

# Efficient and Robust Parameter Optimization of the Unitary Coupled-Cluster Ansatz

Weitang Li,\* Yufei Ge, Shi-Xin Zhang, Yu-Qin Chen, and Shengyu Zhang



Cite This: *J. Chem. Theory Comput.* 2024, 20, 3683–3696



Read Online

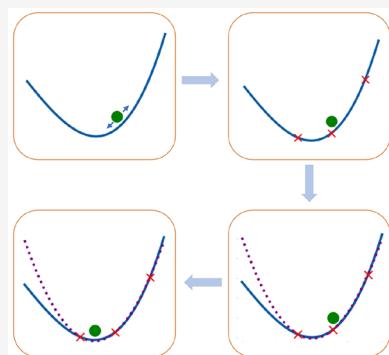
ACCESS |

Metrics & More

Article Recommendations

Supporting Information

**ABSTRACT:** The variational quantum eigensolver (VQE) framework has been instrumental in advancing near-term quantum algorithms. However, parameter optimization remains a significant bottleneck for VQE, requiring a large number of measurements for successful algorithm execution. In this paper, we propose sequential optimization with approximate parabola (SOAP) as an efficient and robust optimizer specifically designed for parameter optimization of the unitary coupled-cluster ansatz on quantum computers. SOAP leverages sequential optimization and approximates the energy landscape as quadratic functions, minimizing the number of energy evaluations required to optimize each parameter. To capture parameter correlations, SOAP incorporates the average direction from previous iterations into the optimization direction set. Numerical benchmark studies on molecular systems demonstrate that SOAP achieves significantly faster convergence and greater robustness to noise compared with traditional optimization methods. Furthermore, numerical simulations of up to 20 qubits reveal that SOAP scales well with the number of parameters in the ansatz. The exceptional performance of SOAP is further validated through experiments on a superconducting quantum computer using a 2-qubit model system.



## 1. INTRODUCTION

Chemistry has emerged as a promising field for the application of noisy intermediate-scale quantum (NISQ) devices.<sup>1–4</sup> The key component for the success is the variational quantum eigensolver (VQE) framework, which leverages quantum computers as specialized devices for storing the wave function of molecular systems.<sup>5–7</sup> More specifically, a quantum circuit, characterized by a parametrized unitary transformation  $\hat{U}(\vec{\theta})$  and the initial state  $|\phi_0\rangle$ , is viewed as an ansatz  $|\phi(\vec{\theta})\rangle = \hat{U}(\vec{\theta})|\phi_0\rangle$  analog to the ansatz in classical computational chemistry. Leveraging the Rayleigh-Ritz variational principle, the parameters in the quantum circuit  $\vec{\theta}$  are optimized on a classical computer using either gradient-based or gradient-free optimizer, with the energy expectation  $E(\vec{\theta}) = \langle\phi|\hat{H}|\phi\rangle$  measured on quantum computers.

Two distinct types of ansatz have been explored in quantum computational chemistry. The disentangled unitary coupled-cluster (UCC) ansatz<sup>8</sup> uses the Hartree–Fock (HF) state as the initial state and applies the UCC factors  $e^{\theta_i \hat{G}_i}$  to the initial state, where  $\vec{\theta}$  is the circuit parameter and  $\{\hat{G}_i\}$  is the anti-Hermitian cluster operator.  $\hat{G}_i$  is composed of excitation operators, and it is mostly commonly truncated to single and double excitation. The UCC ansatz is the most widely studied ansatz due to its accuracy and close relation with classical computational chemistry, yet its circuit depth is usually too deep to be implemented on currently available quantum devices.<sup>9,10</sup> To address this issue, several variants of the UCC ansatz have been proposed. For example, adaptive derivative-assembled pseudo-Trotter ansatz VQE (ADAPT-VQE)<sup>11</sup> and

paired UCC with double excitations<sup>12</sup> aim to reduce the circuit depth while preserving the clear chemical structure of the UCC ansatz. On the other hand, the hardware-efficient ansatz (HEA) features shallow circuit depth and finds widespread application in hardware experiments.<sup>13</sup> However, its scalability is hindered by optimization challenges, which need to be addressed for broader practical application.<sup>14</sup> Furthermore, there is a growing interest in integrating the hardware-efficient ansatz with chemical and physical insights.<sup>15,16</sup>

One of the biggest bottlenecks for VQE is the excessive number of measurements required to achieve a high accuracy. Various basis rotation methods have been proposed to mitigate this issue.<sup>17–19</sup> Meanwhile, the pursuit of an efficient parameter optimization subroutine has been a central focus in recent research.<sup>20–25</sup> Parameter optimization for quantum circuits presents notable challenges due to the large number of parameters involved and the necessity to find the global minimum. Moreover, efficient computation of energy gradients on a quantum computer remains unfeasible. Specifically, computing all the gradients requires  $O(N)$  circuit evaluations, where  $N$  is the number of parameters, using the parameter-shift

Received: February 6, 2024

Revised: April 11, 2024

Accepted: April 11, 2024

Published: April 19, 2024



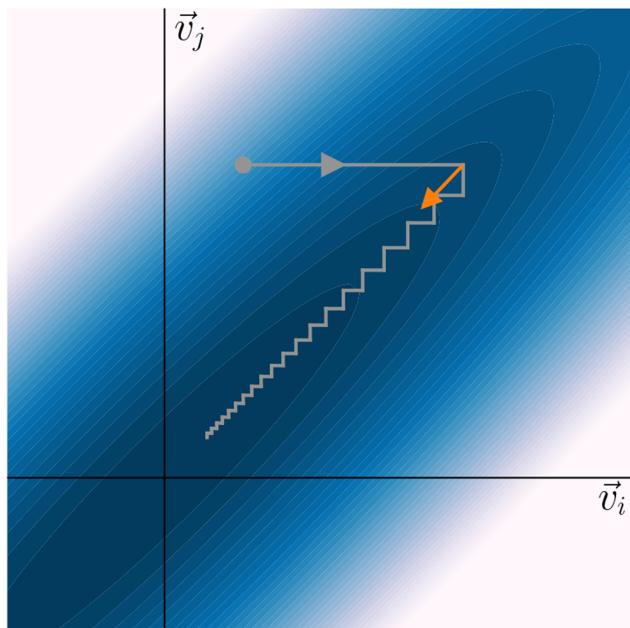
rule.<sup>26,27</sup> In contrast, classical computers can leverage the back-propagation algorithm to compute all gradients with a time complexity of  $O(1)$ .<sup>28</sup> We note that the time complexity is relative to the computation of the target function. In other words, computing all gradients requires roughly the same amount of time as evaluating the function itself. Recently, a quantum gradient algorithm with  $O(1)$  scaling has also been reported.<sup>29</sup> However, the algorithm imposes restrictions on the circuit structure, and its applicability to chemical problems remains to be explored. Another fundamental distinction between quantum computers and classical computers lies in the inherent uncertainty of measurement outcomes in quantum systems, emphasizing the need for noise-resilient optimizers. The difficulty associated with parameter optimization is manifested by the growing popularity of preoptimizing the circuit parameters on a classical simulator in recent hardware experiments.<sup>30–32</sup>

Unfortunately, there has been relatively limited research dedicated to the development of efficient parameter optimization algorithms for quantum computational chemistry. In most quantum computational chemistry research, the prevailing practice involves employing general-purpose optimizers implemented in standard packages such as SciPy.<sup>33</sup> For instance, the Limited-memory Broyden–Fletcher–Goldfarb–Shanno with bounds (L-BFGS-B) algorithm<sup>34</sup> demonstrates efficiency in numerical simulations where the gradients are readily available.<sup>35–37</sup> Conversely, the constrained optimization by linear approximation (COBYLA) algorithm<sup>38</sup> and the simultaneous perturbation gradient approximation (SPSA) algorithm<sup>39</sup> are popular choices for hardware experiments as they do not rely on gradient information.<sup>40–42</sup> However, gradient-free optimizers often suffer from slow convergence rates, further exacerbating the already high number of measurements required for VQE. These optimizers are inefficient, because they fail to fully exploit the unique mathematical structure of unitary quantum circuits and are typically sensitive to the measurement uncertainty inherent in quantum computers. Consequently, there is an urgent need to develop efficient, gradient-free, and noise-resilient optimization algorithms specifically tailored to the unique characteristics of quantum computers and the context of computational chemistry.

Recently, a promising optimizer called NFT or rotosolve optimizer<sup>43–45</sup> has emerged, specifically designed for HEA circuits. This method employs a parameter optimization approach where one parameter is optimized at a time while keeping the other parameters fixed, resembling the basic iteration procedure in Powell's method.<sup>46–48</sup> By exploiting the fact that the energy expectation as a function of a single parameter can be expressed using a sine function, the NFT optimizer constructs an analytical expression for the energy expectation. This construction requires only three independent energy evaluations, eliminating the need for computing gradients. The parameter is then directly optimized to the minimum, and the whole process can be considered as a variant of the usual line search procedures. One notable advantage of the NFT optimizer is its exceptional tolerance to statistical measurement noise. This is achieved by avoiding comparisons of similar energy values, even when the circuit parameters are close to the minimum. However, a limitation of this method is that it is most efficient when each parameter is associated with a single-qubit rotation gate  $\hat{R}_j(\theta) = e^{-i(\theta/2)\hat{A}_j}$ ,

where  $\hat{A}_j \in \{\hat{X}, \hat{Y}, \hat{Z}, \hat{I}\}$ . Thus, the NFT optimizer cannot be readily applied to the UCC ansatz without introducing significant overhead.<sup>49</sup> In the UCC ansatz, a UCC factor  $e^{\hat{G}}$  will be compiled into a quantum circuit in which multiple single qubit rotation gates share the same parameter  $\theta_j$ . The number of parameter-sharing gates is  $2^{2K-1}$ , if the order of the excitation operator is  $K$ . Furthermore, the sequential optimization approach employed by the NFT optimizer neglects the “correlation” between the parameters. In cases where the Hessian matrix  $\frac{\partial E}{\partial \theta_i \partial \theta_j}$  has large nondiagonal elements,

the line search procedure only takes small steps toward the optimal parameter vector  $\vec{\theta}$ . This can result in a large number of steps required for convergence, as illustrated in Figure 1. Consequently, sequential optimization algorithms may face challenges in converging efficiently for complex chemical systems.<sup>50</sup>



**Figure 1.** Schematic diagram showing the optimization trajectory of sequential optimization methods when there are large off-diagonal elements in the Hessian matrix. Traditional sequential optimization methods start from the gray dot and proceed along the gray line. SOAP identifies a more promising average direction marked by the orange arrow which significantly accelerates the convergence.

In this paper, we address these challenges with a parameter optimization algorithm tailored to the UCC ansatz. Our method, termed sequential optimization with approximate parabola (SOAP), shares similarities with the NFT optimizer, as it employs a sequential line-search procedure over a list of directions. For each direction, SOAP obtains the minimum by fitting an approximate parabola using 2 to 4 energy evaluations. The key insight enabling this simple parabola fit is that for the UCC ansatz; the initial guess is already in proximity to the minimum. More specifically, the initial guess is usually generated by second-order Møller–Plesset perturbation theory,<sup>51</sup> one of the most useful computational methods beyond the HF approximation. By sacrificing the exact fit of the complicated energy function for each parameter, SOAP is also capable of performing a line search over the average direction of the last iteration. This feature significantly

accelerates convergence when there are large off-diagonal elements of the Hessian matrix.

To evaluate the performance of SOAP, we conducted extensive benchmark studies on systems ranging up to 20 qubits. The results demonstrate that SOAP exhibits superior efficiency and noise resilience compared to traditional optimization methods. Furthermore, our numerical data indicate that SOAP scales well with the number of parameters in the circuit. For a UCC circuit with  $N$  parameters, on average, SOAP reaches convergences by  $3N$  energy evaluations. Our findings are verified on a superconducting quantum computer based on a simplified 2-electron, 2-orbital model.

## 2. METHODOLOGY

**2.1. Review and Motivation.** Let us begin by reviewing the NFT method. Consider a quantum circuit parametrized by  $\vec{\theta}$ , where for any parameter  $\theta_j$ , the circuit can be expressed as

$$|\phi(\vec{\theta})\rangle = \hat{W}_j \hat{R}_j(\theta_j) \hat{V}_j |\phi_0\rangle \quad (1)$$

Here,  $\hat{W}_j$  represents the part of the circuit that depends on the parameters  $\{\theta_k | k < j\}$ , and  $\hat{V}_j$  represents the part that depends on  $\{\theta_k | k > j\}$ . The rotation gates can be written as

$$\hat{R}_j(\theta_j) = e^{-i\theta_j \hat{A}_j} = \cos \theta_j - i \hat{A}_j \sin \theta_j \quad (2)$$

Here, we intentionally absorbed the  $\frac{1}{2}$  factor into  $\theta_j$  for notation simplicity. It follows that the energy expectation as a function of a single parameter  $\theta_j$  can be expressed as

$$\begin{aligned} E(\theta_j) &= a \cos^2 \theta_j + 2b \sin \theta_j \cos \theta_j + c \sin^2 \theta_j \\ &= \sqrt{b^2 + \left(\frac{a-c}{2}\right)^2} \sin\left(2\theta_j + \arctan \frac{a-c}{2b}\right) \\ &\quad + \frac{a+c}{2} \end{aligned} \quad (3)$$

where  $a$ ,  $b$ , and  $c$  are some coefficients that depend on  $\hat{W}_j$  and  $\hat{V}_j$ . The minimum of  $E(\theta_j)$  locates at  $\theta_j = \frac{3}{4}\pi - \frac{1}{2}\arctan \frac{a-c}{2b}$ . By performing three independent energy evaluations at different  $\theta_j$  values, the coefficients  $a$ ,  $b$ , and  $c$  can be determined. Subsequently, the parameter  $\theta_j$  is optimized to its optimal value, while keeping the other parameters fixed. The NFT method then proceeds to optimize the next parameter until certain convergence criteria are met.

The NFT optimization method described above can be extended to the UCC ansatz by expressing the UCC factors as<sup>52,53</sup>

$$e^{\theta_j G_j} = (1 - \cos \theta_j) \hat{G}_j^2 + \sin \theta_j \hat{G}_j + 1 \quad (4)$$

The energy expectation as a function of  $\theta_j$  can then be written as

$$\begin{aligned} E(\theta_j) &= a_1 \cos^2 \theta_j + a_2 \sin^2 \theta_j + a_3 \sin \theta_j \cos \theta_j + a_4 \sin \theta_j \\ &\quad + a_5 \cos \theta_j \end{aligned} \quad (5)$$

where  $\{a_i | 1 \leq i \leq 5\}$  is the set of coefficients to be determined. Note that the constant term  $a_6$  has been merged into  $a_1 \cos^2 \theta_j + a_2 \sin^2 \theta_j$  by  $a_6 = a_6 \cos^2 \theta_j + a_6 \sin^2 \theta_j$ . Obtaining these coefficients requires 5 independent energy evaluations at different  $\theta_j$  values. Thus, it seems the NFT optimizer can be migrated to the UCC ansatz straightforwardly. However, in

closed-shell systems, which are commonly targeted by the UCC ansatz, UCC factors with complementary spin components are expected to share the same parameter.<sup>54,55</sup> In such cases, the number of energy evaluations required for an exact fit of the energy landscape increases to 14. This is because the analytic expression for  $E$  with respect to a single variable  $\theta_j$  has more parameters to be determined from measurements if two UCC factors share the same parameter  $\theta_j$ . Moreover, due to the increased complexity of eq 5 compared to eq 3, eq 5 is more prone to overfitting due to the presence of statistical measurement noise and the limited number of data points. Therefore, it is desirable to develop a more robust and efficient scheme for sequential optimization of the UCC ansatz. It is worth noting that the more complex expansion in eq 4 and the parameter-sharing scheme also complicate the evaluation of the energy gradients in the UCC ansatz, which prompts the development of gradient-free optimizers.

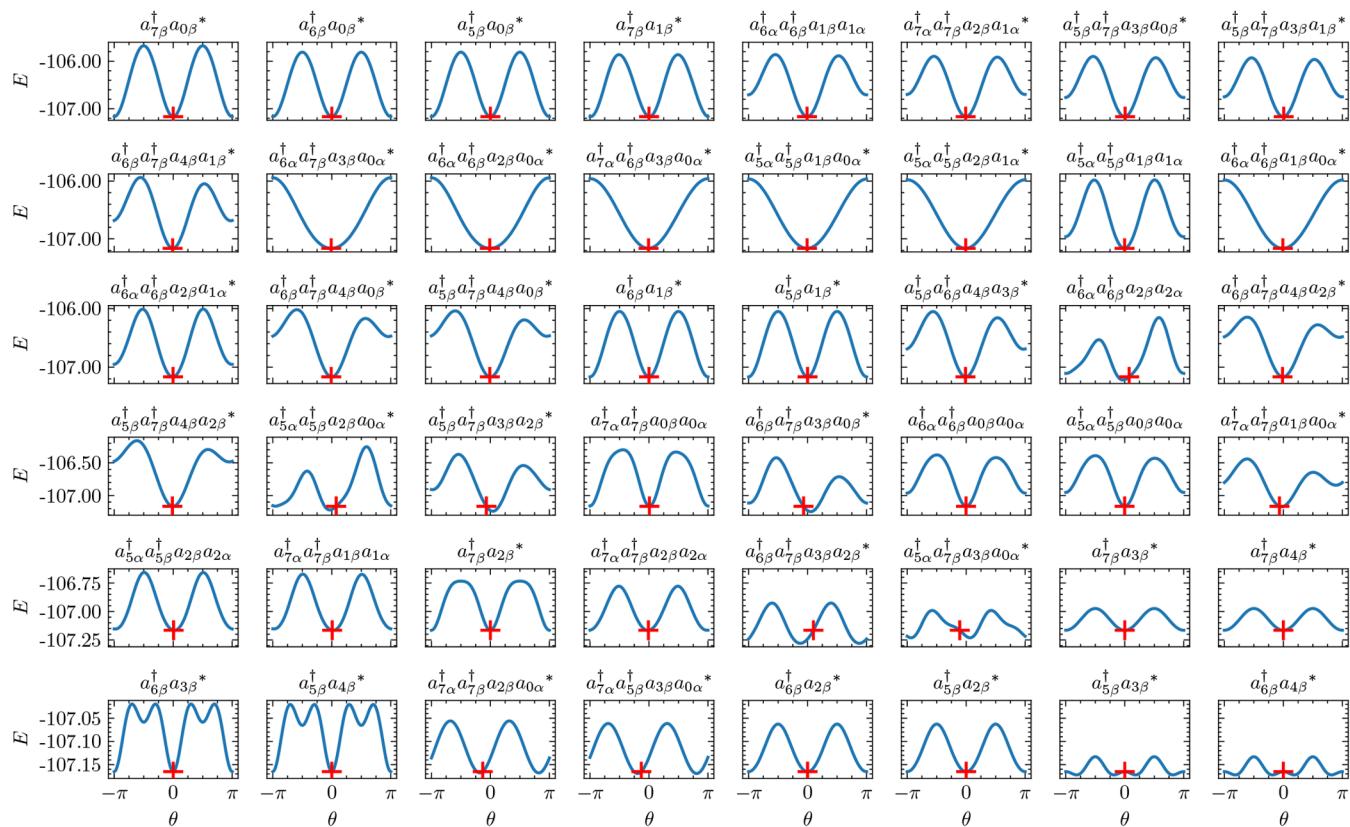
The SOAP method is based on the intuition that the initial guess for the UCC ansatz is very close to a highly favorable local minimum if not the global minimum. The most straightforward approach to initialize the UCC ansatz is to set  $\vec{\theta} = 0$ , resulting in  $|\phi(\vec{\theta})\rangle$  being the Hartree–Fock state. A more efficient approach for parameter initialization is to use second-order Møller–Plesset perturbation theory (MP2) to generate the initial guess.<sup>51</sup> If  $\theta_{ij}^{ab}$  is associated with the excitation from orbital  $ij$  to orbital  $ab$ , it is initialized as

$$\theta_{ij}^{ab} = \frac{h_{ijba} - h_{ijab}}{\varepsilon_i + \varepsilon_j - \varepsilon_a - \varepsilon_b} \quad (6)$$

where  $h_{pqrs}$  represents the two-electron integral and  $\varepsilon_p$  is the HF orbital energy. The amplitudes of single excitations are set to zero. In the weak correlation regime, the difference between the full configuration interaction (FCI) energy  $E_{\text{FCI}}$  and the HF energy  $E_{\text{HF}}$ , known as the correlation energy  $E_{\text{corr}}$ , is much smaller than the energy scale of the Hamiltonian. In other words, if the quantum circuit is initialized randomly, the resulting energy would be much higher than that of  $E_{\text{HF}}$ . From a wave function perspective, the HF state is by definition the most dominating configuration of the ground state wave function in weakly correlated systems, with MP2 serving as a good correction to the HF state. In the strong correlation regime, the validity of the MP2 initial guess is an open question. Nevertheless, as we shall observe from the numerical experiments, the MP2 initialization strategy proves to be effective in the strong correlation regime. It is important to note that applying the UCC ansatz to strongly correlated systems is not advocated in the first place, due to its single-reference nature. For strongly correlated systems, UCC in combination with CASSCF is preferred.<sup>56,57</sup> Compared with traditional UCC, UCC-CASSCF improves the overlap between the initial state and the desired ground state, which justifies the application of SOAP to perform parameter optimization. We note that there are also other ways to do parameter initialization,<sup>58</sup> and we believe that the optimization problem for VQE could be solved through the synergistic development of both initialization and optimization methods.

If we assume  $\vec{\theta}$  is sufficiently close to the minimum, the energy function can be approximated by a parabola at arbitrary direction  $\vec{v}$

$$E(\vec{\theta} + x\vec{v}) \approx ax^2 + bx + c \quad (7)$$



**Figure 2.** Energy landscape with respect to a single parameter for the UCC ansatz at the initial point generated by MP2. The  $\text{N}_2$  molecule with 2.0 Å bond length is used as the model system. The energy unit is Hartree. The red crossing represents the MP2-predicted amplitudes and the corresponding energy. The excitation operators associated with each parameter are shown in the title. The Hermitian conjugation is omitted for brevity. If an asterisk is shown, it represents two excitation operators sharing the same parameter.

The coefficients  $a$ ,  $b$ , and  $c$  can be determined by 3 energy evaluations and then an offset  $-\frac{b}{2a}\vec{v}$  is added to  $\vec{\theta}$  to move it to the minimum along the direction  $\vec{v}$ . By fitting a parabola, we have reduced the number of measurements required and make the algorithm more robust to statistical noise. Such an approximation also allows more flexibility on how to choose  $\vec{v}$ , which will be discussed in the next section.

**2.2. The SOAP Algorithm.** We first describe the basic iteration procedure for SOAP before delving into the treatment of corner cases and minor optimizations within SOAP. The complete algorithm is presented in Algorithm 2. We first initialize the direction vector  $\vec{v}_i = \vec{e}_p$ , where  $\vec{e}_i$  is a unit vector in the parameter space. In each iteration, for each direction vector  $\vec{v}_p$ , the energies along  $\vec{\theta}_{i-1} + x\vec{v}_i$  are measured. Here,  $\vec{\theta}_{i-1}$  is the optimized parameter vector from the previous optimization step along the direction  $\vec{v}_{i-1}$ . A parabola fit based on eq 7 is then performed by using the measured energies. Subsequently, the circuit parameters are optimized by setting  $\vec{\theta}_i$  to the minimum of the parabola, given by  $\vec{\theta}_{i-1} - \frac{b}{2a}\vec{v}_i$ . This process is in essence a simplified line search procedure over the direction  $\vec{v}_i$  based on the heuristics of the UCC ansatz. The algorithm then proceeds to the next direction vector  $\vec{v}_{i+1}$ . Once the entire list of direction vectors  $\mathcal{V} = [\vec{v}_1, \vec{v}_2, \dots, \vec{v}_N]$  has been traversed, the SOAP algorithm then proceeds to the next iteration until certain convergence criteria are satisfied. The initial order of  $\mathcal{V}$  is determined by ensuring  $|\theta_i| > |\theta_{i+1}|$  where  $\theta_i$  is  $i$ th element of the initial  $\vec{\theta}$  determined by the MP2 amplitudes.

Let us now delve into more details about the line search procedure using an approximate parabola based on eq 7, denoted as LSAP in the following discussion. To begin, we define a small scalar  $u$  and let

$$x_m = mu, m \in \mathbb{Z} \quad (8)$$

For example,  $x_0 = 0$ ,  $x_1 = u$ . We then define

$$y_m = E(\vec{\theta}_{i-1} + x_m \vec{v}_i) \quad (9)$$

which is measured on a quantum computer. Since we assume  $\vec{\theta}_{i-1}$  is close to the minimum, we use the set  $\mathcal{F} = \{(x_i, y_i) | i = -1, 0, 1\}$  to construct the parabola. Note that only  $y_{-1}$  and  $y_1$  should be measured on a quantum computer since  $y_0$  is available from the previous iteration.

Next, we turn our attention to the selection of  $u$ . Based on the landscape shown in Figure 2, setting  $u = 0.1$  is a reasonable initial value. In the Supporting Information, we have included benchmark data that demonstrates that the performance of SOAP is not significantly affected by the choice of  $u$  within the range of 0.05 to 0.20. Although it may be tempting to dynamically adjust  $u$  according to the landscape during optimization, we refrain from such enhancements for the following reasons. First, quantum computers are subject to measurement uncertainty, and shrinking  $u$  will decrease the signal-to-noise ratio. As a result, shrinking  $u$  around the minimum does not necessarily lead to a more accurate estimation of  $\vec{\theta}_i$ . Second, enlarging  $u$  does not offer significant potential either, since the landscape of  $E(\vec{\theta}_{i-1} + x\vec{v}_i)$  is periodic

with a period of  $2\pi$  and a good parabola fit can only be expected within a span of  $\sim\pi/2$ . Lastly, we prefer keeping SOAP a simple and robust algorithm that is straightforward to implement and extend. For these reasons, we use  $u = 0.1$  throughout the paper.

It is worth mentioning that the idea of parabola fitting has also been proposed in a recent work.<sup>59</sup> In each iteration, their method sets the optimization direction to the direction of the gradients and then employs parabola fitting to determine the minimum. Therefore, their method is gradient-based, while our method is gradient-free. Newton's method for optimization can also be considered as a second-order approximation of the optimization landscape, in which the Taylor expansion is constructed based on local first- and second-order derivatives. In contrast, by setting  $u = 0.1$ , the LSAP procedure differentiates from Newton's method by exploiting nonlocal landscape information and thereby becomes more robust to measurement noise.

In the LSAP procedure, certain corner cases deserve careful consideration. Because the fitting relies on only three data points, the fitted energy landscape can be severely distorted if  $x_0$  is significantly distant from the minimum. Although this scenario is relatively rare, it is inevitable for strongly correlated systems, as shown in Figure 2. The quality of the parabola fit can be roughly estimated by checking if  $y_0 < y_{-1}$  and  $y_0 < y_1$ . In such instances, the local minimum is bracketed by  $x_{-1}$  and  $x_1$ . Conversely, if the minimum of  $\{y_i | i = -1, 0, \text{ and } 1\}$  is not  $y_0$ , then the assumption of a parabolic landscape is likely invalid. Under such circumstances, it is possible to fit a parabola with  $a < 0$  which would direct  $\theta$  toward the maximum. Thus, more data are required to accurately construct the parabola around the minimum.

In the following discussion, we assume that the minimum is at  $x_1$ . For cases where the minimum is at  $x_{-1}$ , the treatment follows similarly. If the minimum is at  $x_1$ , we proceed to explore  $x_4$  and add  $(x_4, y_4)$  to  $\mathcal{F}$ . The parabola is then fitted using the least-squares method

$$\min \sum_{(x_i, y_i) \in \mathcal{F}} (ax_i^2 + bx_i + c - y_i)^2 \quad (10)$$

We do not expect that using slightly different data points such as  $x_3$  or  $x_5$  will have a significant impact on the overall performance of the method. The arbitrary choice of the additional data point is based on the understanding that these corner cases are rare during the optimization process. Thus, our primary objective for these cases is to prevent significant errors rather than choosing the best possible data point that maximizes the performance. In even rarer corner cases where the newly measured  $y_4$  is smaller than  $y_1$ , we directly use  $(x_4, y_4)$  as the minimum. This avoids the possibility of fitting a parabola with  $a > 0$  while maintaining the simplicity of the algorithm.

To summarize, in typical cases in which  $x_0$  is close to the minimum, two energy evaluations are required to execute the line search. In the worst-case scenario, where  $x_0$  is distant from the minimum, three energy evaluations are required. A keen reader might point out that, since we have assumed  $y_0 = E(\vec{\theta}_{i-1})$  is available from the previous iteration, another energy evaluation at  $\vec{\theta}_i$  is required before completing the current iteration. However, to avoid extra measurements, a shortcut can be taken. When the quality of the fitting is good and the minimum is bracketed by  $x_{-1}$  and  $x_1$ , we assign the minimum

value of the fitted parabola,  $c - \frac{b^2}{4a}$ , to the energy  $E(\vec{\theta}_i)$ . We note that using  $c - \frac{b^2}{4a}$  directly could potentially be more accurate than measuring the energy on quantum computers considering the inherent uncertainty of measurement. On the other hand, if the parabola is fitted with the input from  $(x_{-4}, y_{-4})$  or  $(x_4, y_4)$ , an additional energy evaluation at  $\vec{\theta}_i$  is performed to avoid the bias introduced by the approximate parabola. In practice, we find that this approximation saves approximately one-third of the energy evaluations without compromising the accuracy of the energy estimation. Taking this into account, the number of energy evaluations at each iteration ranges from 2 to 4 depending on the quality of the parabolic fitting. The complete LSAP algorithm is presented in Algorithm 1.

**Algorithm 1:** Line Search with Approximate Parabola (LSAP)

---

```

input : Direction vector  $\vec{v}_i$ , circuit parameter  $\vec{\theta}_{i-1}$ , and the corresponding energy
         $E_{i-1}$ 
output: Circuit parameter at the minimum  $\vec{\theta}_i$  and the corresponding energy  $E_i$ 

1  $y_{-1} = \text{MeasureEnergy}(\vec{\theta} + x_{-1}\vec{v}_i)$ 
2  $y_0 = E_{i-1}$ 
3  $y_1 = \text{MeasureEnergy}(\vec{\theta} + x_1\vec{v}_i)$ 
4  $y_m = \text{MIN}(y_{-1}, y_0, y_1)$ 
5  $\mathcal{F} = \{(x_j, y_j) | j = -1, 0, 1\}$ 
/* treat corner cases
6 if  $y_m$  is not  $y_0$  then
7   if  $y_m$  is  $y_{-1}$  then  $k = -4$ 
8   else  $k = 4$ 
9    $y_k = \text{MeasureEnergy}(\vec{\theta} + x_k\vec{v}_i)$ 
10  if  $y_m < y_k$  then
11    | add  $(x_k, y_k)$  to  $\mathcal{F}$ 
12  else
13    | return  $\vec{\theta} + x_k\vec{v}_i, y_k$ 
14  end
15 end
16 Fit  $y = ax^2 + bx + c$  based on  $\mathcal{F}$ 
17  $\vec{\theta}_i = \vec{\theta}_{i-1} - \frac{b}{2a}\vec{v}_i$ 
18 if  $\mathcal{F}$  has exactly 3 elements then  $E_i = c - \frac{b^2}{4a}$ 
19 else  $E_i = \text{MeasureEnergy}(\vec{\theta}_i)$ 
20 return  $\vec{\theta}_i, E_i$ 
```

---

The sequential optimization method is most effective when the circuit parameters are “uncorrelated”, meaning that adjusting one of the parameters  $\theta_i$  only causes a constant shift in the landscape with respect to other parameters  $\theta_j$ . In numerical analysis, this property is referred to as  $\vec{v}_i$  being a “conjugate” with other  $\vec{v}_j$ . However, one of the main limitations of the sequential optimization procedure is that it neglects the “correlation” between the circuit parameters. More specifically, if the Hessian matrix  $\frac{\partial^2 E}{\partial \theta_i \partial \theta_j}$  has large nondiagonal elements, the optimization may take very small steps along  $\vec{v}_i$  and  $\vec{v}_j$  and results in a large number of energy evaluations. An example based on the function of two variables is depicted in Figure 1.

To deal with this problem, we can draw inspiration from Powell's method. Powell's method is a well-established method for multivariable optimization which features an iterative line-search over each of the variables.<sup>46</sup> Powell's method updates the set of directions  $\mathcal{V}$  to account for the missing “correlation”. After an iteration over  $\mathcal{V}$ , denoting the set of parameters before and after as  $\vec{\theta}_0$  and  $\vec{\theta}_N$  respectively, Powell's method adds  $\vec{\theta}_N - \vec{\theta}_0$  to  $\mathcal{V}$ , which represents the average optimization direction for the last iteration. As shown in Figure 1, this approach works well when the Hessian matrix has large nondiagonal elements. Since  $\mathcal{V}$  is now overcomplete, an element from  $\mathcal{V}$  can be dropped. Powell's method suggests dropping  $\vec{v}_j$ , the direction with the largest energy descent  $\Delta_j = |E_j - E_{j-1}|$ , where  $E_j$  represents the estimated energy after optimization along  $\vec{v}_j$ . This seemingly counterintuitive choice is

made because  $\vec{v}_i$  is likely to be a major component of the new direction  $\vec{\theta}_N - \vec{\theta}_0$ , and dropping  $\vec{v}_i$  avoids the buildup of linear dependence in  $\mathcal{V}$ . In the context of SOAP, the new direction is further normalized to unit length since fixed  $u$  is employed.

Powell's method also suggests several conditions when  $\mathcal{V}$  should not be updated. To assess the quality of the new direction  $\vec{\theta}_N - \vec{\theta}_0$ , an energy evaluation at an extrapolated point along the new direction  $\vec{\theta}_N - \vec{\theta}_0$  is performed to get  $E_{\text{ext}} = E(2\vec{\theta}_N - \vec{\theta}_0)$ . If  $E_{\text{ext}} \geq E_0$ , then moving along the new direction is obviously fruitless. Another condition is to check if the following condition holds

$$2(E_0 - 2E_N + E_{\text{ext}})[(E_0 - E_N) - \Delta] \geq (E_0 - E_{\text{ext}})^2 \Delta \quad (11)$$

Roughly speaking, by rejecting the new direction when eq 11 is satisfied, the set of directions  $\mathcal{V}$  is directed to the limit where all elements are mutually conjugate. For a more comprehensive understanding of this condition, readers are encouraged to refer to the original manuscript by Powell,<sup>46</sup> which provides a clear and concise derivation.

We finalize this session by summarizing the key points of SOAP. The complete algorithm is presented in Algorithm 2. The algorithm begins by initializing the set of directions  $\mathcal{V}$  with unit vectors in the parameter space. It then performs a line search along each direction in  $\mathcal{V}$  to find the minimum energy point. The line search involves fitting a parabola based on 2 to 4 data points and using it to estimate the minimum. The line search procedure is specifically designed for the UCC ansatz and is significantly more efficient and noise-resilient than general line search procedures such as the golden section search. One advantage of SOAP is its ability to handle cases where the circuit parameters are correlated, which is a limitation of traditional sequential optimization methods. After an iteration over  $\mathcal{V}$ , SOAP identifies promising average directions that account for the correlation between parameters and updates  $\mathcal{V}$  with the new direction. The algorithm then proceeds to the next iteration until certain convergence criteria are satisfied.

---

**Algorithm 2:** Sequential Optimization with an Approximate Parabola (SOAP)

---

```

input : MP2 amplitudes
output: Circuit parameter at the minimum  $\vec{\theta}^*$  and the corresponding energy  $E(\vec{\theta}^*)$ 

1 Set  $\vec{\theta}_N$  to MP2 amplitudes
2  $E_N = \text{MeasureEnergy}(\vec{\theta}_N)$ 
3 Initialize  $\mathcal{V}$  by  $\vec{v}_i = \vec{e}_i$ 
4 while not converged do
5    $\vec{\theta}_0 = \vec{\theta}_N$ ,  $E_0 = E_N$ 
6   for each  $\vec{v}_i$  in  $\mathcal{V}$  do
7      $\vec{\theta}_i$ ,  $E_i = \text{LSAP}(\vec{v}_i, \vec{\theta}_{i-1}, E_{i-1})$  /* See Algorithm 1 */
8      $\Delta_i = E_{i-1} - E_i$ 
9     if  $\Delta_i$  is the largest so far in the iteration then
10      |    $j = i$ ,  $\Delta = \Delta_i$ 
11    end
12  end
13  /* Update  $\mathcal{V}$  using the Powell's algorithm */
14   $E_{\text{ext}} = \text{MeasureEnergy}(2\vec{\theta}_N - \vec{\theta}_0)$ 
15  if  $E_{\text{ext}} \geq E_0$  then continue
16  /* Eq. 11 is an inequality of  $E_0$ ,  $E_N$ ,  $E_{\text{ext}}$  and  $\Delta$  */
17  if Eq. 11 is satisfied then continue
18  Remove  $\vec{v}_j$  from  $\mathcal{V}$ 
19  Add  $(\vec{\theta}_N - \vec{\theta}_0)/|\vec{\theta}_N - \vec{\theta}_0|$  to the beginning of  $\mathcal{V}$ 
20 end

```

---

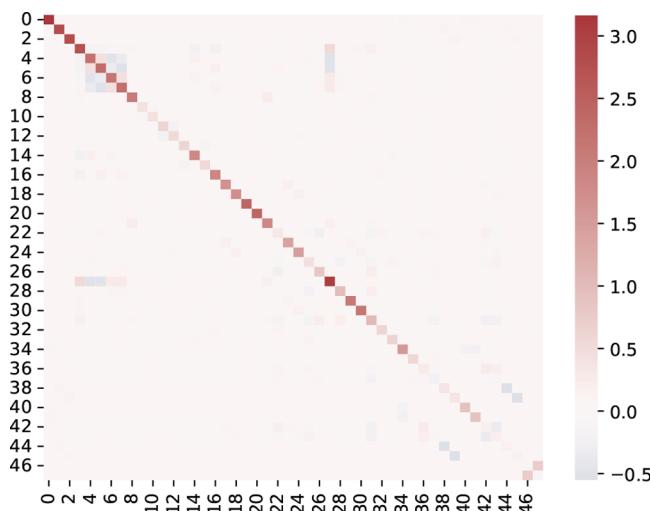
**2.3. Numerical Details.** For benchmark results, we use 5 different molecular systems in this paper. For UCC ansatz, the molecules are N<sub>2</sub> and H<sub>8</sub> chains at even spacing and tetrahedral CH<sub>4</sub>. The bond length  $d$  for H<sub>8</sub> and CH<sub>4</sub> is defined as the distance between H atoms and the distance between C and H atoms, respectively. The STO-3G basis set is used throughout, with the 1s orbital for N<sub>2</sub> and CH<sub>4</sub> frozen. Under the Jordan–

Wigner transformation, all 3 systems correspond to 16 qubits. We focus on the UCC with singles and doubles (UCCSD) ansatz as a representative UCC ansatz. For the HEA ansatz, we employ LiH and BeH<sub>2</sub> with STO-3G basis, which corresponds to 10 and 12 qubits, respectively, with parity transformation. The numerical simulation of quantum circuits and hardware experiments are conducted via TensorCircuit<sup>60</sup> and TenCirChem.<sup>61</sup> TenCirChem is equipped with an efficient algorithm to simulate UCC circuits that allows a 16-qubit simulation in minutes on a laptop. Additionally, TenCirChem evaluates the gradients by autodifferentiation with reversible computing which enables the computation of gradients with  $O(1)$  time and space complexity. The *ab initio* integrals, reference energies, and MP2 amplitudes are calculated via the PySCF package.<sup>62,63</sup>

### 3. RESULTS AND DISCUSSION

**3.1. The Energy Landscape.** The success of SOAP relies on the assumption that the landscape can be approximated by a quadratic function, i.e., a parabola. The assumption is justified by the theoretical analysis in Section 2.1. In this section, we provide numerical evidence to further validate this assumption. The details of the model systems and computational methods are listed in Section 2.3. We examine the landscape of the energy function with respect to a single parameter while keeping the other parameters fixed at the initial guess generated by MP2. We used the N<sub>2</sub> molecule with a bond length of  $d = 2.0$  Å as an example system. The results of the landscape analysis are listed in Figure 2. At the top of each subfigure, the excitation operators associated with each parameter are displayed. The corresponding Hermitian conjugation is omitted for brevity. If an asterisk is shown, it represents two excitation operators sharing the same parameter. The red crossing represents the amplitude of the initial guess and the corresponding energy. From Figure 2, it is evident that, in most cases, the MP2 initial guess is very close to the minimum energy point. As a consequence, a quadratic expansion around the initial guess appears to be a reasonable approximation. Meanwhile, if the starting point deviates from the minimum, a simple update as described in Algorithm 1 should be sufficient to guide the parameter to the minimum. The N<sub>2</sub> molecule at a stretched bond length is considered a prototypical example of strongly correlated systems. Therefore, it is expected that for less correlated systems the parabola expansion assumption holds even more valid. More examples of the landscape, including N<sub>2</sub> but with a zoomed-in  $x$  axis and the landscape of CH<sub>4</sub>, are shown in the Supporting Information.

In order to further investigate the landscape of the energy function and at the proximity to the minimum, we examine the Hessian matrix corresponding to the UCC ansatz shown in Figure 2. The Hessian matrix is estimated numerically using the numdifftools package.<sup>64</sup> In Figure 3, we present a plot of the Hessian matrix. The elements with the largest amplitudes lie in the diagonal of the Hessian matrix, which means that most of the parameters in the ansatz exhibit large second-order derivatives. Meanwhile, the contribution of nondiagonal elements in the Hessian matrix is non-negligible. These nondiagonal elements imply that corresponding excitations could lead to a significant energy decrease if they are excited simultaneously. This indicates the presence of correlations among the parameters and emphasizes the need for optimization methods that account for such correlations.



**Figure 3.** Hessian matrix for the UCC ansatz at the initial parameter generated by MP2. The  $\text{N}_2$  molecule with  $2.0 \text{ \AA}$  bond length is used as the model system. The order of the parameters is the same as that in [Figure 2](#).

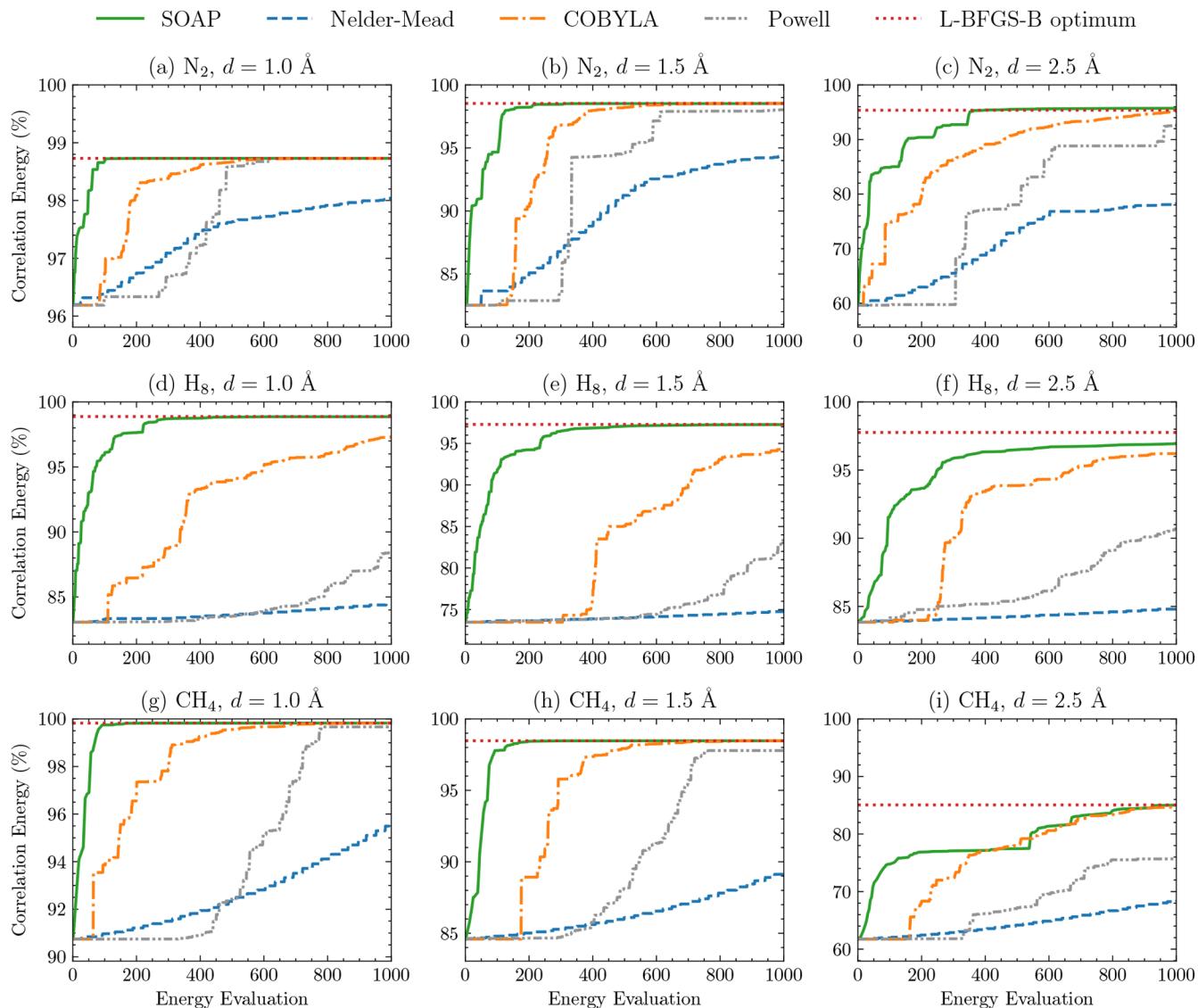
**3.2. Performance of SOAP.** To evaluate the performance of SOAP on the UCCSD ansatz, we conducted experiments on three model systems:  $\text{N}_2$ ,  $\text{H}_8$ , and  $\text{CH}_4$ , as described in [Section 2.3](#). The number of independent parameters for these systems is 48, 108, and 62, respectively. In our comparison, we included three traditional gradient-free optimization algorithms: Nelder–Mead algorithm,<sup>65</sup> COBYLA, and Powell's method. We intentionally excluded gradient-based methods from our comparison due to their higher quantum resource demand. These methods typically require more energy evaluations than gradient-free methods because calculating energy gradients via the parameter-shift rule is very expensive. More specifically, for the three model systems examined in this work, the number of energy evaluations required to compute the gradients using the parameter-shift rule is 936, 2816, and 1344, respectively. The number is much higher than the number of independent parameters because the parameter-shift rule requires each variable to be associated with a single-qubit rotation gate. As presented in [Figure 4](#) and [Table 2](#), SOAP can reach convergence using fewer energy evaluations than what is needed to compute the gradients even once. Additionally, the SPSA algorithm is not included either, as it reduces to a finite-difference-based stochastic gradient descent method when the initial guess is close to the minimum. In our test calculations, we observed that SPSA performs poorly for the UCC ansatz, particularly when measurement noise is present. This finding contradicts a recent paper where SPSA is reported to outperform COBYLA.<sup>66</sup> We conjecture that the reason for this discrepancy is that our benchmark system is significantly larger and we did not conduct extensive hyperparameter tuning.

[Figure 4](#) shows the optimized energy versus the number of energy evaluations by the optimizer by using a noiseless simulator. The energy values are rescaled by the correlation energy  $E_{\text{corr}} = E_{\text{HF}} - E_{\text{FCI}}$ , whose exact values are listed in [Table 1](#). The optimal energy obtained by the L-BFGS-B method is used as the best possible energy for the UCC ansatz. It is worth noting that, although the L-BFGS-B method performs well for numerical simulations where the energy gradients are available in  $O(1)$  time and space complexity, it

requires  $O(N)$  energy evaluations when executed on real quantum devices. Besides, L-BFGS-B also degrades when measurement noise is present. The results in [Figure 4](#) demonstrate that SOAP exhibits convergence significantly faster than that of traditional optimization methods, particularly at intermediate bond lengths. As expected, at the dissociation limit where  $d = 2.5 \text{ \AA}$ , the performance gap between SOAP and other methods becomes smaller because the initial guess runs away from the minimum due to strong correlation effects. Nonetheless, even in these cases, SOAP still demonstrates the fastest convergence except for  $\text{CH}_4$  at  $d = 2.5 \text{ \AA}$ . In the [Supporting Information](#), we include benchmark results where the initial parameter is set to the HF state or random state, and from the results, the relative performance between SOAP and traditional optimizers is not sensitive to the initial parameters. Among the traditional optimization methods, the COBYLA method appears to be the most efficient, which is consistent with previous studies.<sup>50</sup> We also note that MP2 initialization appears to significantly accelerate convergence. For the equilibrium bond distance, the MP2 initialization allows the optimizers to start the optimization from approximately 90% correlation energy, compared with 0% correlation energy with HF initialization. More data with  $d = 0.5 \text{ \AA}$  and  $d = 2.0 \text{ \AA}$  is provided in the [Supporting Information](#).

To provide a quantitative assessment of the relative performance of the optimization methods, we compare the number of energy evaluations required for convergence in [Table 2](#). Here, the convergence is defined as reaching 99% of the correlation energy obtained by the L-BFGS-B method. For example, let us consider a scenario where the correlation energy is 0.150 Hartree and the optimal correlation energy obtained by the L-BFGS-B method is 0.100 Hartree. In this case, convergence is considered to be reached if the optimizer reaches a correlation energy of  $0.100 \times 99\% = 0.099$  Hartree. While this criterion cannot be practically used to determine convergence since the energy by the L-BFGS-B method is not known in advance, it serves as a fair measure of how quickly the optimizer finds the minimum defined by the UCC ansatz. The results presented in [Table 2](#) demonstrate that the SOAP optimizer consistently exhibits the fastest convergence among all the methods studied. In many cases, particularly near equilibrium, SOAP is more than five times faster than the second-best method. Notably, at a bond length of  $d = 1.0 \text{ \AA}$ , SOAP requires approximately  $N$  energy evaluations for an optimization problem with  $N$  parameters. This observation leads us to the conclusion that it is unlikely that any new optimizer will significantly outperform SOAP under such circumstances.

The most important part of this section is the performance of SOAP in the presence of noise from quantum devices. The ideal optimizer for parametrized quantum circuits should be noise-resilient; otherwise, the optimizer is only suitable for noiseless numerical simulation. For the results presented below, we consider the statistical measurement uncertainty modeled using Gaussian noise. In traditional numerical simulations of quantum circuits, the measurement uncertainty is modeled by shot-based simulation, which most faithfully emulates the behavior of quantum computers. However, such a method has a large simulation overhead that limits the simulation to small systems. To address this issue, we model the measurement uncertainty as Gaussian noise that is applied to the noiseless energy. Such a method has negligible computational overhead, and the scale of noisy simulation is



**Figure 4.** Convergence rate of SOAP compared to other gradient-free optimization methods based on noiseless simulation. The red dotted line is the optimum value from the L-BFGS-B method, which is a gradient-based method that is costly on a quantum device. The (a–c), (d–f), and (g–i) panels are for the  $\text{N}_2$ ,  $\text{H}_8$ , and  $\text{CH}_4$  molecules, respectively. From left to right the bond length increases from  $d = 1.0 \text{ \AA}$  to  $d = 2.5 \text{ \AA}$ .

**Table 1.**  $E_{\text{corr}}$  for  $\text{N}_2$ ,  $\text{H}_8$ , and  $\text{CH}_4$  from  $d = 0.5$  to  $2.5 \text{ \AA}^a$

molecule	bond length				
	0.5 $\text{\AA}$	1.0 $\text{\AA}$	1.5 $\text{\AA}$	2.0 $\text{\AA}$	2.5 $\text{\AA}$
$\text{N}_2$	0.0374	0.1294	0.3090	0.5836	0.8234
$\text{H}_8$	0.0529	0.1332	0.3234	0.6353	0.9208
$\text{CH}_4$	0.0277	0.0660	0.1698	0.3678	0.6238

<sup>a</sup>The unit is Hartree.

thus the same as that of noiseless simulation. Gate noise is not taken into consideration for two main reasons. First, under a moderate noise ratio, gate noise primarily causes a shift in the entire energy landscape, resulting in incorrect absolute energy estimation. However, it did not significantly alter the optimization behavior. For a high noise ratio, it is unlikely that VQE will produce meaningful results, and there is no need to be concerned with optimization problems. Second, accurately modeling gate noise requires density matrix simulation or extensive Monte Carlo simulation, which are

considerably more computationally expensive compared to the noiseless case using the statevector simulator.

In Figure 5, we show the optimizer trajectories when Gaussian noise with a standard deviation of 0.001 Hartree is added to the energy. For each method, we run 5 independent optimization trajectories, and the standard deviation is depicted as the shaded area. The results demonstrate that the performance of traditional optimization methods degrades to varying extents when measurement noise is introduced. When  $d = 1.0 \text{ \AA}$ , all of the traditional optimization methods fail catastrophically. This is because in these cases  $E_{\text{corr}}$  is typically less than or around 0.1 Hartree (see Table 1) and the initial guess has already recovered  $\approx 90\%$  of the correlation energy. Therefore, the presence of 0.001 Hartree noise can easily misdirect the traditional methods, leading to incorrect optimization directions or false convergence. In contrast, since SOAP is specifically designed to handle measurement uncertainty, it suffers from the least degradation when noise is introduced in the optimization process. SOAP also exhibits

**Table 2.** Number of Energy Evaluations Required for Convergence for SOAP as well as Traditional Gradient-Free Optimizers<sup>a</sup>

molecule	#params	method	bond length				
			0.5 Å	1.0 Å	1.5 Å	2.0 Å	2.5 Å
N <sub>2</sub>	48	SOAP	9	37	116	354	348
		Nelder-Mead	375	630	—	—	—
		COBYLA	236	178	374	606	864
		Powell	510	441	612	—	1132
H <sub>8</sub>	108	SOAP	123	222	286	404	744
		Nelder-Mead	—	—	—	—	—
		COBYLA	966	1144	1312	1770	1502
		Powell	1205	1229	—	—	—
CH <sub>4</sub>	62	SOAP	39	67	89	144	799
		Nelder-Mead	1175	—	—	—	—
		COBYLA	304	312	432	562	853
		Powell	684	733	751	753	—

<sup>a</sup>For a fair comparison, the convergence criterion for all methods is set to reach 99% of the maximum correlation energy by the UCC ansatz, which is defined by the optimized energy obtained by the L-BFGS-B method. The dashed lines indicate that more than 2000 evaluations are required.

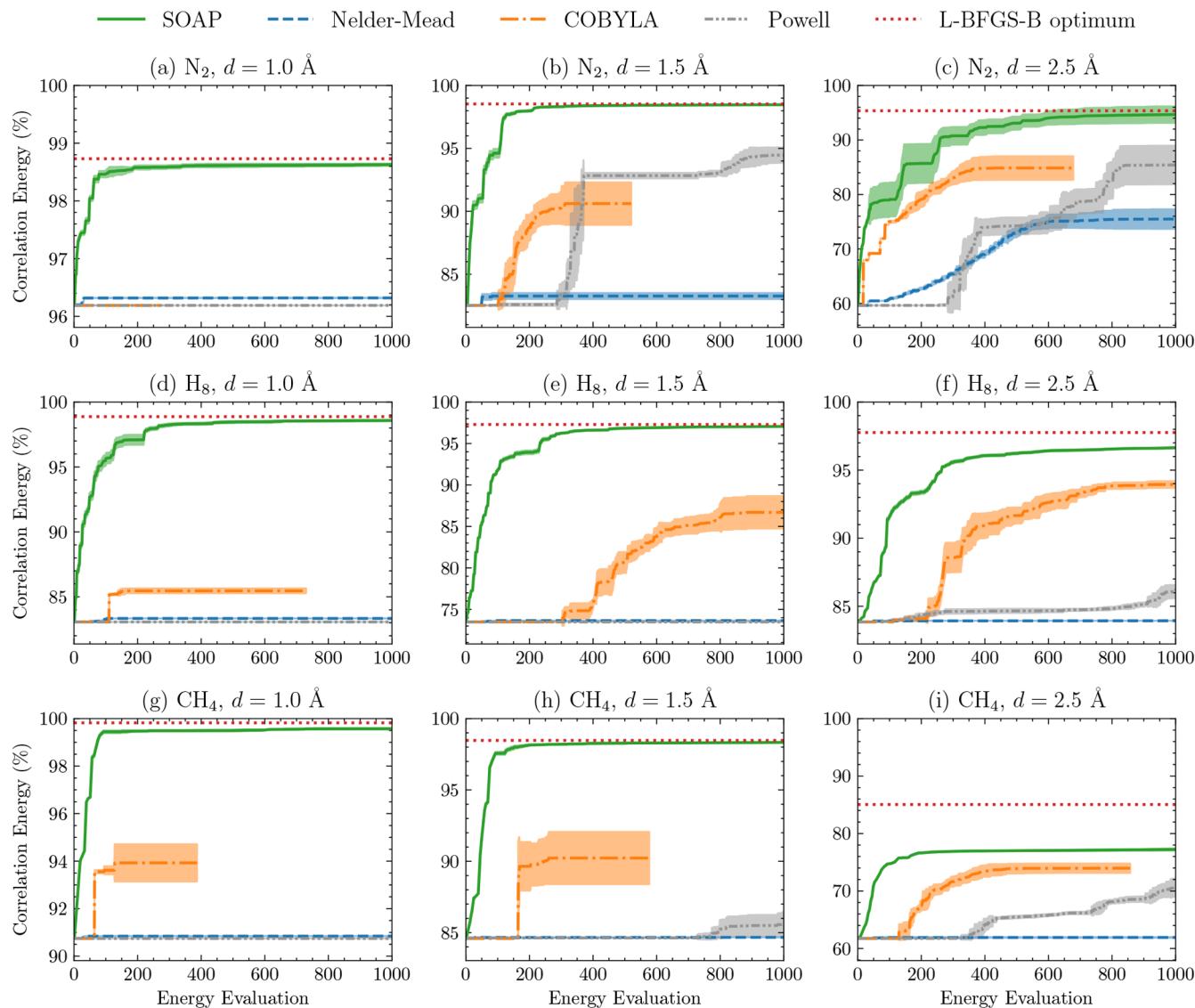
negligible variation over different optimization trajectories. This resilience to noise can be attributed to the fact that, even when the parameter is close to the minimum, SOAP determines the minimum at each optimization step using a finite step size of  $u = 0.1$ . This approach significantly contributes to SOAP's ability to handle noise and maintain robust optimization performance. Similar to Figure 4, we include more data with  $d = 0.5 \text{ \AA}$  and  $d = 2.0 \text{ \AA}$  in the Supporting Information.

To investigate the scaling behavior of SOAP, we examined how the number of energy evaluations scales with the number of parameters. We select the homogeneous hydrogen chain as the model system and gradually increase the number of atoms from 2 to 10, resulting in the largest system composed of 20 qubits. In cases where the number of atoms is odd, a positive charge is added to maintain a closed-shell system. The number of energy evaluations is defined as the number of energy evaluations to achieve convergence, the same as the values presented in Table 2. The results are depicted in Figure 6, where five different bond distances from  $d = 0.5 \text{ \AA}$  to  $d = 2.5 \text{ \AA}$  are considered for each system size. The green dashed line represents the average number of steps across the five different bond distances. Even in scenarios involving strong correlation, specifically at a bond length of  $2.5 \text{ \AA}$ , empirically, the number of energy evaluations for convergence scales linearly with the number of parameters. The fitted black line is  $y = 2.93x$ , which means that on average approximately  $3N$  energy evaluations are required for convergence if there are  $N$  parameters. This finding is also consistent with the data in Table 2. This favorable scaling can be considered to be the best-case scenario for a gradient-free optimizer. Since the number of parameters  $N$  scales as  $M^4$  for the UCCSD ansatz where  $M$  is the system size, the total scaling, including optimization, is  $M^3 \times M^2 \times M^4 = M^9$ , assuming an optimistic scaling of circuit depth as  $M^{3.9}$  and the number of measurements for energy evaluation as  $M^2$ .<sup>18</sup> The  $M^3$  scaling for circuit depth is obtained by assuming there are  $M^4$  gates in the circuit and  $M$  gates can be executed in parallel. The  $M^2$  measurement scaling is achieved by appending a depth  $M$  basis rotation circuit for measurement, allowing  $M^2$  terms to be measured simultaneously. To reduce the overall scaling, the key lies in minimizing the number of excitation operators included in the ansatz. This reduction helps decrease both the circuit execution time and the number of energy evaluations required for optimization. Despite recent advan-

ces,<sup>11,58,67–69</sup> achieving this reduction while maintaining high accuracy poses a significant challenge.

Next, we show that SOAP is a general method that is not limited to the UCCSD ansatz. We employ HEA with Clifford-based Hamiltonian Engineering for Molecules (CHEM) that ensures the initial state is the HF state and the initial gradients are maximized.<sup>15</sup> The CHEM algorithm finds a Clifford transformation  $U_{\text{CHEM}}$  that transforms the Hamiltonian based on the HEA circuit topology. For the results presented below, we employed a 3-layered HEA with linear staircase CNOT gates as entanglers. Each layer contains  $R_x$ ,  $R_y$ , and  $R_z$  parametrized rotation gates for each qubit, and the total number of parameters is  $N = 3lN_{\text{qubit}}$  where  $l$  is the number of layers. The benchmark molecules are LiH with bond distance  $d = 1.6 \text{ \AA}$  and BeH<sub>2</sub> with bond distance  $d = 1.0 \text{ \AA}$ . After applying the parity transformation, the number of qubits for LiH is 10, while for BeH<sub>2</sub> it is 12. The corresponding number of parameters for these molecules is 90 and 108, respectively. The correlation energies obtained for LiH and BeH<sub>2</sub> are 0.0205 and 0.0261 Hartree, respectively. The other setups are the same as those in Figures 4 and 5. The results in Figure 7 demonstrate that SOAP exhibits a faster convergence speed than other gradient-free methods, particularly when considering the measurement noise. Nevertheless, for BeH<sub>2</sub>, SOAP appears to trap at a local minimum, and we are actively working to investigate possible solutions to this issue.

To demonstrate the superiority of SOAP as a parameter optimizer for real quantum devices, we conducted experiments using a superconducting quantum computer. The superconducting quantum processor is composed of nine transmon qubits, and its details are reported in our previous work.<sup>70</sup> In order to reduce circuit depth, we use a model system of H<sub>8</sub> and employ a (2e, 2o) active space approximation. The bond length is set to  $d = 3.0 \text{ \AA}$ , which corresponds to the strong correlation regime. After parity transformation,<sup>71,72</sup> 2 qubits are required to represent the system wave function. Each term in the Hamiltonian is evaluated using  $2^{16}$  measurement shots to determine the expectation value. Quantum circuits are susceptible to readout measurement errors, where it is possible to incorrectly read the state  $|0\rangle$  as  $|1\rangle$  and vice versa. To mitigate these errors, standard readout error mitigation techniques were employed in our study. These techniques aim to correct for the discrepancies between the ideal distribution of bitstrings  $\vec{p}_{\text{ideal}}$  and the noisy distribution of



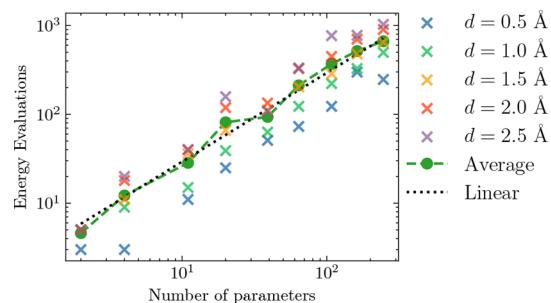
**Figure 5.** Convergence rate of SOAP compared to other gradient-free optimization methods based on noisy simulation. The quantum noise is modeled as Gaussian noise to the measured energy with a standard deviation 0.001 mH. The standard deviation from 5 independent trajectories is depicted as the shaded area. The red dotted line is the optimum value from the L-BFGS-B method based on the noiseless simulation. The (a–c), (d–f), and (g–i) panels are for the  $\text{N}_2$ ,  $\text{H}_8$ , and  $\text{CH}_4$  molecules, respectively. From left to right, the bond length increases from  $d = 1.0 \text{ \AA}$  to  $d = 2.5 \text{ \AA}$ .

bitstrings  $\vec{p}_{\text{noisy}}$ . The relationship between the ideal and noisy distributions is described by the response matrix  $\Lambda$  such that  $\vec{p}_{\text{noisy}} = \Lambda \vec{p}_{\text{ideal}}$ . The response matrix  $\Lambda$  can be decomposed into a direct product of response matrices for each qubit, assuming that the measurements for different qubits are independent. To construct  $\Lambda$ , calibration quantum circuits were used, where each  $|0\rangle$  qubit is subjected to an  $\text{H}$  gate before measurement. The UCCSD ansatz used in the optimization contains three excitation operators with one double excitation and two single excitations. Due to spin symmetry, the two single excitations share the same parameter, resulting in two free parameters in the ansatz. The overall circuit for the ansatz, before being compiled to native gates on the quantum device, is depicted in Figure 8. The  $\text{X}$  gate creates the initial HF state  $|0101\rangle$ , which under parity transformation corresponds to  $|0110\rangle$  and is reduced to  $|01\rangle$  by removing the second and fourth qubit. The first two single qubit  $R_y$  rotation gates are applied for single

excitation, while the following CNOT gates and controlled  $R_y$  gate are applied for double excitation. In Figure 9, we present the convergence behavior of SOAP compared with classical gradient-free optimization methods. To reduce the randomness introduced by measurement uncertainty, the optimization process was independently performed three times for each method. The results clearly demonstrate that SOAP exhibits faster convergence and produces consistent trajectories under realistic hardware conditions.

#### 4. CONCLUSION AND OUTLOOK

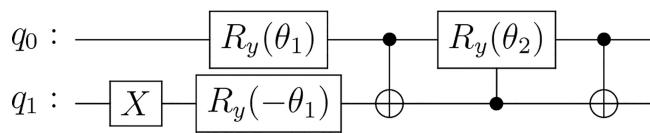
In conclusion, this work introduces SOAP as an efficient and robust parameter optimization method specifically designed for quantum computational chemistry. SOAP demonstrates superior efficiency by leveraging an approximate parabolic expansion of the energy landscape when the initial guess is close to the minimum. Unlike previous sequential optimization



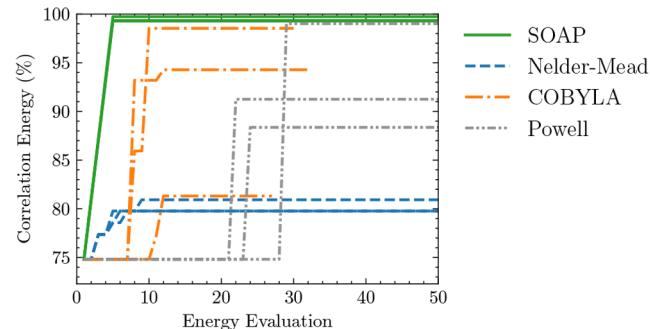
**Figure 6.** Number of energy evaluations required for convergence for SOAP as a function of the number of parameters in the ansatz. The different number of parameters is from hydrogen chains from 2 to 10 atoms, corresponding to a maximum of 20 qubits. Consistent with Table 2, the convergence criteria are set to reach 99% of the maximum correlation energy, which is defined by the optimized energy from the L-BFGS-B method. The green dashed line represents the average number of steps across the five different bond distances, and the black dotted line represents a linear fit of the average number of steps with an expression of  $y = 2.93x$ .

methods, SOAP minimizes the number of energy evaluations by incorporating parameter correlations through the update of the direction set  $\mathcal{V}$ . The algorithm is straightforward to implement and does not require hyperparameter tuning, making it user-friendly and well-suited for quantum chemistry applications.

The performance of SOAP is extensively evaluated through numerical simulations and hardware experiments. Numerical simulations involve model systems, such as  $\text{N}_2$ ,  $\text{H}_8$ , and  $\text{CH}_4$ ,

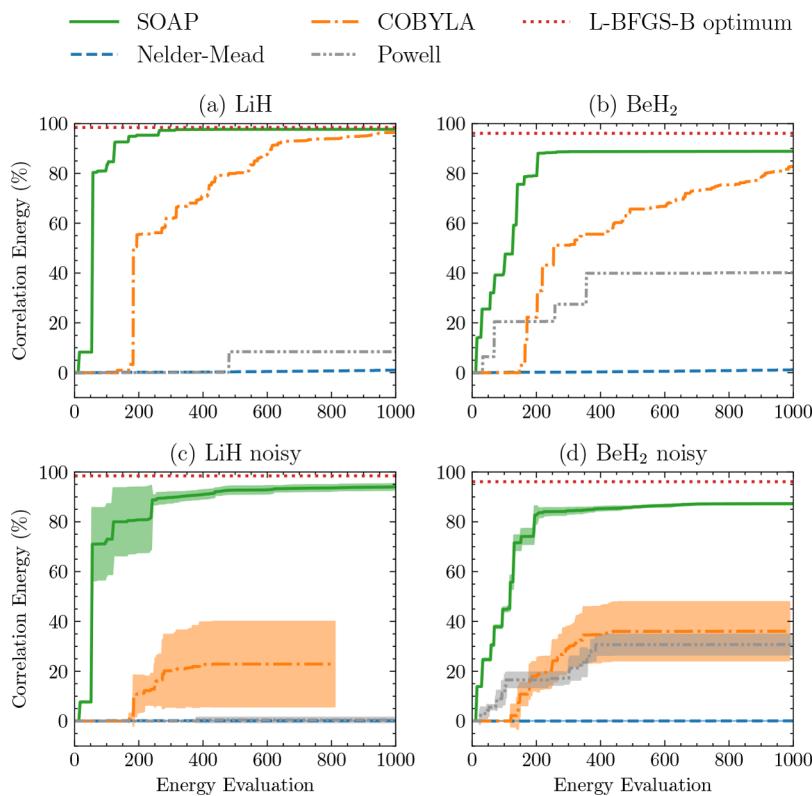


**Figure 8.** Quantum circuit used for hardware experiments.



**Figure 9.** Convergence rate of SOAP compared to other gradient-free optimization methods based on experiments on a superconducting quantum computer. The benchmark platform is UCCSD calculation of  $\text{H}_8$  with a (2e, 2o) active space approximation, which corresponds to 2 qubits and 2 circuit parameters. For each method, 3 independent optimization trials are performed.

which correspond to 16 qubits. The results demonstrate that SOAP achieves significantly faster convergence rates, as measured by the number of energy evaluations required, compared to traditional optimization methods like COBYLA



**Figure 7.** Convergence rate of SOAP compared to other gradient-free optimization methods based on a hardware-efficient ansatz. The results in (a) and (b) are from noiseless simulation, and the results in (c) and (d) take measurement uncertainty into account. The quantum noise is modeled as a Gaussian noise to the measured energy with a standard deviation of 0.001 mH. The standard deviation from 5 independent trajectories is depicted as the shaded area in (c) and (d). The red dotted line is the optimum value from the L-BFGS-B method based on noiseless simulation.

and Powell's method, particularly for bond lengths near equilibrium. Surprisingly, at dissociation bond length where strong correlation is present, the performance of SOAP does not degrade significantly. Furthermore, SOAP exhibits remarkable resilience to measurement noise. Scaling studies up to 20 qubits reveal that, on average, SOAP requires  $3N$  energy evaluations for convergence where  $N$  is the number of parameters in the ansatz, indicating its applicability to larger systems. We also showcase that SOAP is a general method applicable to HEA. The hardware experiments conducted on a (2e, 2o) active space approximation of  $\text{H}_8$  validate SOAP's efficiency and robustness under realistic conditions.

While this paper focuses on the UCCSD ansatz, SOAP is also well-suited for the ADAPT-VQE method,<sup>11</sup> which has a lower circuit depth and good initial guesses during iteration. Additionally, SOAP is a general method that can be applied to other families of ansatz, provided a suitable initial guess is generated.<sup>73,74</sup> In such cases, the SOAP is expected to perform well. In addition, the quality of the initial circuit parameters also affects the number of energy evaluations required for convergence. We expect SOAP can achieve faster convergence with a more sophisticated initial guess.<sup>58</sup>

## ■ ASSOCIATED CONTENT

### Data Availability Statement

The SOAP algorithm is integrated into the open-source TENCIRCHEM package (<https://github.com/tencent-quantum-lab/TenCirChem>).

### ■ Supporting Information

The Supporting Information is available free of charge at <https://pubs.acs.org/doi/10.1021/acs.jctc.4c00155>.

Detailed data for the energy landscape, the benchmark for the choice of  $u$ , the full data for the performance of SOAP in logarithmic scale, and test calculations using different parameter initialization strategies ([PDF](#))

## ■ AUTHOR INFORMATION

### Corresponding Author

Weitang Li – Tencent Quantum Lab, Tencent, Shenzhen 518057, China;  [orcid.org/0000-0002-8739-641X](https://orcid.org/0000-0002-8739-641X); Email: [liwt31@gmail.com](mailto:liwt31@gmail.com)

### Authors

Yufei Ge – Department of Chemistry, Tsinghua University, Beijing 100084, China;  [orcid.org/0000-0001-7150-8483](https://orcid.org/0000-0001-7150-8483)  
Shi-Xin Zhang – Tencent Quantum Lab, Tencent, Shenzhen 518057, China  
Yu-Qin Chen – Tencent Quantum Lab, Tencent, Shenzhen 518057, China  
Shengyu Zhang – Tencent Quantum Lab, Tencent, Hong Kong 999077, China

Complete contact information is available at: <https://pubs.acs.org/10.1021/acs.jctc.4c00155>

### Notes

The authors declare no competing financial interest.

## ■ ACKNOWLEDGMENTS

We thank Jonathan Allcock for insightful discussions. W.L. is supported by the Young Elite Scientists Sponsorship Program by CAST, 2023QNRC001.

## ■ REFERENCES

- Bharti, K.; Cervera-Lierta, A.; Kyaw, T. H.; Haug, T.; Alperin-Lea, S.; Anand, A.; Degroote, M.; Heimonen, H.; Kottmann, J. S.; Menke, T.; Mok, W.-K.; Sim, S.; Kwek, L.-C.; Aspuru-Guzik, A. Noisy intermediate-scale quantum algorithms. *Rev. Mod. Phys.* **2022**, *94*, 015004.
- Hoeffler, T.; Häner, T.; Troyer, M. Disentangling hype from practicality: on realistically achieving quantum advantage. *Commun. ACM* **2023**, *66*, 82–87.
- McArdle, S.; Endo, S.; Aspuru-Guzik, A.; Benjamin, S. C.; Yuan, X. Quantum computational chemistry. *Rev. Mod. Phys.* **2020**, *92*, 015003.
- Ma, H.; Liu, J.; Shang, H.; Fan, Y.; Li, Z.; Yang, J. Multiscale quantum algorithms for quantum chemistry. *Chem. Sci.* **2023**, *14*, 3190–3205.
- Peruzzo, A.; McClean, J.; Shadbolt, P.; Yung, M.-H.; Zhou, X.-Q.; Love, P. J.; Aspuru-Guzik, A.; O'Brien, J. L. A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **2014**, *5*, 1–7.
- Cerezo, M.; Arrasmith, A.; Babbush, R.; Benjamin, S. C.; Endo, S.; Fujii, K.; McClean, J. R.; Mitarai, K.; Yuan, X.; Cincio, L.; Coles, P. J. Variational quantum algorithms. *Nat. Rev. Phys.* **2021**, *3*, 625–644.
- Tilly, J.; Chen, H.; Cao, S.; Picozzi, D.; Setia, K.; Li, Y.; Grant, E.; Wossnig, L.; Rungger, I.; Booth, G. H.; Tennyson, J. The variational quantum eigensolver: a review of methods and best practices. *Phys. Rep.* **2022**, *986*, 1–128.
- Anand, A.; Schleich, P.; Alperin-Lea, S.; Jensen, P. W.; Sim, S.; Diaz-Tinoco, M.; Kottmann, J. S.; Degroote, M.; Izmaylov, A. F.; Aspuru-Guzik, A. A quantum computing view on unitary coupled cluster theory. *Chem. Soc. Rev.* **2022**, *51*, 1659–1684.
- Lee, J.; Huggins, W. J.; Head-Gordon, M.; Whaley, K. B. Generalized unitary coupled cluster wave functions for quantum computation. *J. Chem. Theory Comput.* **2019**, *15*, 311–324.
- Evangelista, F. A.; Chan, G. K.-L.; Scuseria, G. E. Exact parameterization of fermionic wave functions via unitary coupled cluster theory. *J. Chem. Phys.* **2019**, *151*, 244112.
- Grimsley, H. R.; Economou, S. E.; Barnes, E.; Mayhall, N. J. An adaptive variational algorithm for exact molecular simulations on a quantum computer. *Nat. Commun.* **2019**, *10*, 1–9.
- Elfving, V. E.; Millaruelo, M.; Gámez, J. A.; Gogolin, C. Simulating quantum chemistry in the seniority-zero space on qubit-based quantum computers. *Phys. Rev. A* **2021**, *103*, 032605.
- Kandala, A.; Mezzacapo, A.; Temme, K.; Takita, M.; Brink, M.; Chow, J. M.; Gambetta, J. M. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* **2017**, *549*, 242–246.
- McClean, J. R.; Boixo, S.; Smelyanskiy, V. N.; Babbush, R.; Neven, H. Barren plateaus in quantum neural network training landscapes. *Nat. Commun.* **2018**, *9*, 4812.
- Sun, J.; Cheng, L.; Li, W. Toward Chemical Accuracy with Shallow Quantum Circuits: A Clifford-Based Hamiltonian Engineering Approach. *J. Chem. Theory Comput.* **2024**, *20*, 695–707.
- Xiao, X.; Zhao, H.; Ren, J.; Fang, W.-H.; Li, Z. Physics-constrained hardware-efficient ansatz on quantum computers that is universal, systematically improvable, and size-consistent. *J. Chem. Theory Comput.* **2024**, *20*, 1912.
- Verteletskyi, V.; Yen, T.-C.; Izmaylov, A. F. Measurement optimization in the variational quantum eigensolver using a minimum clique cover. *J. Chem. Phys.* **2020**, *152*, 124114.
- Huggins, W. J.; McClean, J. R.; Rubin, N. C.; Jiang, Z.; Wiebe, N.; Whaley, K. B.; Babbush, R. Efficient and noise resilient measurements for quantum chemistry on near-term quantum computers. *npj Quantum Inf.* **2021**, *7*, 23.
- Wu, B.; Sun, J.; Huang, Q.; Yuan, X. Overlapped grouping measurement: A unified framework for measuring quantum states. *Quantum* **2023**, *7*, 896.
- Rivera-Dean, J.; Huembeli, P.; Acín, A.; Bowles, J. Avoiding local minima in Variational Quantum Algorithms with Neural Networks. *arXiv* **2021**, arXiv:2104.02955.

- (21) Slattery, L.; Villalonga, B.; Clark, B. K. Unitary block optimization for variational quantum algorithms. *Phys. Rev. Research* **2022**, *4*, 023072.
- (22) Tamiya, S.; Yamasaki, H. Stochastic gradient line Bayesian optimization for efficient noise-robust optimization of parameterized quantum circuits. *npj Quantum Inf.* **2022**, *8*, 90.
- (23) Cheng, L.; Chen, Y.-Q.; Zhang, S.-X.; Zhang, S. Error-mitigated quantum approximate optimization via learning-based adaptive optimization. *arXiv* **2023**, arXiv:2303.14877.
- (24) Choy, B.; Wales, D. J. Molecular energy landscapes of hardware-efficient ansatz in quantum computing. *J. Chem. Theory Comput.* **2023**, *19*, 1197–1206.
- (25) Liu, S.; Zhang, S.-X.; Jian, S.-K.; Yao, H. Training variational quantum algorithms with random gate activation. *Phys. Rev. Res.* **2023**, *5*, L032040.
- (26) Mitarai, K.; Negoro, M.; Kitagawa, M.; Fujii, K. Quantum circuit learning. *Phys. Rev. A* **2018**, *98*, 032309.
- (27) Schuld, M.; Bergholm, V.; Gogolin, C.; Izaac, J.; Killoran, N. Evaluating analytic gradients on quantum hardware. *Phys. Rev. A* **2019**, *99*, 032331.
- (28) Rumelhart, D. E.; Hinton, G. E.; Williams, R. J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536.
- (29) Bowles, J.; Wierichs, D.; Park, C.-Y. Backpropagation scaling in parameterised quantum circuits. *arXiv* **2023**, arXiv:2306.14962; DOI: 10.48550/arXiv.2306.14962.
- (30) Kawashima, Y.; Lloyd, E.; Coons, M. P.; Nam, Y.; Matsuura, S.; Garza, A. J.; Johri, S.; Huntington, L.; Senicourt, V.; Maksymov, A. O.; et al. Optimizing electronic structure simulations on a trapped-ion quantum computer using problem decomposition. *Commun. Phys.* **2021**, *4*, 245.
- (31) O'Brien, T. E.; Anselmetti, G.; Gkritis, F.; Elfving, V. E.; Polla, S.; Huggins, W. J.; Oumarou, O.; Kechedzhi, K.; Abanin, D.; Acharya, R.; Aleiner, I.; Allen, R.; Andersen, T. I.; Anderson, K.; Ansmann, M.; Arute, F.; Arya, K.; Asfaw, A.; Atalaya, J.; Bardin, J. C.; Bengtsson, A.; Bortoli, G.; Bourassa, A.; Bovaerd, J.; Brill, L.; Broughton, M.; Buckley, B.; Buell, D. A.; Burger, T.; Burkett, B.; Bushnell, N.; Campero, J.; Chen, Z.; Chiaro, B.; Chik, D.; Cogan, J.; Collins, R.; Conner, P.; Courtney, W.; Crook, A. L.; Curtin, B.; Debroy, D. M.; Demura, S.; Drozdov, I.; Dunsworth, A.; Erickson, C.; Faoro, L.; Farhi, E.; Fatemi, R.; Ferreira, V. S.; Flores Burgos, L.; Forati, E.; Fowler, A. G.; Foxen, B.; Giang, W.; Gidney, C.; Gilboa, D.; Giustina, M.; Gosula, R.; Grajales Dau, A.; Gross, J. A.; Habegger, S.; Hamilton, M. C.; Hansen, M.; Harrigan, M. P.; Harrington, S. D.; Heu, P.; Hoffmann, M. R.; Hong, S.; Huang, T.; Huff, A.; Ioffe, L. B.; Isakov, S. V.; Iveland, J.; Jeffrey, E.; Jiang, Z.; Jones, C.; Juhas, P.; Kafri, D.; Khattar, T.; Khezri, M.; Kieferová, M.; Kim, S.; Klimov, P. V.; Klots, A. R.; Korotkov, A. N.; Kostritsa, F.; Kreikebaum, J. M.; Landhuis, D.; Laptev, P.; Lau, K.-M.; Laws, L.; Lee, J.; Lee, K.; Lester, B. J.; Lill, A. T.; Liu, W.; Livingston, W. P.; Locharla, A.; Malone, F. D.; Mandrà, S.; Martin, O.; Martin, S.; McClean, J. R.; McCourt, T.; McEwen, M.; Mi, X.; Mieszala, A.; Miao, K. C.; Mohseni, M.; Montazeri, S.; Morvan, A.; Movassagh, R.; Mruczkiewicz, W.; Naaman, O.; Neeley, M.; Neill, C.; Nersisyan, A.; Newman, M.; Ng, J. H.; Nguyen, A.; Nguyen, M.; Niu, M. Y.; Omonije, S.; Opremcak, A.; Petukhov, A.; Potter, R.; Pryadko, L. P.; Quintana, C.; Rocque, C.; Roushan, P.; Saei, N.; Sank, D.; Sankaragomathi, K.; Satzinger, K. J.; Schurkus, H. F.; Schuster, C.; Shearn, M. J.; Shorter, A.; Shutty, N.; Shvarts, V.; Skruzny, J.; Smith, W. C.; Somma, R. D.; Sterling, G.; Strain, D.; Szalay, M.; Thor, D.; Torres, A.; Vidal, G.; Villalonga, B.; Vollgraff Heidweiller, C.; White, T.; Woo, B. W. K.; Xing, C.; Yao, Z. J.; Yeh, P.; Yoo, J.; Young, G.; Zalcman, A.; Zhang, Y.; Zhu, N.; Zobrist, N.; Bacon, D.; Boixo, S.; Chen, Y.; Hilton, J.; Kelly, J.; Lucero, E.; Megrant, A.; Neven, H.; Smelyanskiy, V.; Gogolin, C.; Babbush, R.; Rubin, N. C. Purification-based quantum error mitigation of pair-correlated electron simulations. *Nat. Phys.* **2023**, *19*, 1787–1792.
- (32) Zhao, L.; Goings, J.; Shin, K.; Kyoung, W.; Fuks, J. I.; Kevin Rhee, J.-K.; Rhee, Y. M.; Wright, K.; Nguyen, J.; et al. Orbital-optimized pair-correlated electron simulations on trapped-ion quantum computers. *npj Quantum Inf.* **2023**, *9*, 60.
- (33) Virtanen, P.; Gommers, R.; Oliphant, T. E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; van der Walt, S. J.; Brett, M.; Wilson, J.; Millman, K. J.; Mayorov, N.; Nelson, A. R. J.; Jones, E.; Kern, R.; Larson, E.; Carey, C. J.; Polat, İ.; Feng, Y.; Moore, E. W.; VanderPlas, J.; Laxalde, D.; Perktold, J.; Cimrman, R.; Henriksen, I.; Quintero, E. A.; Harris, C. R.; Archibald, A. M.; Ribeiro, A. H.; Pedregosa, F.; van Mulbregt, P. SciPy 1.0 Contributors SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261–272.
- (34) Byrd, R. H.; Lu, P.; Nocedal, J.; Zhu, C. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comput.* **1995**, *16*, 1190–1208.
- (35) Lang, R. A.; Ryabinkin, I. G.; Izmaylov, A. F. Unitary transformation of the electronic Hamiltonian with an exact quadratic truncation of the Baker-Campbell-Hausdorff expansion. *J. Chem. Theory Comput.* **2021**, *17*, 66–78.
- (36) Liu, J.; Wan, L.; Li, Z.; Yang, J. Simulating periodic systems on a quantum computer using molecular orbitals. *J. Chem. Theory Comput.* **2020**, *16*, 6904–6914.
- (37) Khamoshi, A.; Chen, G. P.; Evangelista, F. A.; Scuseria, G. E. AGP-based unitary coupled cluster theory for quantum computers. *Quantum Sci. Technol.* **2023**, *8*, 015006.
- (38) Powell, M. J. *A direct search optimization method that models the objective and constraint functions by linear interpolation*; Springer, 1994.
- (39) Spall, J. C. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Trans. Autom. Control* **1992**, *37*, 332–341.
- (40) Kandala, A.; Temme, K.; Córcoles, A. D.; Mezzacapo, A.; Chow, J. M.; Gambetta, J. M. Error mitigation extends the computational reach of a noisy quantum processor. *Nature* **2019**, *S67*, 491–495.
- (41) McCaskey, A. J.; Parks, Z. P.; Jakowski, J.; Moore, S. V.; Morris, T. D.; Humble, T. S.; Pooser, R. C. Quantum chemistry as a benchmark for near-term quantum computers. *npj Quantum Inf.* **2019**, *5*, 99.
- (42) Eddins, A.; Motta, M.; Gujarati, T. P.; Bravyi, S.; Mezzacapo, A.; Hadfield, C.; Sheldon, S. Doubling the size of quantum simulators by entanglement forging. *PRX Quantum* **2022**, *3*, 010309.
- (43) Parrish, R. M.; Iosue, J. T.; Ozaeta, A.; McMahon, P. L. A Jacobi diagonalization and Anderson acceleration algorithm for variational quantum algorithm parameter optimization. *arXiv* **2019**, arXiv:1904.03206; DOI: 10.48550/arXiv.1904.03206.
- (44) Nakanishi, K. M.; Fujii, K.; Todo, S. Sequential minimal optimization for quantum-classical hybrid algorithms. *Phys. Rev. Res.* **2020**, *2*, 043158.
- (45) Ostaszewski, M.; Grant, E.; Benedetti, M. Structure optimization for parameterized quantum circuits. *Quantum* **2021**, *5*, 391.
- (46) Powell, M. J. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *Comput. J.* **1964**, *7*, 155–162.
- (47) Press, W. H. *Numerical recipes: The art of scientific computing*, 3rd ed.; Cambridge University Press, 2007.
- (48) Brent, R. P. *Algorithms for minimization without derivatives*; Courier Corporation, 2013.
- (49) Kottmann, J. S.; Anand, A.; Aspuru-Guzik, A. A feasible approach for automatically differentiable unitary coupled-cluster on quantum computers. *Chem. Sci.* **2021**, *12*, 3497–3508.
- (50) Singh, H.; Majumder, S.; Mishra, S. Benchmarking of different optimizers in the variational quantum algorithms for applications in quantum chemistry. *J. Chem. Phys.* **2023**, *159*, 044117.
- (51) Romero, J.; Babbush, R.; McClean, J. R.; Hempel, C.; Love, P. J.; Aspuru-Guzik, A. Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz. *Quantum Sci. Technol.* **2019**, *4*, 014008.

- (52) Chen, J.; Cheng, H.-P.; Freericks, J. K. Quantum-inspired algorithm for the factorized form of unitary coupled cluster theory. *J. Chem. Theory Comput.* **2021**, *17*, 841–847.
- (53) Rubin, N. C.; Gunst, K.; White, A.; Freitag, L.; Throssell, K.; Chan, G. K.-L.; Babbush, R.; Shiozaki, T. The Fermionic quantum emulator. *Quantum* **2021**, *5*, 568.
- (54) Tsuchimochi, T.; Mori, Y.; Ten-no, S. L. Spin-projection for quantum computation: A low-depth approach to strong correlation. *Phys. Rev. Res.* **2020**, *2*, 043142.
- (55) Mehendale, S. G.; Peng, B.; Govind, N.; Alexeev, Y. Exploring Parameter Redundancy in the Unitary Coupled-Cluster Ansätze for Hybrid Variational Quantum Computing. *J. Phys. Chem. A* **2023**, *127*, 4526–4537.
- (56) Sokolov, I. O.; Barkoutsos, P. K.; Ollitrault, P. J.; Greenberg, D.; Rice, J.; Pistoia, M.; Tavernelli, I. Quantum orbital-optimized unitary coupled cluster methods in the strongly correlated regime: Can quantum algorithms outperform their classical equivalents? *J. Chem. Phys.* **2020**, *152*, 124107.
- (57) de Gracia Triviño, J. A.; Delcey, M. G.; Wendin, G. Complete active space methods for NISQ devices: The importance of canonical orbital optimization for accuracy and noise resilience. *J. Chem. Theory Comput.* **2023**, *19*, 2863–2872.
- (58) Filip, M.-A.; Fitzpatrick, N.; Ramo, D. M.; Thom, A. J. Reducing unitary coupled cluster circuit depth by classical stochastic amplitude prescreening. *Phys. Rev. Res.* **2022**, *4*, 023243.
- (59) Armaos, V.; Badounas, D. A.; Deligiannis, P.; Lianos, K.; Yordanov, Y. S. Efficient Parabolic Optimisation Algorithm for Adaptive VQE Implementations. *SN Comput. Sci.* **2022**, *3*, 443.
- (60) Zhang, S.-X.; Allcock, J.; Wan, Z.-Q.; Liu, S.; Sun, J.; Yu, H.; Yang, X.-H.; Qiu, J.; Ye, Z.; Chen, Y.-Q.; Lee, C.-K.; Zheng, Y.-C.; Jian, S.-K.; Yao, H.; Hsieh, C.-Y.; Zhang, S. Tensorcircuit: a quantum software framework for the NISQ era. *Quantum* **2023**, *7*, 912.
- (61) Li, W.; Allcock, J.; Cheng, L.; Zhang, S.-X.; Chen, Y.-Q.; Mailoa, J. P.; Shuai, Z.; Zhang, S. TenCirChem: An Efficient Quantum Computational Chemistry Package for the NISQ Era. *J. Chem. Theory Comput.* **2023**, *19*, 3966–3981.
- (62) Sun, Q.; Berkelbach, T. C.; Blunt, N. S.; Booth, G. H.; Guo, S.; Li, Z.; Liu, J.; McClain, J.; Sayfutyarova, E. R.; Sharma, S.; Wouters, S.; Chan, G. K.-L. PySCF: the Python-based simulations of chemistry framework. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2018**, *8*, e1340.
- (63) Sun, Q.; Zhang, X.; Banerjee, S.; Bao, P.; Barbry, M.; Blunt, N. S.; Bogdanov, N. A.; Booth, G. H.; Chen, J.; Cui, Z.-H. Recent developments in the PySCF program package. *J. Chem. Phys.* **2020**, *153*, 024109.
- (64) Brodtkorb, P. A.; D'Errico, J. *numdifftools 0.9.11*; 2015; <https://github.com/pbrod/numdifftools> (accessed 2024-04-11).
- (65) Nelder, J. A.; Mead, R. A simplex method for function minimization. *Comput. J.* **1965**, *7*, 308–313.
- (66) Bonet-Monroig, X.; Wang, H.; Vermetten, D.; Senjean, B.; Moussa, C.; Bäck, T.; Dunjko, V.; O'Brien, T. E. Performance comparison of optimization methods on variational quantum algorithms. *Phys. Rev. A* **2023**, *107*, 032407.
- (67) Cao, C.; Hu, J.; Zhang, W.; Xu, X.; Chen, D.; Yu, F.; Li, J.; Hu, H.-S.; Lv, D.; Yung, M.-H. Progress toward larger molecular simulation on a quantum computer: Simulating a system with up to 28 qubits accelerated by point-group symmetry. *Phys. Rev. A* **2022**, *105*, 062452.
- (68) Burton, H. G.; Marti-Dafcik, D.; Tew, D. P.; Wales, D. J. Exact electronic states with shallow quantum circuits from global optimization. *npj Quantum Inf.* **2023**, *9*, 75.
- (69) Burton, H. G. Accurate and gate-efficient quantum ansätze for electronic states without adaptive optimization. *arXiv* **2023**, arXiv:2312.09761; DOI: [10.48550/arXiv.2312.09761](https://doi.org/10.48550/arXiv.2312.09761).
- (70) Li, W.; Ren, J.; Huai, S.; Cai, T.; Shuai, Z.; Zhang, S. Efficient quantum simulation of electron-phonon systems by variational basis state encoder. *Phys. Rev. Res.* **2023**, *5*, 023046.
- (71) Bravyi, S. B.; Kitaev, A. Y. Fermionic quantum computation. *Ann. Phys.* **2002**, *298*, 210–226.
- (72) Seeley, J. T.; Richard, M. J.; Love, P. J. The Bravyi-Kitaev transformation for quantum computation of electronic structure. *J. Chem. Phys.* **2012**, *137*, 224109.
- (73) Zhou, L.; Wang, S.-T.; Choi, S.; Pichler, H.; Lukin, M. D. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Phys. Rev. X* **2020**, *10*, 021067.
- (74) Ravi, G. S.; Gokhale, P.; Ding, Y.; Kirby, W.; Smith, K.; Baker, J. M.; Love, P. J.; Hoffmann, H.; Brown, K. R.; Chong, F. T. CAFQA: A classical simulation bootstrap for variational quantum algorithms. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, 2022; Vol. 1, pp 15–29.