Schrödinger-Heisenberg Variational Quantum Algorithm

YunGu 21110301

Main Reference Paper: Shang, Z.-X., Chen, M.-C., Yuan, X., Lu, C.-Y., & Pan, J.-W. (2023). Schrödinger-Heisenberg Variational Quantum Algorithms. *Physical Review Letters*, *131*(6), 060406. https://doi.org/10.1103/PhysRevLett.131.060406

1 Variational Quantum Algorithms

1.1 VQE: Current Mainstream Paradigm and its Bottlenecks

1.1.1 Working Principle

The Variational Quantum Eigensolver (VQE) is a hybrid quantum-classical algorithm tailored for NISQ hardware [1]. Its goal is to find the approximate ground state energy of a Hamiltonian H based on the Rayleigh-Ritz variational principle: $E_0 \leq \langle\psi(\vec{\theta})|H|\psi(\vec{\theta})\rangle$. Here, $|\psi(\vec{\theta})\rangle$ is a trial state prepared by evolving an initial state (e.g., $|0\ldots0\rangle$) with a parameterized quantum circuit $U(\vec{\theta})$. VQE iteratively minimizes the energy expectation value $E(\vec{\theta}) = \langle\psi(\vec{\theta})|H|\psi(\vec{\theta})\rangle$ through a hybrid computational loop.

Its workflow is shown in Figure [1], and the specific algorithm steps are detailed in Algorithm [1].

Algorithm 1 Variational Quantum Eigensolver (VQE)

Initialize: The classical optimizer selects an initial set of parameters $\vec{\theta}_0$.
repeat
  //— Quantum Module —
  On the quantum processor, prepare the trial state $|\psi(\vec{\theta}_k)\rangle = U(\vec{\theta}_k)|0\ldots0\rangle$ using circuit $U(\vec{\theta}_k)$.
  For each sub-term H_i (Pauli string) in the Hamiltonian $H = \sum_i c_i H_i$:
    Repeat state preparation and measurement multiple times to estimate the expectation value $\langle H_i\rangle_k$.
  //— Classical Module —
  The classical computer collects all measurement results.
  Calculate the total energy expectation value: $E(\vec{\theta}_k) \leftarrow \sum_i c_i\langle H_i\rangle_k$.
  Pass $E(\vec{\theta}_k)$ and $\vec{\theta}_k$ to the classical optimizer to compute the next set of parameters: $\vec{\theta}_{k+1} \leftarrow$ optimize$(E(\vec{\theta}_k), \vec{\theta}_k)$.
  $k \leftarrow k+1$.
until Energy $E(\vec{\theta})$ converges
Return: Final energy $E(\vec{\theta}_k)$ and corresponding parameters $\vec{\theta}_k$.

This mode cleverly adapts to the limitations of NISQ hardware by decomposing the demand for long coherence times into multiple short-depth quantum circuit executions.

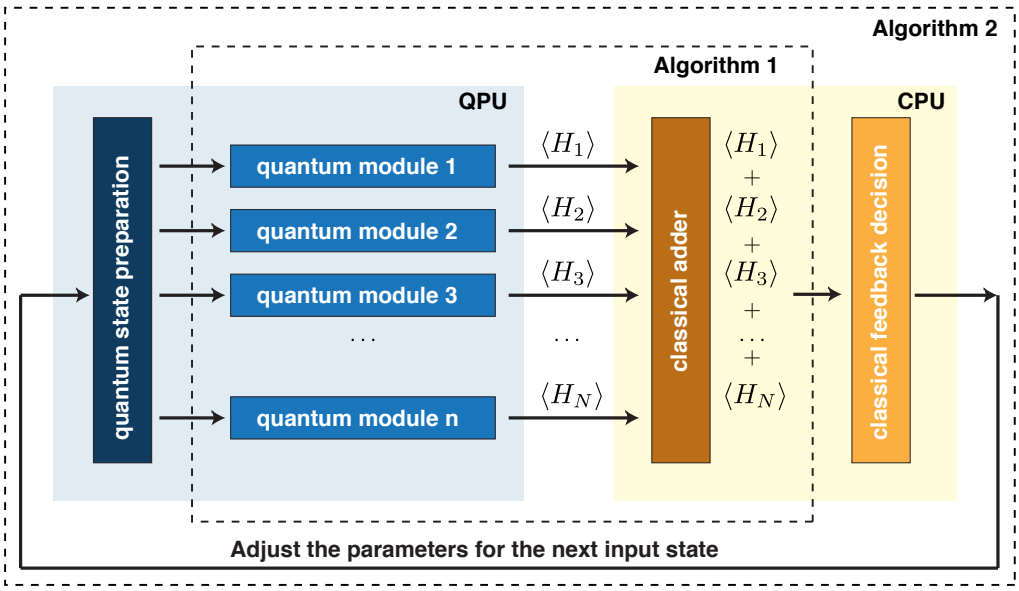

Figure 1: Hybrid architecture of the VQE algorithm. The quantum module is responsible for calculating the expectation values of each sub-term of the Hamiltonian, and the classical CPU is responsible for reconstructing the total energy and performing optimization [1].

1.1.2 Experimental Progress on NISQ Hardware

VQE has been widely experimentally validated on various mainstream NISQ hardware platforms. Early milestone work simulated the ground state energies of small molecules such as lithium hydride (LiH) and beryllium hydride (BeH_2) on superconducting qubits, successfully demonstrating the potential of VQE [2]. In recent years, significant progress has been made in both the scale and precision of experiments. For example, Google's Quantum AI team, using their Sycamore processor, accurately simulated the chemical bond breaking process of a 12-orbital hydrogen chain (H_{12}) through a 12-qubit VQE experiment, achieving chemical accuracy when compared to classical exact calculations [3]. The success of these cutting-edge experiments largely relies on the application of sophisticated error mitigation techniques to compensate for the impact of hardware noise on computational results. Nevertheless, these successful principle validations also highlight the constraints of noise and system scale on algorithm performance, which are core challenges that VQE must address to become practical.

1.1.3 Bottleneck Analysis

Despite its clever design, VQE's core ansatz faces an inherent contradiction: a trade-off between "Expressibility" and "Hardware Execution Fidelity" and trainability. To accurately solve a problem, the ansatz needs to be powerful enough to express the true ground state, which usually requires more complex circuits. However, on NISQ devices, deeper circuits accumulate more noise, thereby reducing the fidelity of the results. Neither problem-inspired ansätze (such as Unitary Coupled Cluster, UCC) [4], which aim to precisely describe physical systems, nor hardware-efficient ansätze (HEA) [2], designed to maximize hardware performance, have perfectly resolved this contradiction. Furthermore, some highly expressive ansätze can lead to the Barren Plateaus problem, where the gradient of the cost function vanishes in the parameter space, making optimization impossible [5]. This fundamental bottleneck is a key obstacle preventing VQE from becoming practical.

1.2 Schrödinger-Heisenberg Variational Quantum Algorithm: A New Approach

Facing the sharp contradiction between Expressivity and Fidelity in traditional VQA on NISQ hardware, where deeper circuits can express more complex quantum states but their fidelity

sharply declines due to noise accumulation, the academic community urgently needs a new paradigm that can decouple these two core elements. The Schrödinger-Heisenberg Variational Quantum Algorithm (SH-VQA) was proposed as an innovative framework in this context.

The core idea of SH-VQA is to decompose a deep, ideal quantum circuit into two parts. Among them, the deep and complex Heisenberg circuit T acts as a "virtual circuit" that is not executed on quantum hardware. Instead, it acts on the Hamiltonian H through efficient classical computation, transforming it into an isospectral but formally more complex $H_T = T^\dagger HT$. This step greatly enhances the overall expressivity of the algorithm without noise on a classical computer. Subsequently, a parameterized, very shallow Schrödinger circuit $U(\theta)$ acts as a "real circuit" that runs on a quantum computer. Its task is merely to find the ground state of the transformed Hamiltonian H_T under the premise of high fidelity. In this way, SH-VQA cleverly entrusts the heavy responsibility of pursuing high expressivity to noiseless classical computation, while reducing the requirements for quantum hardware to executing a simple shallow circuit, thereby breaking the "depth-fidelity" trade-off dilemma.

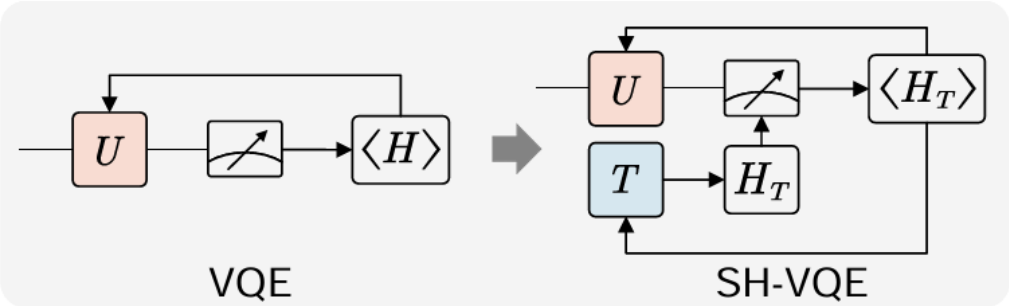

Figure 2: Comparison of VQE and SH-VQE algorithm structures. In SH-VQE, the transformed Hamiltonian H_T replaces the original H[6].

2 Principles of the Schrödinger-Heisenberg Variational Algorithm

2.1 Core Idea

The core idea of SH-VQA is to decompose a complete quantum evolution process $TU(\vec{\theta})|0\rangle$ into two parts:

- **Schrödinger Circuit** $U(\vec{\theta})$: This is a parameterized, relatively shallow quantum circuit, similar to circuits in traditional VQA. It runs on a real quantum computer, and its parameters $\vec{\theta}$ are optimizable. Because the circuit is shallow, it can be executed with high fidelity on NISQ hardware.
- **Heisenberg Circuit** T: This is a "virtual" circuit that can theoretically be very deep and complex. It is not executed on a quantum computer but acts as an operator on the Hamiltonian H of the problem to be solved. This transformation process is entirely performed on a classical computer, thus it is noiseless.

In standard VQE [1], we need to minimize the energy expectation value $E(\vec{\theta}) = \langle 0|U(\vec{\theta})^\dagger HU(\vec{\theta})|0\rangle$. In SH-VQA, the objective function becomes:

$$E(T, U) = \langle 0^{\otimes n}|U^\dagger T^\dagger HTU|0^{\otimes n}\rangle \tag{1}$$

We can define $T^\dagger HT$ as a new, transformed Hamiltonian H_T:

$$H_T = T^\dagger HT \tag{2}$$

Thus, the objective function can be written as:

$$E(T, U) = \langle 0^{\otimes n}|U^\dagger H_T U|0^{\otimes n}\rangle \tag{3}$$

Mathematically, H_T has exactly the same energy spectrum as the original H. However, by carefully designing a powerful virtual circuit T, we can transform a Hamiltonian H, whose ground state is difficult to prepare with a shallow circuit U, into a Hamiltonian H_T, whose ground state can be easily prepared with a shallow circuit U.

In this way, SH-VQA breaks the "depth-fidelity" contradiction. The total expressivity of the system is jointly determined by T and U, allowing it to explore a much wider Hilbert space than U alone (Figure [3]). The quantum computer only needs to execute the shallow circuit U and measure the expectation value of H_T, thereby ensuring high fidelity. Numerical experiments in the original paper [6] show that SH-VQA can achieve the expressivity that traditional VQE requires 40 depths to achieve with only 4 depths of the Schrödinger circuit, while reducing the hardware fidelity requirements by nearly two orders of magnitude.

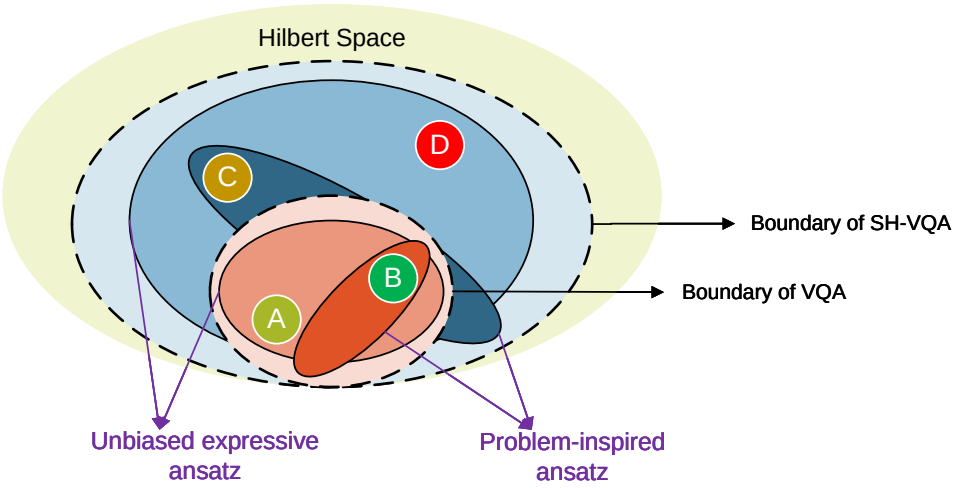

Figure 3: SH-VQA is more a general method for improving existing variational algorithms than a specific algorithm. For specific problems (A, B, C, D), we hope to solve them using real quantum circuits. We can see that for VQA, we can only possibly solve problems A and B, because C and D are beyond the boundaries of VQA. Conversely, we can solve all problems A, B, C, and D with SH-VQA, which demonstrates the advantage of our scheme. When actually running the algorithm, we need to construct quantum circuits. We know that for VQA, circuits are either unbiased with high expressivity but low trainability (e.g., hardware-efficient circuits), or biased with low expressivity but high trainability (e.g., UCC and HVA). Similarly, for SH-VQA, we can also expect to propose unbiased high-expressivity circuits or problem-inspired circuits [6].

2.2 Specific Algorithm

The SH-VQA algorithm can be decomposed into a classical hybrid quantum-classical optimization loop. The specific algorithm steps are detailed in Algorithm [2].

2.3 Clifford Circuits

In the SH-VQA framework, the choice of the virtual Heisenberg circuit T is crucial. It must be powerful enough to provide significant expressivity gains, while also ensuring that the classical computation of $H_T = T^\dagger H T$ and subsequent measurements are feasible. Clifford circuits are an ideal choice for achieving this balance.

Clifford circuits are quantum circuits composed of a series of gates selected from a specific set (e.g., Hadamard gates, S gates, and CNOT gates). Their most critical property is that Clifford circuits always map elements of the Pauli group back to the Pauli group itself (at

Algorithm 2 Schrödinger-Heisenberg Variational Quantum Algorithm (SH-VQA)

Initialize: The classical optimizer selects an initial set of parameters $(\vec{\theta}_0, \vec{\phi}_0, \vec{\alpha}_0)$.
repeat
    //— Classical Preprocessing Module —
    On the classical computer, calculate the transformed Hamiltonian based on the current Heisenberg circuit $T_k = T(\vec{\alpha}_k, \vec{\phi}_k)$:
        $H_{T_k} \leftarrow T_k^\dagger H T_k = \sum_j c_j P'_j$.
    //— Quantum Module —
    On the quantum processor, prepare the trial state $|\psi(\vec{\theta}_k)\rangle = U(\vec{\theta}_k)|0\dots0\rangle$ using the shallow circuit $U(\vec{\theta}_k)$.
    For each sub-term P'_j in the transformed Hamiltonian H_{T_k}:
        Repeat state preparation and measurement multiple times to estimate the expectation value $\langle P'_j\rangle_k$.
    //— Classical Optimization Module —
    The classical computer collects all measurement results.
    Calculate the total energy expectation value: $E_k \leftarrow \sum_j c_j \langle P'_j\rangle_k$.
    Pass E_k and $(\vec{\theta}_k, \vec{\phi}_k, \vec{\alpha}_k)$ to the classical optimizer to compute the next set of parameters:
        $(\vec{\theta}_{k+1}, \vec{\phi}_{k+1}, \vec{\alpha}_{k+1}) \leftarrow \text{optimize}(E_k, \vec{\theta}_k, \vec{\phi}_k, \vec{\alpha}_k)$.
    $k \leftarrow k+1$.
until Energy E converges
Return: Final energy E_k and corresponding parameters $(\vec{\theta}_k, \vec{\phi}_k, \vec{\alpha}_k)$.

most introducing a phase factor). This means that if a Pauli term in the original Hamiltonian is P_i, after transformation by a Clifford circuit T_c, $T_c^\dagger P_i T_c$ will still be a single Pauli term P'_i. Therefore, Clifford circuits do not increase the number of terms in the Hamiltonian, thus effectively controlling the growth of measurement overhead.

Furthermore, according to the Gottesman-Knill theorem [7], the effect of a quantum circuit composed of Clifford gates acting on Pauli operators can be efficiently simulated on a classical computer. This ensures that the computational complexity of the classical preprocessing step (calculating H_T) is polynomial, not exponential.

To introduce continuously optimizable parameters based on discrete Clifford gates, the original paper proposed that the T circuit consists of two parts: a pure Clifford circuit (responsible for building long-range, non-local entanglement structures) and a single-qubit gate layer (responsible for introducing continuous parameters $\vec{\phi}$). This hybrid structure achieves an excellent balance (trade-off) between enhancing expressivity and controlling computational cost.

3 Clifford Circuit Design for the Heisenberg Circuit T

3.1 Optimizing Clifford Circuits

In the SH-VQA framework, the final quantum state is given by $TU|0^{\otimes n}\rangle$, where U is a shallow Schrödinger circuit run on real quantum hardware, and T is a virtual Heisenberg circuit simulated on a classical computer, acting on the Hamiltonian. The design of the T circuit is crucial, as it directly determines whether the algorithm can effectively expand the range of the accessible Hilbert space without increasing the depth of the real circuit, thereby finding better solutions than traditional VQE.

A well-designed T circuit is key to performance improvement. This is clearly demonstrated in Figure 3(c) of the paper [6] (numerical experiments): when the optimal "fully connected graph" was chosen as the Clifford part of T for the 8-qubit XXZ model, its solution accuracy significantly surpassed that of traditional VQE without a T circuit (Figure [4c]); however, if an unsuitable T circuit was chosen, its performance could even be worse than traditional VQE. This indicates that the introduction of T is a double-edged sword, and its structure must match

the specific problem.

However, designing an optimal T circuit faces significant challenges. The T circuit is primarily composed of Clifford circuits. For n qubits, the size of the Clifford group grows super-exponentially, meaning that the space of possible Clifford circuit structures is extremely vast, making brute-force search or random selection completely infeasible. Therefore, an efficient strategy must be developed to constrain the search space and systematically find approximately optimal T circuit structures.

3.2 Search Space Reduction Based on Problem Symmetry: Taking the XXZ Model as an Example

To address the above challenges, the paper proposes a "problem-specific" search space reduction strategy for solving the XXZ spin chain model (Hamiltonian H see (4)). The core idea of this strategy is to **utilize the symmetry of the Hamiltonian itself to guide and constrain the structure of the Clifford part in the T circuit**. This Hamiltonian has translational invariance (TI) under periodic boundary conditions. Based on this, the authors infer that the optimal T circuit is likely to also possess similar symmetry, thereby significantly reducing the search range from the entire Clifford group to a subset with translational invariance.

$$H_{XXZ} = \sum_{i=1}^{n} \Big(\sigma_i^x \sigma_{i+1}^x + \sigma_i^y \sigma_{i+1}^y + \Delta\, \sigma_i^z \sigma_{i+1}^z \Big) \tag{4}$$

The specific implementation details of this method are as follows:

Through this method, for the 8-qubit XXZ model's T circuit search problem, the structural search space was successfully reduced from the astronomical size of the Clifford group to only $2^4 = 16$ candidate structures with specific symmetries. This is a huge and decisive simplification.

3.3 Probabilistic Architecture Search

Although the search space has been greatly reduced, for larger-scale problems (e.g., 2^{15} candidates for $n = 30$), testing them one by one is still infeasible. To this end, the paper draws on the idea of Differentiable Neural Architecture Search (NAS) [8], adopting a probabilistic method to automatically and efficiently search for the optimal circuit structure. The specific algorithm steps are detailed in Algorithm [**??**], showing the results of different connection methods under Probabilistic Architecture Search for SH-VQE (Figure [4b]).

3.4 Overall Idea for Clifford Circuit Construction in SH-VQA Algorithm: The XXZ Model Example

In short, our method can be summarized in two steps: First, by utilizing the translational symmetry of the XXZ Hamiltonian, the vast Clifford structure space, originally containing $n(n-1)/2$ CZ edges, is compressed into a symmetric subspace composed of only $\lfloor n/2 \rfloor$ "distance-j" basic graphs. This reduces the search scale from $2^{n(n-1)/2}$ to $2^{\lfloor n/2 \rfloor}$ (e.g., for $n = 16$, only 256 candidate graphs remain). Subsequently, within this greatly simplified space, Probabilistic Differentiable Architecture Search (PAS) is applied: a trainable logits vector $\vec{\alpha}$ is assigned to each basic graph, which is then converted into an on/off probability via softmax($\vec{\alpha}$). During the training loop, Clifford graph circuits are sampled according to this distribution, and their energy is evaluated together with continuous gate parameters $\vec{\theta}, \vec{\phi}$. Adam-SPSA is used to synchronously update $\vec{\alpha}, \vec{\theta}, \vec{\phi}$. As iterations progress, the probabilities naturally collapse to 0/1, ultimately automatically selecting the deterministic optimal Clifford structure (the 8-qubit example converges to the "1111" fully connected graph). The immense capability of the SH-VQA algorithm is fully demonstrated in the comparison in Figure [4d].

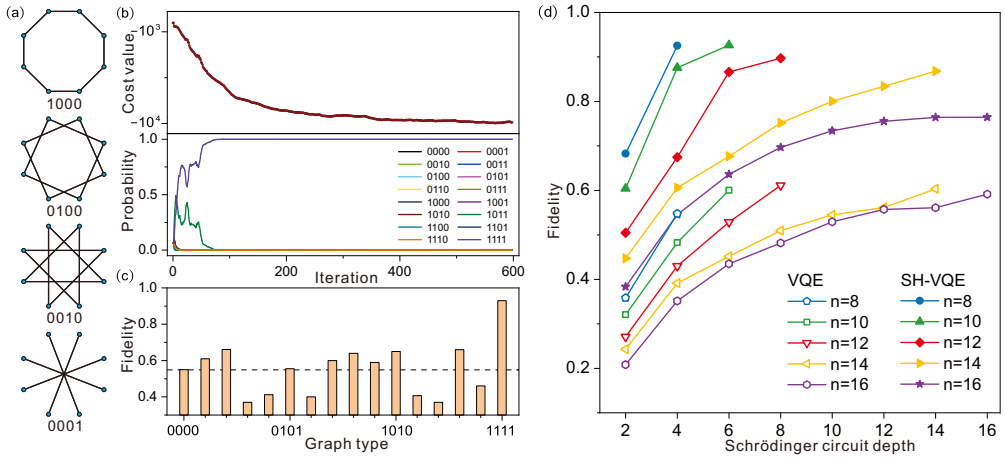

Figure 4: Clifford circuit for the XXZ model[6]

4 Algorithm Capability Comparison

4.1 Expressivity

The expressivity of a quantum circuit determines the "coverage" of the set of states it can generate in Hilbert space (Figure [5a]). If the expressivity is insufficient, even the most precise optimization will struggle to approximate the target state. The paper uses the t-design framework to quantify expressivity and defines

$$\Delta_t = \log\Big(E_{\text{Haar}}\big[\text{Tr}(\rho_{n/2}^t)\big]\Big) - \log\Big(E_{\text{SH}}\big[\text{Tr}(\rho_{n/2}^t)\big]\Big), \tag{5}$$

where E_{Haar} and E_{SH} represent the average over Haar random states and SH-VQE output states, respectively. $\Delta_t \to 0$ means that the circuit family has formed a t-design—i.e., it is equivalent to the Haar distribution in the first t moments. The reasons for choosing this metric are: *(i)* it has a known lower bound with circuit depth, directly reflecting hardware resource requirements; *(ii)* t-design is also related to various performance measures such as entanglement entropy and randomized benchmarking, thus having uniformity. Figure [5b] compares the convergence behavior of Δ_t for the two algorithms at different depths.

4.2 Depth and Gate Fidelity Requirements

In the NISQ era, quantum circuit depth and gate fidelity are hard constraints on experimental feasibility. Measuring "how shallow a quantum circuit can achieve a given expressivity" helps clarify the benefits of reducing hardware gate count and error thresholds. By mapping the "equivalent VQE depth required to reach the same order t-design" to the actual SH-VQE depth, and then calculating the difference in two-qubit gate fidelity requirements based on this, the direct value of this metric in experimental planning can be demonstrated.

4.3 Trainability

High expressivity can lead to the barren plateau phenomenon. Evaluating trainability ensures that the algorithm can not only "represent" but also "be found." By analyzing the inverse relationship between expressivity and gradient variance, strategies such as restricted Clifford pools and alternating optimization are introduced to maintain gradient scale, and this is verified in the reproduction of XXZ chains and molecular ground states. The necessity of trainability measurement lies in: without this verification, the benefits of increased expressivity might be completely offset by optimization difficulty.

5 Future Outlook

When truly running these algorithms, we must construct specific ansätze. We know that in traditional VQA, there are two common types of ansätze: unbiased high-expressivity ansätze

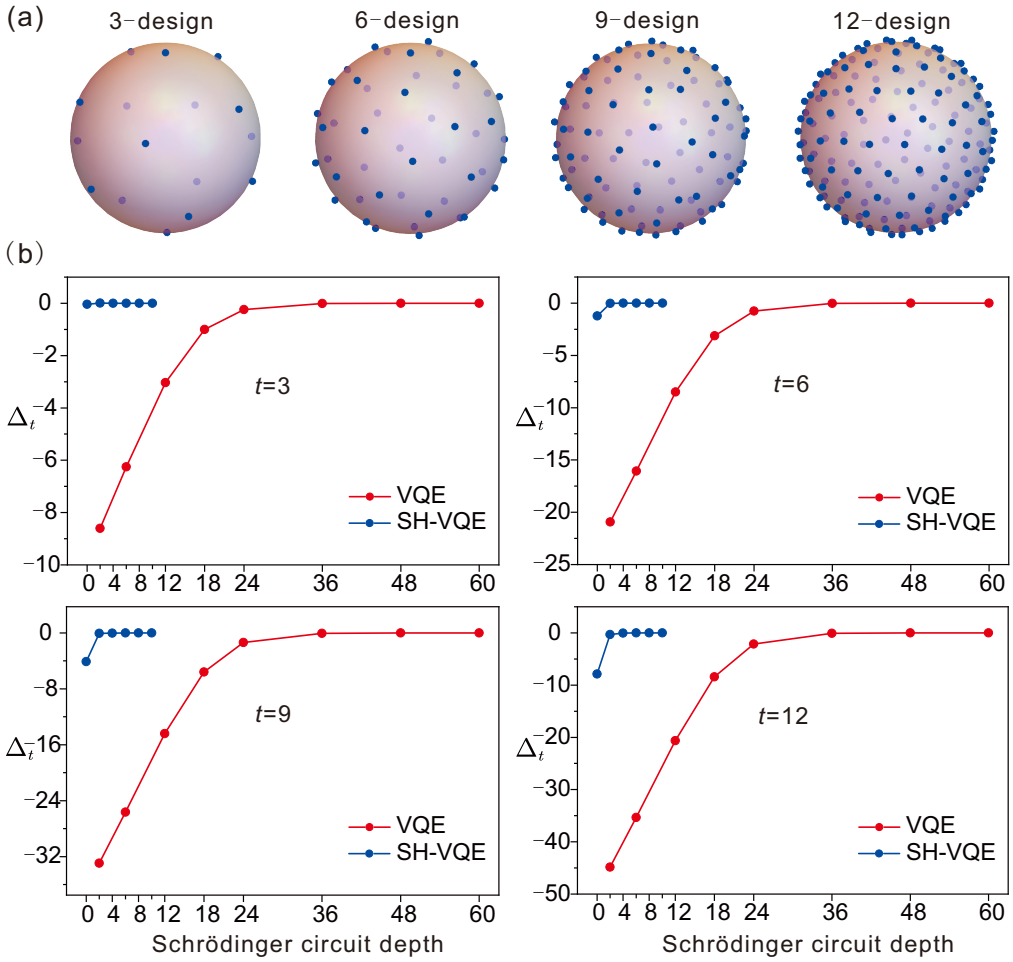

Figure 5: Illustrate the stronger expressivity of SH-VQA[6]

(e.g., hardware-efficient ansätze), which have high expressivity but low trainability; and biased problem-oriented ansätze (e.g., UCC and HVA), which have relatively low expressivity but high trainability. The same structure can be analogously applied to SH-VQA. We can construct for SH-VQA: an unbiased, highly expressive but poorly trainable ansatz; or a biased, problem-driven, well-trainable ansatz. First, use a shallow U with verified trainability (such as a simplified UCC/HVA variant) as the base, then carefully select a small set of complementary Clifford-T transformations based on the problem structure and hardware coupling graph, and introduce alternating or adaptive optimization strategies to dynamically adjust the size of the T pool; simultaneously, combine sparse Pauli measurement compression with gradient diagnostic tools to ensure that the number of measurement rounds and classical update overhead are predictable and convergent. Through this "first determine U, then generate T, and real-time convergence monitoring" phased process, medium-precision verification of small chemical molecules or one-dimensional spin chains can be completed on 100-200 qubit NISQ devices.

References

[1] Alberto Peruzzo et al. "A variational eigenvalue solver on a quantum processor". In: *Nature Communications* 5.1 (2014), p. 4213. DOI: 10.1038/ncomms5213.

[2] Abhinav Kandala et al. "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets". In: *Nature* 549.7671 (2017), pp. 242–246. DOI: 10.1038/nature23879.

[3] Google AI Quantum and Collaborators. "Hartree-Fock on a superconducting qubit quantum computer". In: *Science* 369.6507 (2020), pp. 1084–1089. DOI: 10.1126/science.abb9811.

[4] Abhishek Anand et al. *A review of the unitary coupled cluster ansatz*. 2022. arXiv: 2109.15176 [quant-ph].

[5] Jarrod R. McClean et al. "Barren plateaus in quantum neural network training landscapes". In: *Nature Communications* 9.1 (2018), p. 4812. DOI: 10.1038/s41467-018-07090-4.

[6] Zhong-Xia Shang et al. "Schrödinger-Heisenberg Variational Quantum Algorithms". In: *Physical Review Letters* 131.6 (Aug. 2023), p. 060406. DOI: 10.1103/PhysRevLett.131.060406.

[7] Daniel Gottesman. *The Heisenberg Representation of Quantum Computers*. 1998. arXiv: quant-ph/9807006 [quant-ph].

[8] Shuize Zhang et al. "Differentiable quantum architecture search". In: *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE. 2020, pp. 257–265.