $S$ based on the parameters in $\mathcal{P}$, the (1-rank) solution of Problem 3 is defined as the non-dominated set $U_\circ$ consisting of all $U$ such that no $V \in S$ simultaneously establishes inequalities

$C(V, \theta_j) < C(U, \theta_j)$

$$\frac{\|VC(V, \theta_j)\|_2}{|\theta_j|} \geq \frac{\|VC(U, \theta_j)\|_2}{|\theta_j|}$$

As a result, structures of ansatzes with low cost function values or large gradient magnitudes, in other words, high absolute performance or high potentiality, are selected as solutions. Furthermore, the $k$-rank solution $U_{\circ}(k)$ of the problem is defined by $U_{\circ}(k) := U_{\circ}(k-1) \cap U'_{\circ}$, where $U'_{\circ}$ is the 1-rank solution of Problem 3 with search space $S \setminus U_{\circ}(k-1)$.

We solve the VA-VQE problem by alternately solving Problem 1 and Problem 3 as summarized in Algorithm 1. When the gate set $G$ and the task $T$ are specified, given $\mathbb{P}_0$, an initialization of $\mathcal{P}$, the VA-VQE can be solved iteratively. Note that the constraint of $\mathbb{P}_0$ is quite mild. The requirement is that there exists an optimal ansatz $V \in S$ and $\theta_v \in \mathbb{P}_0$ such that $\theta_v$ is not the local maximum point, which can even be readily satisfied by uniformly random initialization. At each iteration, the non-dominated set $U_{\circ}$ is determined by solving Problem 3 the GRAMO with $\mathbb{P}$. Then, the optimal parameters $\theta_j \in \mathcal{P}$ are updated via solving Problem 1 the VQE, $\forall U \in U_{\circ}$. The alternate optimization procedure terminates until $U_{\circ}$ and $\mathbb{P}$ are converged. As a result, solutions of Algorithm 1 compose a subset of solutions of VA-VQE described in Problem 2 as shown in Theorem 1 (c.f., Section SC, Supporting Information, for the proof).
The reference step size $\epsilon_i$ and $\epsilon_f$. In the pool training, they are, respectively, the number of sampled ansatzes $N_i$, the maximum number of ranks of ansatzes whose corresponding parameters are updated $N_s$, the stable threshold for terminating the main process $N_f$, the maximum iteration times in the prethermalization $N_s$, and the maximum iteration times in the main process $N_f$. In the alternate training, they are, respectively, the population size $N_r$, the maximum number of ranks of ansatzes whose corresponding parameters are updated $N_i$, the stable threshold $N_f$, the optimization step in a generation $N_g$, and the maximum iteration times $N_s$.

In the VQE retraining, they include the maximum iteration times $N_i$.

This section is organized as follows: In Section 4.1, we concretely propose the structure of the search space of ansatz for the GCMA and conduct the size reduction to the search space based on the gate commutation rules. Subsequently, the pool training, alternate training, and VQE retraining stages are explained in Sections 4.2, 4.3, and 4.4, respectively.

### 4.1. Search Space of Ansatz

Without violation of Equation (2), we compose an ansatz in a layer-by-layer fashion as shown in Figure 1. To be specific, each layer consists of a set of native disjoint single-qubit gates followed by a set of hardware-compliant disjoint CNOT gates. It can be easily derived that the number of all possible structures $N_i$ of a layer is exponentially large with respect to the number of qubits $n$. We use the term state denoted by $s_i$ to indicate the $i$th structure is used. Then, an ansatz can be identified by a tuple of states

$$\left(s_1, s_2, \ldots, s_N\right)$$

(28)

where $s_i \in S := \{s_1, s_2, \ldots, s_{N_i}\}$. As a result, the search space of ansatz $S(N_i)$ can be specifically represented by a tree with $N_i + 1$ layers (from layer 0 to $N_i$). A node at the layer $l > 0$ in the tree is uniquely identified by its parent node and the state, that is, $v' := (v^{-1}, s)$, and linked to its child nodes. In layer 0, the root node of the tree is defined with no parent node and the state $s^0$ in which no layer information is stored. Furthermore, we call the nodes in layer $N_i$ linked to no child nodes the leaf nodes. Finally, a path from the root node to a leaf node $(v^0, v^1, \ldots, v^k)$ represents an ansatz $(s^0, s^1, \ldots, s^k)$.

The unrefined $S(N_i)$ is intuitively a full $N_i$-ary tree with $(N_i)^{N_i}$ leaf nodes. However, this simple-minded construction cannot establish the injection from paths to practical implementations of ansatzes (cf. Example 2). Moreover, gates in distinct layers may be deletable or mergeable (cf., Example 3). To further eliminate the redundant ansatzes and improve the efficiency, the paths violating the following cross-layer constraints based on gate commutation rules will be pruned.

**Constraints:** For node $v' := (v^l, s')$ in the path $(v^0, v^1, \ldots, v^k)$:

1. $R_i^q \not\subseteq s'$, if $R_i^q$ and $CNOT^{\gamma_q} \not\subseteq s'$, $\forall q$ with $|0 >$ initialization, $q'$ and $q''$, $l > k > 0$;
2. $CNOT^{\gamma_q} \not\subseteq s'$, if $\exists CNOT^{\gamma_q} \not\subseteq s'$ such that $R_i^q$, $R_i^q$, $CNOT^{\gamma_q}$, $CNOT^{\gamma_q}$, and $\not\subseteq s''$, $\forall q, q', q''$, $k < m < l < 0$;
3. $R_i^q \not\subseteq s''$, if $R_i^q$ and $CNOT^{\gamma_q} \not\subseteq s''$, $\forall q, q', l > 0$;

In this section, the gradient-cost multiobjective alternate framework (GCMA) for VA-VQE is proposed in detail. As the VA-VQE problem described, GCMA takes the gate set $G$ and the task $T$ as inputs. Based on the gate commutation rules, the search space of ansatz $S(N_i)$ under the maximum number of layers $N_i$ is reduced to boost the time efficiency of the optimal ansatz and parameters determination. Then, the quasi-optimal ansatz and corresponding trained parameters are output through three stages: pool training, alternate training, and VQE retraining. In the stage of pool training, a candidate tree $T$ is constructed for double $\epsilon$-greedy sampling. A parameter pool in which parameters are shared among ansatzes with similar structures is trained via exploiting the double $\epsilon$-greedy strategy based on the candidate tree. As a result, a reasonable set of parameters $P_o$ is generated as the initialization of parameters for the next stage. Based on the multiobjective genetic algorithm, the alternate training applies the framework described in Algorithm 1 to solve the VA-VQE problem via alternately solving the GRAMO in Problem 3 and the VQE in Problem 1. Since the evolutionary algorithm is applied, the output ansatz is quasi-optimal and corresponding parameters may be incompletely trained. Therefore, the third stage VQE retraining is required to guarantee the completion of parameter training of the quasi-optimal ansatz. We summarize the GCMA as Algorithm 2.

### Algorithm 2 GCMA

**Input:** $G$ 
begin 
$S(N_i) \leftarrow$ Construct search space by $G$; 
$P_o, T \leftarrow$ PoolTraining$(S(N_i))$; 
$U^*, \theta_U \leftarrow$ AlternateTraining$(S(N_i), P_o, T)$; 
$U^*, \theta^* \leftarrow$ VQETraining$(U^*, \theta_U)$; 
end 
return $U^*, \theta^*$.
Based on the weight-sharing policy, the number of parameters is significantly reduced. We provide an example, Example 4, to demonstrate the efficiency obtained from our constraints.

4) \( R_q \not\in s', \) if \( R_q, \text{CNOT} \not\in s', \) and \( \text{CNOT} \not\in s', \) for all \( q, q', l > 1. \)

5) \( s' \) is empty if \( s' \) is empty, \( l > 1. \)

The first constraint follows the fact that \( R_q, \) and \( \text{CNOT} \) preserve the state of the quantum system when the quantum state of \( q \) is \( |0\rangle. \) The second constraint avoids two consecutive CNOT gates with identical control and target qubits. As a result, the first two constraints eliminate deletable combinations of quantum gates. The mergeable combinations are extinguished by CNOTs with identical control and target qubits. As a result, we construct a parameter pool with linear size with respect to \( N_l \) via applying the weight-sharing policy and train the pool to eventually derive \( P_0. \) The parameter pool can be matrix-like defined as

\[
P := \begin{bmatrix}
\theta_{s_1,1} & \theta_{s_1,2} & \cdots & \theta_{s_1,N_l} \\
\theta_{s_2,1} & \theta_{s_2,2} & \cdots & \theta_{s_2,N_l} \\
\vdots & \vdots & \ddots & \vdots \\
\theta_{s_N,1} & \theta_{s_N,2} & \cdots & \theta_{s_N,N_l}
\end{bmatrix}
\] (29)

where \( \theta_{s_l} \) represents the parameters at the \( l \)th layer corresponding to the state \( s_l. \) Then, parameters of an ansatz \( (\nu^0, \nu^1, \ldots, \nu^K) \) is

\[
\theta := \theta_{s_l,1} \oplus \theta_{s_l,2} \oplus \cdots \oplus \theta_{s^N_l,N_l},
\] (30)

where \( \oplus \) indicates the direct sum such that

\[
\begin{bmatrix}
a_1 \\
a_2 \\
\vdots \\
a_N
\end{bmatrix} \oplus \begin{bmatrix}
b_1 \\
b_2 \\
\vdots \\
b_N
\end{bmatrix} = 
\begin{bmatrix}
a_1 \\
a_N \\
\vdots \\
b_N
\end{bmatrix}
\] (31)

It can be simply derived that parameters in identical layers and states are shared among ansatzes, which is the direct effect of applying the weight-sharing policy. Then, the number of trainable parameters is in \( O(N_lN_s) \), which is linear with respect to \( N_s. \) The exponential reduction significantly boosts the efficiency of parameter training. However, the training competitions are therefore introduced.

In this subsection, the pool training stage is presented in detail. Based on the weight-sharing policy, the number of parameters required to be trained is reduced to \( O(N_s) \). Moreover, the double \( e \)-greedy strategy is exploited accompanied by a candidate tree to mitigate the training competitions among ansatzes. We treat the pool training as a one-shot training program. At each iteration, several ansatzes are randomly sampled and estimated. According to the estimated performance, several temporarily outstanding ansatzes are selected to update parameters in specified steps (typically one step) via a gradient-based optimizer.

Recall that solving VQ-VQE in the framework as described in Algorithm 1 requires the initialization of \( P_0 \) in which each structure of ansatz links to an independent set of parameters. Intuitively, that parameters \( \theta_{s_l} \in P_0 \) substantially reflect the actual performance of ansatz \( U \), that is, \( \theta_{s_l} \approx \arg \min_{\theta} C(U, \theta) \), facilitates the optimal ansatz determination. Therefore, we conduct a pre-training of parameters before the alternate optimization solving VA-VQE instead of random initialization. Unfortunately, the size of \( P_0 \) is exponentially large with respect to \( n \) and \( N_l \), resulting from the exponentially large \( S(N_l) \). It is impractical to adequately train \( P_0 \) efficiently.

Instead, we construct a parameter pool with linear size with respect to \( N_l \) via applying the weight-sharing policy and train the pool to eventually derive \( P_0. \) The parameter pool can be matrix-like defined as

\[
P := \begin{bmatrix}
\theta_{s_1,1} & \theta_{s_1,2} & \cdots & \theta_{s_1,N_l} \\
\theta_{s_2,1} & \theta_{s_2,2} & \cdots & \theta_{s_2,N_l} \\
\vdots & \vdots & \ddots & \vdots \\
\theta_{s_N,1} & \theta_{s_N,2} & \cdots & \theta_{s_N,N_l}
\end{bmatrix}
\] (29)

where \( \theta_{s_l} \) represents the parameters at the \( l \)th layer corresponding to the state \( s_l. \) Then, parameters of an ansatz \( (\nu^0, \nu^1, \ldots, \nu^K) \) is

\[
\theta := \theta_{s_l,1} \oplus \theta_{s_l,2} \oplus \cdots \oplus \theta_{s^N_l,N_l},
\] (30)

where \( \oplus \) indicates the direct sum such that

\[
\begin{bmatrix}
a_1 \\
a_2 \\
\vdots \\
a_N
\end{bmatrix} \oplus \begin{bmatrix}
b_1 \\
b_2 \\
\vdots \\
b_N
\end{bmatrix} = 
\begin{bmatrix}
a_1 \\
a_N \\
\vdots \\
b_N
\end{bmatrix}
\] (31)

It can be simply derived that parameters in identical layers and states are shared among ansatzes, which is the direct effect of applying the weight-sharing policy. Then, the number of trainable parameters is in \( O(N_lN_s) \), which is linear with respect to \( N_s. \) The exponential reduction significantly boosts the efficiency of parameter training. However, the training competitions are therefore introduced.

Figure 1. The layer-by-layer ansatz a) with each layer being decomposed into two sublayers, that is, b) a single-qubit gate layer and a CNOT layer, and c) four exemplary constructions of layers.
Inspired by the $\epsilon$-greedy strategy from traditional machine learning,[49] we propose the double $\epsilon$-greedy strategy based on the candidate tree to mitigate the training competitions. Remarkably, our proposed method not only differentiates “good” and “bad” ansatzes but also ansatzes among the two categories.

Recall that the search space $S(N_l)$ can be represented by a tree. It is intuitive that constructing a tree to save potentially “good” ansatzes and discarding potentially “bad” ansatzes are reasonable for the differentiation. The tree spanned by paths representing potentially “good” ansatzes is named the candidate tree. Each node $v$ in the tree maintains a leaf count $c_l(v)$ to indicate the number of leaf nodes below $v$ and a training count $c_t(v)$ to record the total times of training of the node. At each path sampling procedure, the GCMA samples a path from the candidate tree with the probability $\epsilon_1$ and the $S(N_l)$ uniformly with the probability $1 - \epsilon_1$. While sampling from the candidate tree, nodes are successively sampled from the root to a leaf. Since only potentially “good” ansatzes are appended to the candidate tree and recorded the training count of nodes via the one-shot training scheme, nodes in the candidate tree with a large training count may intuitively have more probability to construct potentially “good” ansatzes. Let the last selected node be $v_{l-1}$, $1 \leq l \leq N_l$, linked to child nodes $v_{l1}, v_{l2}, \ldots, v_{lN_l}$. Then, the next node is sampled as $v_{lk}$ with the probability

$$\Pr(v_{lk}; \eta) = \frac{c_l(v_{lk}) + \eta c_t(v_{lk})}{\sum_{i=1}^{N_l} c_l(v_i) + \eta c_t(v_i)}$$

where $\eta = 1$ with the probability $\epsilon_1$ representing the greedy sampling and $\eta = 0$ with the probability $1 - \epsilon_2$ representing the uniform sampling. The ansatz sampling procedure is summarized as Algorithm 3.

As depicted in Figure 2, at each main process iteration of the pool training, the GCMA samples $N_s$ paths exploiting the double $\epsilon$-greedy strategy. Then, the cost function values and gradients of sampled ansatzes are estimated with corresponding parameters from the parameter pool. Subsequently, the $N_{r_1}$-rank solution

---

**Algorithm 3 USampling**

**Input:** $S(N_l)$, $T$, $\epsilon_1$, $\epsilon_2$

**begin**

$J_1 \sim \{(True, \epsilon_1), (False, 1 - \epsilon_1)\}$

if $J_1$ is True then

$v \leftarrow v^0 \in T$

$U \leftarrow \{v\}$

$\eta \sim \{(1, \epsilon_2), (0, 1 - \epsilon_2)\}$

for $v$ is not a leaf node do

$V_c \leftarrow \text{GetChildNodes}(v)$

$v' \sim \{(v'_k \in V_c, \Pr(v'_k; \eta))\}$

$U \leftarrow U \oplus \{v'\}$

$v \leftarrow v'$

end

else

$U \sim S(N_l)$

end

**return** $U$.

---

**Figure 2.** Single iteration of pool training with $N_l = 3$. 
Algorithm 4 Pool Training Main Process

Input: $S(N_1), P, T$

begin
    $S \leftarrow S(N_1), t \leftarrow 0, c_t' \leftarrow 1$
    for $i = 1 : N_{i_3}$ do
        $U_s \leftarrow \{\}$
        for $|U_s| < N_{s_1}$ do
            $U \leftarrow USampling(S, T, \epsilon_1, \epsilon_2)$
            $U_s \leftarrow U_s \cup \{U\}$
        end
        $\mathbb{U}_n(N_{r_2}) \leftarrow \text{SolveProb3}(U_s, P)$;
        foreach $U \in \mathbb{U}_n(N_{r_1})$ do
            $\theta \leftarrow \text{GetParameters}(U, P)$;
            $\alpha \leftarrow \text{GetStepSize}(\alpha_0)$;
            $\theta \leftarrow \theta - \alpha \nabla C(U, \theta)$;
            $P \leftarrow \text{UpdatePool}(\theta, P)$;
            $T \leftarrow \text{AppendPool}(\theta, T)$;
        end
        if $c_t(v^0) = c_t'$ then
            $t \leftarrow t + 1$;
            if $t = N_{i_3}$ then
                break;
            end
        else
            $t \leftarrow 0$;
            $c_t' \leftarrow c_t(v^0)$;
        end
    end
return $P, T$.

$U_s(N_{r_1})$ of Problem 3 is determined among the sampled ansatzes. For each ansatz $U \in U_s(N_{r_1})$, corresponding parameters in the parameter pool are updated in one step via gradient descent optimizer. The main process terminates at iteration $N_0$ or when $c_t(v^0)$ is stable that the value has been preserved for $N_{i_3}$ iterations which means there is no new path appended on the tree. As a result, the main process of the pool training is summarized as Algorithm 4.

For the stability of pool training, we provide a prethermalization (c.f. Algorithm S8, Supporting Information, for the pseudocode) before the main process. Given an $N_{i_0}$, at iteration $i$, the GCMA processes as the main process iteration with $c_t' = (i - 1)\epsilon_i/N_{i_0}$ instead of $\epsilon_i$.

Finally, the pool training can be described by Algorithm 5. After the initialization of $P$ and $T$, the prethermalization and the main process are conducted to train $P$ as well as $T$. Subsequently, the set of parameters $P_0$ can be constructed by expanding $P$ that $\theta_U$ is generated by Equation (29) for any ansatz $U \in S(N_0)$. The $P_0$ and $T$ are output for the next stage.

4.3. Alternate Training

Recall that the VA-VQE can be solved via alternately solving Problem 3 and Problem 1. We exploit the multiobjective genetic algorithm with novel modification. The set of parameters $P$ is initialized as $P_0$. The individuals in the first generation are sampled independently via the double $\epsilon$-greedy strategy as described in Section 4.2 to compose the initial population.

As shown in Figure 3, at each generation (iteration), ansatzes in $U_s(N_{i_0})$ from solving Problem 3 with respect to the population are trained to update $P$ in $N_0$ steps via gradient descent optimizer. Note that the number of ansatzes in $U_s(N_{i_0})$ should be less than $N_{i_0}/2$. Similar to the traditional genetic algorithm NGSAG-II, $N_{i_0}/2$ ansatzes are survived. Specifically, the GCMA finds an $N_r$ such that

$$|U_n(N_r)| < N_{i_0}/2 \leq |U_n(N_r + 1)|$$

Then, ansatzes are sequentially inserted into survivals in increasing order of the cost function value from $U_n(N_r + 1) \setminus U_n(N_r)$. Subsequently, new ansatzes are generated by applying asexual genetic operators on survivals to fill the population. We leave the description of asexual genetic operators in Section SD, Supporting Information. Besides, we introduce the explicit elimination to the near-completely trained survivals $U$ satisfying $\alpha \frac{|\nabla C(U, \theta)|}{\|\nabla C(U, \theta)\|} < \xi$, where $\alpha$ is the step size. The eliminated ansatz is recorded if it has the temporarily lowest cost function value and is erased if there exists an ansatz with a lower cost function value. Notice that the elimination is conducted after the application of genetic operators. Therefore, the sampling based on the double $\epsilon$-greedy is required to refill the population.

The alternate training terminates at the generation $N_9$ or when the record eliminated ansatz is preserved for $N_{i_0}/2$ generations. As a result, the ansatz $U^*$ with the temporarily lowest cost function value is output as the quasi-optimal ansatz. Meanwhile, the corresponding parameters $\theta_{U^*}$ are output for the next stage as the parameter initialization. We summarize the alternate training in Algorithm 6.

4.4. VQE Retraining

The final stage of VQE retraining inherits the quasi-optimal ansatz $U^*$ and corresponding parameters $\theta_{U^*}$ output from the alternate training and provides a guarantee of the sufficiency of the parameter training of $\theta_{U^*}$. As traditional VQE training does, this stage simply trains the parameters of $\theta_{U^*}$ with the initialization $\theta_{U^*}$ until the cost function value converges or
the iteration count reaches $N_i$. We summarize this stage as Algorithm 7 for completeness.

5. Numerical Simulations

In this section, we conduct numerical simulations on VQE tasks for finding ground state energies of $H_6$ molecule to show the improvement of the proposed framework. Moreover, to demonstrate the flexibility, we implement a modified version of the proposed framework for meta-VQE learning energy profiles of $H_2$ molecule with respect to the bond distance.

5.1. Finding Ground State Energies

To showcase the improvement of the proposed framework, we compare it with a fully randomized (RND) baseline algorithm, the quantum circuit architecture search (QAS), and a hardware-efficient adaptive derivative assembled (ADA) method derived from refs. [27, 28]. To make a fair comparison, we use Python with the **Pennylane** and **Qulacs** package to implement all methods mentioned above. The technical details for algorithms to be compared are elaborated in Section SF, Supporting Information, and summarized as follows:

i) RND: We sample $N_s$ circuits with $N_l$ layers and unique random parameters, then process the VQE retraining and output the circuit with the minimum cost function value;

ii) QAS: QAS utilizes resource sharing strategy and adversarial bandit training technique to implement the VA-VQE. Its performance has been verified by experiments on real quantum devices and can be treated as the cost-criterion-based state-of-the-art.

iii) Revised quantum circuit architecture search (QAS'): For the fairness, we also implement a modified QAS (denoted by QAS'). The only difference to QAS is that QAS' samples layers with our adopted layer structure as shown in Figure 1.

iv) ADA: ADA is a moderate hardware-efficient VA-VQE implementation modified based on gradient-criterion-based ADAPT-VQE and qubit-ADAPT-VQE. The ADA starts with an empty ansatz and append layers iteratively. In each iteration, the layer with maximum norm of gradient magnitude among $N_s$ sampled layers append to the ansatz. The expanded ansatz is trained by VQE completely to determine the current optimal parameters.

We assume that amplitude damping and depolarization dominates the quantum error in the quantum hardware. In our configuration, an amplitude damping channel and a depolarizing channel with error probability 0.001 are applied after a single-qubit gate. We simultaneously apply two pairs of amplitude damping and depolarizing channels with error probability 0.01 after a CNOT, that is, apply one on the control qubit, and one on the target qubit.

The criteria used for all methods are absolute error between the obtained and exact optimal cost function values to show the accuracy of methods, since it is the most direct reflection of effectiveness of a VA-VQE implementation. The invoking times for the calculation of the cost function (termed quantum cost henceforth) is adopted to indicate the resource consumption, since the calculation of the cost function requires multiple quantum measurements, which is the most time-consuming procedure in simulation. To be specific, each estimation of cost function value requires 1 quantum cost, while the estimation of partial derivative of a parameter requires 2.

The quantum cost reflects the real resource consumption in quantum-classical hybrid scheme than the absolute execution time while simulating quantum mechanics by classical computer. First, all methods implemented in this paper iteratively solve the VA-VQE. Each iteration requires sampling ansatzes from the search space and estimating performances of ansatzes. The resource consumption of estimating performances of ansatzes reflects both the times of ansatz sampling and the times the quantum measurements, which can be indicated by quantum cost. Second, the multi-thread technique is generally
Algorithm 6 Alternate Training

Input: $\mathcal{S}(N_i), P_0, T$
begin
| $P \leftarrow P_0$, $U_s \leftarrow \emptyset$, $U_{best} \leftarrow \text{None}$; |
| for $|U_s| \leq N_{12}$ do |
| $U_s \leftarrow U_s \cup \{\text{USampling}([S, T, \epsilon_1, \epsilon_2])\}$; |
| end |
| for $i = 1 : N_{12}$ do |
| $U_{n}(N_{12}), U_{n}(N_{12} + 1) \leftarrow \text{SolveProb3}(U_s, P)$; |
| foreach $U \in U_{n}(N_{12})$ do |
| $\theta_U \leftarrow \text{Update}\theta_U$ by gradient based optimizer in $N_{12}$; |
| end |
| $U_{survival} \leftarrow \text{GetSurvivals}(U_{n}(N_{12}), U_{n}(N_{12} + 1))$; |
| if $\exists U \in U_{survival}$ such that $C(U, \theta_U) < C(U_{best}, \theta_{U_{best}})$ then |
| $U_{best} \leftarrow \text{None}$; |
| end |
| $U_{new} \leftarrow \emptyset$; |
| foreach $U \in U_{survival}$ do |
| $U_{new} \leftarrow \text{RandomGeneticOperator}(U)$; |
| $U_{new} \leftarrow U_{new} \cup \{U_{new}\}$; |
| $\alpha \leftarrow \text{GetStepSize}(\alpha_0)$; |
| if $\alpha \frac{|\nabla C(U, \theta_U)|}{|\theta_U|} \leq \xi$ then |
| $U_{survival} \leftarrow U_{survival} \setminus \{U\}$; |
| if $C(U, \theta_U) < C(U_{best}, \theta_{U_{best}})$ then |
| $U_{best} \leftarrow U$; |
| end |
| end |
| $U_s \leftarrow U_{survival} \cap U_{new}$; |
| for $|U_s| \leq N_{12}$ do |
| $U_s \leftarrow U_s \cup \{\text{USampling}([S, T, \epsilon_1, \epsilon_2])\}$; |
| end |
| if $U_{best}$ has been preserved for $N_{12}$ generations then |
| break; |
| end |
| $U^* \leftarrow \text{arg min}_{U \in U_s \cup U_{best}} C(U, \theta_U)$; |
| end |
return $U^*, \theta_U$.

used in classical simulation and is not available on quantum device. For example, determining the ansatz with the minimum cost function value among a set of ansatzes requires estimating the cost function value of each ansatz in the set. It can be calculated parallelly by a multi-core classical computer but sequentially by a quantum device.

We analyze the distribution of performances of output ansatzes and calculate the mean square error (MSE) mathematically represented by

$$\text{MSE} = \frac{1}{M} \sum_{i=1}^{M} (\tilde{C}_i - C_{\text{exact}})^2$$ (34)

Algorithm 7 VQE Retraining

Input: $U$, $\theta_0$
begin
| $\theta \leftarrow \theta_0$; |
| for $i = 1 : N_{13}$ do |
| $\alpha \leftarrow \text{GetStepSize}(\alpha_0)$; |
| $\theta \leftarrow \theta - \alpha \nabla C(U, \theta)$; |
| if $\alpha \frac{|\nabla C(U, \theta)|}{|\theta|} \leq \xi$ then |
| break; |
| end |
| $U^* \leftarrow U$, $\theta^* \leftarrow \theta$; |
| end |
return $U^*$, $\theta^*$.

to indicate the stability of obtaining the quasi-optimal cost function values, where $M$ is the total running time, $\tilde{C}$ is the output quasi-optimal cost function value and $C_{\text{exact}}$ is the exact optimal cost function value.

To quantify the expressibility and trainability, we first demonstrate the number of parameters required to be trained of each method. Then, we exploit the Jensen–Shannon distance (JSD) of the fidelity distribution of states generated from the output quasi-optimal ansatz $P_{\text{ansatz}}(F; \theta)$ and Haar random states $P_{\text{Haar}}(F)$ to showcase the expressibility

$$\text{Expr} = \text{JSD}(P_{\text{ansatz}}(F; \theta), P_{\text{Haar}}(F))$$ (35)

As a result, an ansatz with high expressibility exhibits low Expr value. For the trainability, a quantitative analyze of the BP phenomenon is conducted by testing the maximum expectation of the module of partial derivatives with respect to parameters subject to uniform distribution

$$\langle d \rangle = \max_{\theta \in U_\theta} \int_{U_\theta} |\partial C(U, \theta)| d\theta$$ (36)

where $U_\theta$ indicates the uniform distribution of $\theta$.

We first conduct numerical simulations on determining the ground state energies of Hamiltonians of $H_6$ chain in various bond distances. The quantum device is assumed to have 12 qubits with ring connectives and amplitude damping and depolarization errors being imposed by the classical simulator. The implementation details of all methods are described in Section SF, Supporting Information.

For each bond distance, we conduct the GCMA and comparison methods 100 times, respectively. The results on average are shown in Figure 4. It can be found that our algorithm can determine ground state energy with consistently lower absolute error on average than the compared algorithms in various bond distances with almost lower quantum cost. Specifically, GCMA shows average improvement in terms of absolute error by 58.8%, 69.8%, 59.6%, and 54.1% and in terms of quantum cost by 16.8%, 28.2%, 27.6%, and 10.2% compared with RND, QAS, QAS', and ADA, respectively. This conclusion holds while considering the ansatzes with the lowest cost function values among the
Figure 4. The average result of determining ground state energy of H₆ in various bond distances among 100 running times. Black dashed lines from bottom to top demonstrate energy = $\zeta$, 100$\zeta$, 200$\zeta$, and 300$\zeta$, respectively, where $\zeta = 0.0015$ Ha corresponding to the chemical accuracy.

Figure 5. The best result of determining ground state energy of H₆ in various bond distances among 100 running times. Black dashed lines from bottom to top demonstrate energy = $\zeta$, 100$\zeta$, and 200$\zeta$, respectively, where $\zeta = 0.0015$ Ha corresponding to the chemical accuracy.

Figure 6. Mean square error among 100 running times.

100 running times as depicted in Figure 5. Moreover, GCMA exhibits outstanding stability as shown in Figure 6 with average improvement by 81.1%, 87.5%, 78.8%, and 73.5% compared with RND, QAS, QAS', and ADA, respectively.

For bond distance 0.9Å, we also record the quantum cost and absolute error for each running time in Figures 7 and 8 when the algorithm converges to the final output result. It can be concluded that our algorithm can obtain the best solution on average with a rather small quantum cost (about 26149.67) which is conspicuously better than RND (67.8%, 14.3%, 86.8%) and QAS (78.2%, 25.7%, 92.4%), QAS' (67.4%, 25.2%, 83.2%), and ADA (55.0%, 6.1%, 70.0%). Remarkably, GCMA asymptotically reach the best ansatz among the training, which means that a temporarily quasi-optimal ansatz can be output at any generation of alternate training. However, our framework shows large variance of the quantum cost. This may result from the termination conditions of the three stages. It can be concluded that the termination conditions prevent the redundant iterations which can not significantly benefit the training, and thus the performance of our framework may not be sensitive to hyperparameters $N_{i0}$, $N_{i1}$, $N_{i2}$, and $N_{i3}$ when they are sufficiently large.

To showcase the expressibility and trainability, we first demonstrate the number of parameters in Figure 9. It is evident that the number of parameters in our proposed method is reduced significantly. This intuitively results in the relatively lower expressibility and higher trainability. Then, we estimate the Expr and $<d>$ of 50 lowest-cost ansatzes generated from each method, where the fidelity distribution of the ansatz states $P_{\text{ansatz}}(F; \theta)$ in Expr is determined by uniformly sampling two groups of 200 sets of parameters and calculate the fidelity (overlap) between states generated by the two groups of parameters and $\int_{\theta} |\partial_C(U, \theta)|d\theta$ is determined by averaging the modules of partial derivatives among 200 samples of parameters $\theta$. The results are shown in Table 1 and Figure 10. As a result, the GCMA exhibits mitigated BP phenomenon ($<d> = 0.416$) at a cost of relatively low expressibility (Expr = 0.252). However, it is sufficient to reach the highest accuracy with the limited expressibility by eliminating the redundant quantum gates.

We also conduct a modified version of the proposed framework, marked by GCMA-PT, based on the CCSD and pseudo-Trotterization to showcase the flexibility. The state of a node in layer $l$ represents a Pauli operator $P_l$ in CCSD. Hence, a path with $N_l$ nodes indicates an ansatz by pseudo-Trotterization

$$U(\theta) = e^{iP_{N_l}} e^{iP_{N_l-1}} \ldots e^{iP_1}$$

(37)

The only cross-layer constraint is that $P_l \neq P_{l-1}$ without delicate design. The implementation detail and simulation results are described in Section SG, Supporting Information.
Figure 7. Detail of cost function values with respect to quantum cost in solving the ground state energy of $H_6$ in bond distance 0.9 Å among 100 running times. a) The average cost function values with respect to quantum cost. Note that we demonstrate the final result in average instead of the detail for RND because of its randomness. b) The cost function values among 100 running times with respect to quantum cost in the running time which output the ansatz with the lowest cost function value among 100 running times. The ▲, ▼, and × represent the termination of pool training’s prethermalization, pool training’s main process, and alternate training, respectively. For RND, we only show the detail of its retraining stage because of the randomness of sampling.

![Figure 7](image1.png)

Figure 8. Output quasi-optimal cost function values with respect to the quantum cost of 100 running times in solving the ground state energy. Dashed lines present cost function values on average.

As a result, the GCMA-PT shows better accuracy (cost function value of $-3.120$ at $N_l = 5$) in noisy situation with more quantum cost required compared with qubit-ADAPT-VQE.[28] However, the performance can not exceed that of the hardware-efficient based GCMA, since the implementation of pseudo-Trotterization requires deep circuits and massive CNOT gates and hence suffers serious quantum noise which also result in the degradation of accuracy at $N_l = 7$. In the noiseless situation, GCMA-PT cannot obtain the best accuracy. This may result from the absence of elaborate designed cross-layer constraints and the instability of probabilistic sampling. Remarkably, our method shows conspicuously better accuracy when $N_l$ is small. Especially, the cost

![Figure 8](image2.png)

Figure 9. The number of parameters required to be trained in various bond distances. a) The number of parameters on average among 100 running times in each bond distance. b) The number of parameters of the ansatz with the lowest ground state energy among 100 running times in each bond distance.

![Figure 9](image3.png)

| Method | Cost | Expr | $<d>$ |
|--------|------|------|-------|
| GCMA   | $-3.153$ | $0.252$ | $0.416$ |
| QAS    | $-2.774$ | $0.317$ | $0.381$ |
| QAS$'$ | $-2.918$ | $0.280$ | $0.389$ |
| RND    | $-2.981$ | $0.313$ | $0.401$ |
| ADA    | $-3.048$ | $0.283$ | $0.411$ |

Table 1. The average result of cost function values, Expr, and $<d>$ among 50 lowest-cost ansatzes of each method.
function values of GCMA-PT are $-3.100$ and $-3.173$ when $N_l = 3$ while it of qubit-ADAPT are $-0.752$ and $-0.761$ by noisy and ideal simulators, respectively. This feature of GCMA-PT shows great suitability for VQE on NISQ devices.

5.2. Learning Energy Profiles

To showcase the flexibility, we implement a modified version of the proposed framework for Meta-VQE.\cite{50} Let the parameterized Hamiltonian be $H(\Delta)$. Then, sample $M$ available values of $\Delta$ denoted by $\Delta_1, \Delta_2, \ldots, \Delta_M$ as training bonds. Hence, the cost function for training Meta-VQE is defined by

$$C_{\text{Meta}}(\theta) = \sum_{i=1}^{M} \text{Tr}[H(\Delta_i) f(\Delta_i; \theta)^2 U^\dagger f(\Delta_i; \theta)]$$

where $f(\Delta_i; \cdot)$ represents the encoding function of the parameter of the Hamiltonian with respect to operand trainable parameters. After parameter training, the landscape of

$$V_{\text{Meta}}(\Delta) = \text{Tr}[H(\Delta) U f(\Delta_i; \theta)^2 U^\dagger f(\Delta_i; \theta)]$$

exhibits the energy profile of the parameterized Hamiltonian.

We implement the Meta-VQE adapted GCMA by simply identifying each state $s$ by both the structure and the encoding function of each single-qubit gate in the structure to enable the auto-decision of the encoding methods. Here we provide three encoding functions for parameters of single-qubit gates. They are, respectively,

$$f_1(\Delta; \theta) = \theta$$

$$f_2(\Delta; \theta, \gamma) = \theta \Delta + \gamma$$

$$f_3(\Delta; \theta, \gamma) = \theta e^{\Delta} + \gamma$$

We compare our framework with the HEA implementation. Each HEA consists of $N_e$ encoding layers and $N_p$ processing layers and is denoted by HEA$-N_e$-$N_p$. Specifically, $f_2$ and $f_1$ are applied on each single-qubit gate in encoding layers and processing layers, respectively.

Simulations are conducted for $H_2$, where $\Delta$ in $H(\Delta)$ represents the bond distance. Training bonds are set as $0.5, 0.9, 1.3, 1.7, 2.1, 2.5$, and $2.9$. We set $N_l = 4$ and other configurations as that for the VQE task of $H_2$. Results are shown in Figure 11 and Table 2. It can be concluded that our framework learns the energy profile with the conspicuously lower variance of absolute error, which indicates more information on chemical properties.

6. Conclusion

In this paper, addressing the issues in mitigating the BP phenomenon in VQE, we propose a gradient-cost multiobjective alternate framework for VQE with the variable ansatz strategy. We propose a theoretical framework by comprehensively considering cost function values and gradient magnitudes and exploiting the alternate optimization scheme so that the local optimum can be avoided from the ansatz perspective. It can be theoretically

![Figure 10](image-url)  
Figure 10. The Expr with respect to $<d>$ of 50 lowest-cost ansatzes generated from each methods. Dashed lines present average values.

![Table 2](image-url)  
Table 2. The variance of absolute error and corresponding improvement.

|        | GCMA | HEA-1-1 | HEA-2-2 | HEA-3-1 | HEA-4-0 |
|--------|------|---------|---------|---------|---------|
| Variance | 0.001 | 0.078   | 0.024   | 0.020   | 0.033   |
| Improvement | -    | 98.8%   | 96.1%   | 95.4%   | 97.2%   |

![Figure 11](image-url)  
Figure 11. The result of determining the energy profile of $H_2$. Black dashed lines from bottom to top demonstrate energy $= \zeta, 100\zeta, 200\zeta$, and $300\zeta$, respectively, where $\zeta = 0.0015$ Ha corresponding to the chemical accuracy.
proved that the result of the proposed theoretical framework is a subset of the results of the original VA-VQE.

Then, based on the theoretical framework, a novel implementation named GCMA is proposed with three stages. We reduce the size of the search space of the ansatz via applying gate commutation rules and establishing an injection from the search space to the practical implementations of ansatizes to boost the time efficiency of the optimal ansatz and parameter determination. Exploiting the double ε-greedy strategy based on the candidate tree, an initialization of parameters is determined, so that the training competitions are mitigated and the local optimum can be evaded from the parameter perspective. Based on the initialization of parameters, the GCMA follows the framework with an elaborately modified genetic algorithm to find and train a quasi-optimal ansatz.

Finally, we conduct numerical simulations on quantum chemistry to find the ground state energy of a quantum system. We adopt relatively fair criteria for measuring the performance of VA-VQE so that the transverse comparison among methods of VA-VQE can be clearly conducted. As a result, the GCMA exhibits conspicuously better performance in terms of error, quantum cost, and stability on average compared with the state-of-the-art VQE so that the transverse comparison among methods of VA-VQE can be clearly conducted. According to the initialization of parameters, the GCMA follows the framework with an elaborately modified genetic algorithm to find and train a quasi-optimal ansatz.

Although the proposed method obtained better results, further research is required to be conducted. For example, GCMA requires a large number of hyperparameters to be adjusted such as $N$, Research on automatic hyperparameter adjustment may be meaningful for the VA-VQE.

Supporting Information

Supporting Information is available from the Wiley Online Library or from the author.

Acknowledgements

Z.-T.L. and F.-X.M. contributed equally to this work. This work was supported by the National Natural Science Foundation of China (No. 61871111 and No. 61960206005).

Conflict of Interest

The authors declare no conflict of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Keywords

alternate optimization, quantum architecture search, variable ansatz, variational quantum eigensolver

Received: September 30, 2022
Revised: March 2, 2023
Published online: April 7, 2023

[1] L. K. Grover, Phys. Rev. Lett. 1997, 79, 4709.
[2] P. W. Shor, in SIAM Review, Vol. 41, SIAM, Philadelphia 1999, pp. 303–332.
[3] A. W. Harrow, A. Hassidim, S. Lloyd, Phys. Rev. Lett. 2009, 103, 150502.
[4] G. H. Low, I. L. Chuang, Quantum 2019, 3, 163.
[5] J. Preskill, Quantum 2018, 2, 79.
[6] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, J. L. O’brien, Nat. Commun. 2014, 5, 4213.
[7] R. J. Bartlett, M. Musial, Rev. Mod. Phys. 2007, 79, 291.
[8] M.-H. Yung, J. Casanova, A. Mezzacapo, J. McClean, L. Lamata, A. Aspuru-Guzik, E. Solano, Sci. Rep. 2014, 4, 3589.
[9] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, J. M. Gambetta, Nature 2017, 549, 242.
[10] C. Bravo-Prieto, R. LaRose, M. Cerezo, Y. Subasi, L. Cincio, P. J. Coles, arXiv:1909.05820 [quant-ph], 2020.
[11] X. Yuan, S. Endo, Q. Zhao, Y. Li, S. C. Benjamin, Quantum 2019, 3, 191.
[12] X. Wang, Z. Song, Y. Wang, Quantum 2021, 5, 483.
[13] F.-X. Meng, Z.-T. Li, X. Yu, Z. Zhang, Quantum Sci. Technol. 2021, 7, 025002.
[14] Z.-T. Li, F.-X. Meng, X.-T. Yu, Z.-C. Zhang, Quantum Inf. Process. 2022, 21, 11.
[15] K. Beer, D. Bondarenko, T. Farrelly, T. J. Osborne, R. Salzmann, D. Scheiermann, R. Wolf, Nat. Commun. 2020, 11, 808.
[16] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, S. Woerner, Nat. Comput. Sci. 2021, 1, 403.
[17] J. Biarmont, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, S. Lloyd, Nature 2017, 549, 195.
[18] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, J. M. Gambetta, Nature 2019, 567, 209.
[19] Y. Du, T. Huang, S. You, M.-H. Hsieh, D. Tao, npj Quantum Inf. 2022, 8, 62.
[20] J. R. McClean, S. Boixo, M.-H. Yung, J. Mcclean, L. Lamata, A. Aspuru-Guzik, E. Solano, Sci. Rep. 2020, 10, 163.
[21] T. L. Patti, K. Najafi, X. Gao, S. F. Yelin, Phys. Rev. Res. 2021, 3, 033090.
[22] C. Ortiz Marrero, M. Kieferová, N. Wiebe, PRX Quantum 2021, 2, 040316.
[23] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, P. J. Coles, Nat. Commun. 2021, 12, 1791.
[24] S. H. Sack, R. A. Medina, A. A. Michailidis, R. Kueng, M. Serbyn, arXiv:2201.08194 [quant-ph], 2022.
[25] Z. Holmes, K. Sharma, M. Cerezo, P. J. Coles, PRX Quantum 2022, 3, 010313.
[26] S. Wang, E. Fontana, M. Cerezo, K. Sharma, A. Sone, L. Cincio, P. J. Coles, Nat. Commun. 2021, 12, 6961.
[27] H. R. Grimsmo, S. E. Economou, E. Barnes, N. J. Mayhall, Nat. Commun. 2019, 10, 3007.
[28] H. L. Tang, V. Shkolnikov, G. S. Barron, H. R. Grimsmo, N. J. Mayhall, E. Barnes, S. E. Economou, PRX Quantum 2021, 2, 020310.
[29] A. G. Rattew, S. Hu, M. Pistoia, R. Chen, S. Wood, arXiv:1910.09694 [quant-ph], 2020.
[30] S.-X. Zhang, C.-Y. Hsieh, S. Zhang, H. Yao, arXiv:2010.08561 [quant-ph], 2021.
[31] M. Ostaszewski, L. M. Trenkwalder, W. Masarczyk, E. Scerri, V. Dunjko, in Advances in Neural Information Processing Systems, Vol. 34, Curran Associates, Inc., Red Hook, NY 2021, pp. 18182–18194.

[32] M. Bilikis, M. Cerezo, G. Verdon, P. J. Coles, L. Cincio, arXiv:2103.06712 [quant-ph, stat], 2022.

[33] Z.-J. Zhang, T. H. Kyaw, J. Kottmann, M. Degroote, A. Aspuru-Guzik, Quantum Sci. Technol. 2021, 6, 035001.

[34] F. Meng, Z.-T. Li, X.-T. Yu, Z.-C. Zhang, IEEE Trans. Quantum Eng. 2021, 2, 3103910.

[35] D. Chivilikhin, A. Samarini, V. Ulyantsev, I. Iorsh, A. R. Oganov, O. Kyriienko, arXiv:2007.04424 [cond-mat, physics:quant-ph], 2020.

[36] K. Mitarai, M. Negoro, M. Kitagawa, K. Fujii, Phys. Rev. A 2018, 98, 032309.

[37] T. Elsken, J. H. Metzen, F. Hutter, arXiv:1808.05377 [cs, stat], 2019.

[38] S. Sim, P. D. Johnson, A. Aspuru-Guzik, Adv. Quantum Technol. 2019, 2, 1900070.

[39] K. Nakaji, N. Yamamoto, Quantum 2021, 5, 434.

[40] M. Cerezo, P. J. Coles, Quantum Sci. Technol. 2021, 6, 035006.

[41] A. Arrasmith, M. Cerezo, P. Czarnik, L. Cincio, P. J. Coles, Quantum 2021, 5, 558.

[42] D. Stilck França, R. Garcia-Patron, Nat. Phys. 2021, 17, 1221.

[43] A. Pesah, M. Cerezo, S. Wang, T. Volkoff, A. T. Sornborger, P. J. Coles, Phys. Rev. X 2021, 11, 041011.

[44] T. Volkoff, P. J. Coles, Quantum Sci. Technol. 2021, 6, 025008.

[45] A. Skolik, J. R. McClean, M. Mohseni, P. van der Smagt, M. Leib, Quantum Mach. Intell. 2021, 3, 5.

[46] E. Grant, L. Wossnig, M. Ostaszewski, M. Benedetti, Quantum 2019, 3, 214.

[47] G. Verdon, M. Broughton, J. R. McClean, K. J. Sung, R. Babbush, Z. Jiang, H. Neven, M. Mohseni, arXiv:1907.05415 [quant-ph], 2019.

[48] S. You, T. Huang, M. Yang, F. Wang, C. Qian, C. Zhang, in Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition, IEEE, Piscataway, NJ 2020, pp. 1999–2008.

[49] A. Cervera-Lierta, J. S. Kottmann, A. Aspuru-Guzik, PRX Quantum 2021, 2, 020329.

[50] V. Bergholm, J. Izaac, M. Schuld, M. S. Alam, S. Ahmed, J. M. Arrazola, C. Blank, A. Delgado, S. Jahangiri, K. McKiernan, J. J. Meyer, Z. Niu, A. Száva, N. Killoran, arXiv:1811.04968 [physics, physics:quant-ph], 2020.

[51] Y. Suzuki, Y. Kawase, Y. Masumura, Y. Hiraga, M. Nakadai, J. Chen, K. M. Nakanishi, K. Mitarai, R. Imai, S. Tamiya, Quantum 2021, 5, 559.

[52] K. Linghu, Y. Qian, R. Wang, M.-J. Hu, Z. Li, X. Li, H. Xu, J. Zhang, T. Ma, P. Zhao, D. E. Liu, M.-H. Hsieh, X. Wu, Y. Du, D. Tao, Y. Jin, H. Yu, arXiv:2201.00934 [quant-ph], 2022.

[53] U. Azad, A. Sinha, arXiv:2205.02095, 2022.

[54] Percentages in order in brackets, henceforth, are improvements in terms of error, quantum cost, and stability, respectively.