Optimizing Variational Quantum Algorithms with qBang: Efficiently Interweaving Metric and Momentum to Navigate Flat Energy Landscapes

David Fitzek[1,2], Robert S. Jonsson[1,3], Werner Dobrautz[4], and Christian Schäfer[1,5]

[1]Department of Microtechnology and Nanoscience, MC2, Chalmers University of Technology, 412 96 Gothenburg, Sweden
[2]Volvo Group Trucks Technology, 405 08 Gothenburg, Sweden
[3]Future Technologies, Saab Surveillance, 412 76 Gothenburg, Sweden
[4]Department of Chemistry and Chemical Engineering, Chalmers University of Technology, 412 96 Gothenburg, Sweden
[5]Department of Physics, Chalmers University of Technology, 412 96 Gothenburg, Sweden

Variational quantum algorithms (VQAs) represent a promising approach to utilizing current quantum computing infrastructures. VQAs are based on a parameterized quantum circuit optimized in a closed loop via a classical algorithm. This hybrid approach reduces the quantum processing unit load but comes at the cost of a classical optimization that can feature a flat energy landscape. Existing optimization techniques, including either imaginary time-propagation, natural gradient, or momentum-based approaches, are promising candidates but place either a significant burden on the quantum device or suffer frequently from slow convergence. In this work, we propose the quantum Broyden adaptive natural gradient (qBang) approach, a novel optimizer that aims to distill the best aspects of existing approaches. By employing the Broyden approach to approximate updates in the Fisher information matrix and combining it with a momentum-based algorithm, qBang reduces quantum-resource requirements while performing better than more resource-demanding alternatives. Benchmarks for the barren plateau, quantum chemistry, and the max-cut problem demonstrate an overall stable performance with a clear improvement over existing techniques in the case of flat (but not exponentially flat) optimization landscapes. qBang introduces a new development strategy for gradient-based VQAs with a plethora of possible improvements.

David Fitzek: davidfi@chalmers.se
Robert S. Jonsson: robejons@chalmers.se
Werner Dobrautz: werner.dobrautz@gmail.com
Christian Schäfer: christian.schaefer.physics@gmail.com

1 Introduction

Fostered by its anticipated potential, recent technological progress, and a surge of widespread interest, quantum computing is approaching the next level of popularity. Despite its impressive progress over the past years [1, 2, 3, 4, 5] much remains to be accomplished before a practical use moves into reach [6, 7]. Two of the most severe constraints are the limited number of qubits and short coherence times [8]. In order to combat those challenges, mixed quantum-classical algorithms, labeled variational quantum algorithms (VQAs) [1, 9, 10, 11, 12, 2, 3], have been devised. VQAs split an optimization task into two entwined steps: (i) an energy estimation using the quantum processing unit (QPU) and (ii) a classical optimization of the characterizing parameters. Due to the existing challenges, the aim of developing VQAs is to ensure convergence while limiting the number of function evaluations on the QPU to a minimum.

Classical optimizers have come a long way, from vanilla gradient descent, over natural gradient methods to the modern widely used adaptive gradient-based methods (Adam) [13]. Similar gradient-based approaches have been introduced for quantum algorithms [14, 15, 16]. The nature of quantum mechanics implies that, as the system size grows, the associated Hilbert space

grows exponentially. While it is our goal to leverage this complexity, the majority of available eigenstates are closely packed in energy, mimicking de facto thermal behavior for a local operator according to the eigenstate thermalization hypothesis [17]. Consequently, gradients, which result in small local changes in a high-dimensional Hilbert space, decrease exponentially with increasing system size, a feature known as a barren plateau, making parametrized quantum circuits (PQCs) prone to poor convergence. Albeit not directly mitigating BPs [18, 19, 20, 21], higher-order derivative information can aid in maneuvering the optimization landscape by accounting for its local curvature or metric [22, 23]. A quantity related to local curvature is the quantum Fisher information matrix (QFIM), which appears also in the context of multi-parameter estimation [24].

Estimating gradients and higher-order derivatives of quantum circuits is, unfortunately, costly, and requires many function evaluations. Given its quadratic form, for $n_{\boldsymbol{\theta}}$ parameters the QFIM requires $\mathcal{O}({n_{\boldsymbol{\theta}}}^2)$ function evaluations which, considering the cost of measurements, renders its use for relevant problems challenging. Stokes *et al.* [22] introduced for pure quantum states the quantum natural gradient (QNG). Block-diagonal approximations of the latter require only a linear amount of function calls but discard essential information about parameter correlation which severely limits its performance [25]. Generalizations of QNG to non-unitary circuits [26] as well as alternative approximation strategies have been proposed [27, 28]. While the specific cost of estimating the QFIM depends on the specific problem at hand, the cost for performing $\mathcal{O}({n_{\boldsymbol{\theta}}}^2)$ evaluations is particularly prohibitive in systems that feature vanishing gradients due to a quickly rising number of variables (e.g., the BP circuit [18]). Practical use of VQAs requires the availability of optimization strategies that provide reliable predictions with as few as possible evaluations on the QPU.

In this work, we introduce the quantum Broyden adaptive natural gradient (qBang) approach – an optimization strategy that augments the reliable momentum-based optimization Adam with an efficient update of the local metric based on the QFIM using the Broyden method [29]. After initialization, qBang requires only $\mathcal{O}(n_{\boldsymbol{\theta}})$ evaluations and, yet, shows considerable performance gain over QNG, Adam, and even quantum imaginary time evolution (QITE) [30, 31, 32, 33] on flat optimization landscapes.

The remainder of this article is structured as follows: Section 2.1 recapitulates VQAs, comprising the quantum approximate optimization algorithm (QAOA) and the variational quantum eigensolver (VQE), followed by a brief review of gradient-based optimization paradigms in Sec. 2.2. Sec. 2.3 subsequently introduces the newly developed qBang algorithm which is extensively benchmarked and discussed in Sec. 3 for BP, max-cut, and quantum chemical systems. We finally conclude the discussion in Sec. 4 and provide an outlook toward possible applications, improvements, and future challenges.

2 Theory

2.1 Variational quantum algorithms

VQAs are a collection of practically applicable algorithms that harness the computational capabilities of programmable quantum devices [1, 9, 32]. These algorithms are well suited for the hardware constraints imposed by the current generation of quantum computers, namely short coherence times, noisy operations, and the limited number of qubits [8]. These near-term algorithms have been proposed for a wide range of applications, including quantum chemistry [3], classical optimization [2] and machine learning [1, 34].

VQAs are composed of three key elements, which are represented in Fig. 1. The first component is the objective/cost function to be minimized. In our work, the cost function is expressed as the expectation value of the Hamiltonian,

$$\mathcal{L}(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})|\,\hat{H}\,|\psi(\boldsymbol{\theta})\rangle\,, \tag{1}$$

and provides information about the energy of the ground state of the Hamiltonian $\hat{H}$. Depending on the complexity of the Hamiltonian, different Pauli strings have to be measured to get an accurate estimate of the energy. The state $|\psi(\boldsymbol{\theta})\rangle$ is represented by a parametrized quantum circuit, and the optimizable parameters of the circuit are denoted as $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_{n_\theta})^\top$. These parameters commonly represent the angles of unitary rotation operators. The Hamiltonian is composed of quantum operators that encode information about a chemical or classical system, such as a

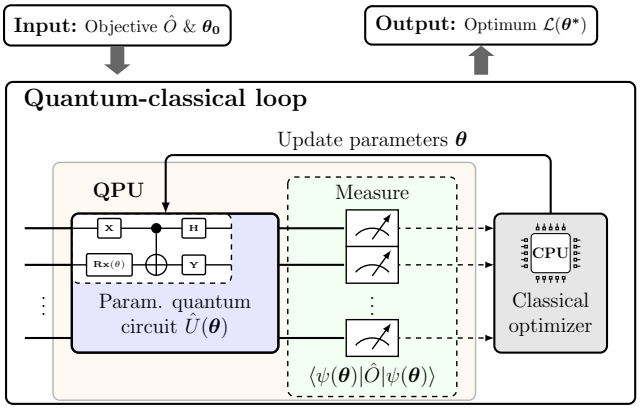

Figure 1: A diagrammatic representation of a VQA consists of three main elements: an objective function that defines the problem to be solved, a PQC $\hat{U}(\boldsymbol{\theta})$ in which parameters $\boldsymbol{\theta}$ are adjusted to minimize the objective and a classical optimizer that performs this minimization. The inputs for a VQA are the circuit Ansatz and initial parameter $\boldsymbol{\theta}_0$ values, while the outputs are the optimized parameter values $\boldsymbol{\theta}^*$ and the minimum value of the objective function, $\langle\psi(\boldsymbol{\theta})|\,\hat{O}\,|\psi(\boldsymbol{\theta})\rangle$.

molecule or an optimization problem. The second component is the problem-specific circuit ansatz, $|\psi(\boldsymbol{\theta})\rangle$. These ansätze are tailored to the specific problem, and numerous works focus on finding optimal PQCs [35, 36, 37]. A shared aspect is the use of only unitary operations, a limitation that will become relevant in the subsequent sections. The final component is the classical optimizer, which is used to find parameters that minimize the objective function [2, 38, 26, 22].

The task of VQAs is to optimize the cost function, Eq. (1), by adjusting the tunable parameters $\boldsymbol{\theta}$ of the circuit ansatz in a closed loop. This is done by iterating between evaluating the cost function on the quantum computer and updating the parameters using a classical optimizer. The objective is to find the set of parameters, $\boldsymbol{\theta}^*$, that minimizes the cost function and provides a solution to the problem at hand. The process of evaluating the cost function and updating the parameters is repeated until the cost function converges to its minimum value or a stopping criterion is met. Current limitations in the available complexity of circuits are thus circumvented by dividing the optimization problem into small sets of quantum evaluations steered via classical parameter optimization. The circuit ansatz, cost function, and classical optimizer are problem-specific, and the choice of these components can significantly affect the algorithm's performance.

VQAs offer a versatile framework that can be broadly categorized into several areas of application. While QAOA [36] is often employed for classical optimization problems and VQE [9, 35, 3] is commonly used for solving quantum eigenvalue problems, these categories are not exhaustive.

QAOA has been proposed to solve various classical optimization problems [36, 39, 1, 40, 41] and is a candidate for hybrid quantum-classical computation. Here, optimization problems are encoded into an Ising Hamiltonian [39]. QAOA typically suggests a circuit ansatz $|\psi(\boldsymbol{\theta})\rangle$ composed of the consecutive application of two non-commuting operators. One operator encodes the optimization problem and the other serves as a mixing Hamiltonian. The goal is to optimize the parameters, $\boldsymbol{\theta}$, of the quantum circuit to minimize $\mathcal{L}(\boldsymbol{\theta})$, and thereby find the solution to the optimization problem. Once the quantum circuit has been optimized, bitstrings are sampled to obtain approximate solutions to the classical optimization problem.

In contrast, the VQE is the most widely studied quantum algorithm to minimize a given cost function, usually the energy, of a given quantum system, Eq. (1). A prominent example is the solution of Schrödinger's equation for molecular systems. A selected PQC is initialized, and the corresponding energy of the output state is subsequently evaluated on a quantum computer. Information about energy, gradients, and the metric can be inferred from multiple evaluations of the circuit and then used to update the parameters of the circuit with classical optimization methods [42]. This process is repeated until the expectation value converges to the ground-state energy of the system (see Section 2.1). The VQE algorithm has been applied in various fields, including quantum chemistry [43] and materials science [44].

2.2 Existing optimization paradigms

Here we review the existing optimization paradigms that inspire the qBang approach.

2.2.1 Gradient-based Optimization

A vital component of every variational algorithm is the classical optimizer. Here, the task of the classical computer is to iterate the parameters from an initial guess $\boldsymbol{\theta}_0$ such that the cost,

Eq. (1), is minimized. Generally, this requires several iterations, depending on the quality of the initial guess. Assuming the cost function is differentiable, this procedure can be realized with *gradient descent* (GD). GD uses the parameter update rule $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \nabla \mathcal{L}_k$, where the step size $\eta \in \mathbb{R}^+$ controls how much each iteration is allowed to change the parameters and $\nabla \mathcal{L}_k \equiv \nabla \mathcal{L}(\boldsymbol{\theta}_k)$ is the gradient of the cost function at iteration k. The norm of the gradient $\|\nabla \mathcal{L}_k\|_2$ can be used as a criterion to determine when to stop the GD algorithm, as a zero norm gradient implies a stationary point. Gradients of quantum circuits can be obtained via finite-difference methods, linear combination of unitaries [16] and without the need for additional hardware by evaluating the cost function at two shifted parameter positions and using the rescaled difference of the results as an unbiased estimate of the derivative [45, 38, 16].

GD-based methods have apparent limitations. If the cost function is relatively flat, the gradient will be small, and the GD may require unfeasibly many iterations to converge, even on ideal quantum devices. The noisy results on realistic devices put additional strain on the optimizer to escape flat energy landscapes as quickly as possible. As long as the cost function gradients are not completely vanishing, this problem may be mitigated by the extension of GD to include higher-order derivatives. For a second-order algorithm, this introduces the Hessian $\mathbf{H}$ and results in the Newton method, $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \mathbf{H}_k^{-1} \nabla \mathcal{L}_k$. However, these higher-order methods are not always applicable, as the Hessian may not be positive semi-definite [46, 47]. Additionally, computing the Hessian is computationally expensive if the parameter space is large. To overcome these challenges, there are several quasi-Newton methods that can efficiently estimate the Hessian, such as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm or the Gauss-Newton method [48, 29, 49, 50].

Other methods exist that are tailored to navigate flat energy landscapes. For an intuitive picture, consider a ball rolling down in a frictionless bowl. Instead of stopping at the bottom, the accumulated momentum pushes it forward and keeps the ball rolling back and forth. This idea is used in momentum-based optimizers, illustrated in a simplified form

$$\boldsymbol{m}_k = \beta \boldsymbol{m}_{k-1} + (1-\beta) \nabla \mathcal{L}_k \tag{2}$$
$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \boldsymbol{m}_k, \tag{3}$$

where each step is a linear combination of the previous update and the current gradient, with $\boldsymbol{m}_k$ being the momentum accumulated during the optimization process, β the decay rate and η the step size. Compared to GD, these methods are more effective at escaping local minima [51]. The Adam [13] momentum-based optimizer is widely used throughout different scientific disciplines and has proven versatile and consistent in performance.

The optimization of VQAs can suffer when the energy surface becomes flat. To handle this issue, two directions can be taken. One approach is to find a good initial state that can be easily obtained and prepared on the quantum device. A typical example for chemistry applications is the uncorrelated Hartree–Fock state, but it can be expected that more complex systems will require correlated initial states. In the second approach, we utilize information about the local metric to guide each step toward the minimum, which will be discussed in detail in the following section. Overall, finding practical solutions to this problem is crucial for successfully implementing VQAs.

2.2.2 Metric-informed Optimization: quantum imaginary time evolution and quantum natural gradient

As stated above, VQAs rely on a parametrization of the wave function in which the parameters represent phases of unitary gates acting on an input state. A small change in a parameter $\delta\theta_i$ not only results in changes in the observable of interest, as utilized by GD, but also in the associated metric $\langle \psi(\delta\theta_j) | \psi(\delta\theta_i) \rangle$. This additional information can provide a more suitable direction for the optimization trajectory. We will briefly review QITE and QNG, representing the two most widely discussed metric-informed optimization strategies.

QITE [30, 31, 32, 33] is based on the "Wick-rotated"($\tau = it$) [52] imaginary time Schrödinger

equation

$$\begin{aligned} \frac{\partial |\Psi(\tau)\rangle}{\partial \tau} &= -\hat{H} |\Psi(\tau)\rangle \\ \text{or } |\Psi(\tau+\Delta\tau)\rangle &= N(\tau)^{-1} \mathrm{e}^{-\Delta\tau \hat{H}} |\Psi(\tau)\rangle , \\ \text{with } N(\tau) &= \sqrt{\langle\Psi(\tau)| \mathrm{e}^{-2\Delta\tau\hat{H}} |\Psi(\tau)\rangle} \end{aligned} \tag{4}$$

and is a quantum algorithm to find the ground and excited states [53] of a quantum system. It is a variant of the imaginary time evolution (ITE) algorithm [54, 55, 56, 57], which is a well-established technique in "classical" computational physics for finding the ground state of a system. The iterative application of the exponential operator with sufficiently small time-steps $\Delta\tau$ [56] is exponentially damping higher energy contributions, resulting in a convergence to the ground state $|\Psi_0\rangle$ if the initial state $|\Psi(0)\rangle$ has a non-zero overlap with the ground state [30, 31]. However, since $\mathrm{e}^{-\Delta\tau\hat{H}}$ is not unitary, it is not straightforward to directly implement ITE on quantum hardware. One option, which we will pursue in this work, is to cast QITE into a hybrid quantum-classical variational form (VarQITE) [31, 32] (Fig. 1), where the target state $|\Psi(\tau)\rangle$ is encoded by a PQC $|\psi(\boldsymbol{\theta}(\tau))\rangle = \hat{U}(\boldsymbol{\theta}(\tau))|\psi_0\rangle$ and the time-evolution is mapped to the parameters $\boldsymbol{\theta}(\tau)$ of the variational ansatz. The rule to update the parameters $\boldsymbol{\theta}_k$ for the next iteration $k+1$ at (imaginary) time $\tau+\Delta\tau$ is obtained by applying McLachlan's variational principle [58] to Eq. (4), minimizing the difference of the time evolution of the ansatz state $|\psi(\tau)\rangle \equiv |\psi(\boldsymbol{\theta}(\tau))\rangle$ to the exact imaginary time evolution

$$\delta \| \left(\partial/\partial\tau + \hat{H} - E_\tau \right) |\psi(\tau)\rangle \|_2 = 0, \tag{5}$$

where $\||\psi\rangle\|_2 = \sqrt{\langle\psi|\psi\rangle}$ is the 2-norm of a quantum state $|\psi\rangle$ and $E_\tau = \langle\psi(\tau)|\hat{H}|\psi(\tau)\rangle$ is the expected energy at time τ. Solving Eq. (5) yields the imaginary-time derivative of the parameters

$$\frac{\partial\boldsymbol{\theta}}{\partial\tau} = -2\,\mathbf{F}^{-1}\nabla\mathcal{L}, \tag{6}$$

where $\mathbf{F}$ is the QFIM and $\nabla\mathcal{L}$ the cost gradient. Eq. (6) allows updating the parameters for the next iteration, i.e., with a fixed time-step $\Delta\tau$ and the Euler method

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \Delta\tau\frac{\partial\boldsymbol{\theta}}{\partial\tau} = \boldsymbol{\theta}_k - \frac{\Delta\tau}{2}\mathbf{F}_k^{-1}\nabla\mathcal{L}_k, \tag{7}$$

or higher-order methods [59]. $\Delta\tau$ is equivalent to a step size, η, in the above mentioned GD update rule. The elements of the QFIM are given by

$$\mathbf{F}_{ij} = 4\mathrm{Re}\left[\left\langle\partial_{\theta_i}\psi|\partial_{\theta_j}\psi\right\rangle - \langle\partial_{\theta_i}\psi|\psi\rangle\left\langle\psi|\partial_{\theta_j}\psi\right\rangle\right], \tag{8}$$

where, $\partial_{\theta_i} \equiv \frac{\partial}{\partial\theta_i}$. There is a close relation between the QFIM and the Fubini-Study metric, which is the metric of parametrized pure quantum states $|\psi\rangle$, see the Supplemental Information (SI) Section E and Refs. numbers [60, 61, 62, 63, 64, 65, 24, 66, 67] for details. The QFIM $\mathbf{F}$ encodes the nontrivial geometry of the parameter space [68, 67] and is the quantum-analog of the classical Fisher information matrix, which is the unique Riemannian metric associated to a probability density function [69, 70, 71].

QNG [22] is another metric-informed optimization technique based on the principles of natural gradient descent by Amari *et al.* [72, 73, 74, 75, 69], initially developed for optimizing neural networks. As VarQITE, the natural gradient considers the geometry of the function's parameter space and is calculated using the inverse of the QFIM [24, 76]. Thus, using the QNG results in steps that are more aligned with the geometry of the parameter space and allows for faster convergence, crossing of local minima, and helps the algorithm to escape regions with vanishing gradients [22, 25, 23, 77, 18, 78, 20]. VarQITE and QNG are equivalent when the energy of the system, $E = \langle\hat{H}\rangle$, is used as the cost function [73, 46, 22, 26], as considered in this work, see Eq. (1).

The major drawback of QITE and QNG is that computing the entire QFIM for an ansatz with $n_{\boldsymbol{\theta}}$ parameters is computationally expensive and requires measuring $\mathcal{O}(n_{\boldsymbol{\theta}}^2)$ terms every iteration. Existing approximations such as the (block-) diagonal approximation of Stokes *et al.* [22] reduce the scaling to linear in the number of parameters, but discarding the off-diagonal elements omits essential information about correlation within the system and leads to an overall suboptimal performance [25].

The metric $\mathbf{F}$ and the gradient $\nabla\mathcal{L}$ can be directly evaluated on quantum hardware [16, 32, 79, 80]. It should be noted that the metric is frequently singular due to over-parametrization of the chosen circuit ansatz and requires regularization [22, 25] or comparable strategies [81, 59].

2.3 Quantum Broyden Adaptive Natural Gradient

In this section, we introduce qBang, that combines the Broyden quasi-Newton method with the natural gradient and adaptive momentum approaches. We discuss the core components of qBang, as well as its motivation, mechanics, and resources required on the programmable quantum device. We also introduce a simplified version of our optimization approach, which we refer to as qBroyden.

The algorithms qBang and qBroyden utilize an adaptive approach to approximate the QFIM, drawing inspiration from the works of Amari, Park, and Fukumizu [75, 82]. The intuition behind this approach can be understood as follows. We would like to retain the benefits of the natural gradient method without computing the QFIM at each iteration. For this reason, we assume that the QFIM varies slowly as the parameter space is traversed. For time step k, we use a metric denoted by the matrix $\mathbf{B}_k$. Between steps, the metric is updated with a rank-1 perturbation given by the current gradient. In particular, $\mathbf{B}_{k+1}$ is realized as a low-pass filter process with learning rate ε_k, allowing the metric to pick up momentum as the parameter space is traversed, given by the relation

$$\mathbf{B}_{k+1} = (1-\varepsilon_k)\mathbf{B}_k + \varepsilon_k \nabla\mathcal{L}_k \nabla\mathcal{L}_k^\top. \tag{9}$$

Conceptually, this updates the local metric with an approximation of the Hessian. In the classical setting, the Hessian is equivalent to the Fisher information matrix for certain classes of optimisation problems, e.g., with Gaussian statistics or if the connection between the probability of encountering a given state decreases exponentially with its energy density (see SI Section E). More generally, the connection to curvature is also found in the equivalence between the *classical* Fisher information matrix and the Hessian of the relative entropy between two parametrically separated distributions [83]. We want to note that recently, Dash *et al.* [84] have related the QFIM with the Hessian in the context of neural quantum states by using the infidelity with respect to the exact ground state as the cost function. The famous BFGS algorithm uses similar ideas as Eq. (9) but differs in approximating the Hessian using *two* rank-1 updates.

Instead of updating and then inverting $\mathbf{B}_{k+1}$, we utilise the Sherman-Morrison formula to equivalently perform the update on the inverse as

$$\mathbf{B}_{k+1}^{-1} = \left[\mathbb{1} - \frac{\varepsilon_k \mathbf{B}_k^{-1}\nabla\mathcal{L}_k\nabla\mathcal{L}_k^\top}{1-\varepsilon_k(1-\nabla\mathcal{L}_k^\top\mathbf{B}_k^{-1}\nabla\mathcal{L}_k)}\right]\frac{\mathbf{B}_k^{-1}}{1-\varepsilon_k}. \tag{10}$$

We select the hyperparameter ε_k according to a decaying filter $\varepsilon_k = \varepsilon_0/(k+1)$ [73].

Algorithm 1 presents the pseudo-code of the qBang optimizer, which will be briefly exercised in the following. The algorithm takes as input the learning rates $\eta = 0.01$ and $\varepsilon_0 = 0.2$, the decay rates $\beta_1 = 0.9$ and $\beta_2 = 0.999$, the convergence criterion γ, and the PQC $U(\boldsymbol{\theta})$ with the initial parameter vector $\boldsymbol{\theta}_0 \in \mathbb{R}^{n_\theta}$. In the initialization step, the algorithm sets the iteration counter $k \leftarrow 0$, the momentum vector $\boldsymbol{m}_{-1} \leftarrow \mathbf{0}$ and the biased variance vector $\boldsymbol{v}_{-1} \leftarrow \mathbf{0}$, whose role will become apparent in the following. The matrix $\mathbf{B}_0$ is initialized using, either, the full Fisher information matrix ($\mathbf{F}$) or an approximation as introduced in [22]. Other choices for the matrix $\mathbf{B}_0$ would result in variations of the algorithm. The optimization starts with the estimation of the cost function $\mathcal{L}(\boldsymbol{\theta}_k)$ and its gradient $\nabla\mathcal{L}(\boldsymbol{\theta}_k)$ through quantum circuits, followed by the update of the momentum and variance vectors, similar to the Adam algorithm [13]. Specifically, the algorithm calculates a weighted average of past gradients $\boldsymbol{m}_k$, with the weight given by a parameter β_1, and uses this as a moving direction. It incorporates a moving average of the squared gradient, $\boldsymbol{v}_k \leftarrow \beta_2\boldsymbol{v}_{k-1} + (1-\beta_2)\,\nabla\mathcal{L}(\boldsymbol{\theta}_k)\odot\nabla\mathcal{L}(\boldsymbol{\theta}_k)$, with the weight given by a second parameter β_2. The vector $\boldsymbol{v}_k$ can be interpreted as the variance under the assumption of a vanishing average. Its magnitude provides information about the reliability of a gradient estimate. The moving averages are then adjusted for bias via division with $(1-\beta_{(1/2)}^{k+1})$, delivering $\widehat{\boldsymbol{m}}_k$ and $\widehat{\boldsymbol{v}}_k$. The variance vector $\widehat{\boldsymbol{v}}_k$ is then used to rescale the effective momenta into a sliding trust region $\{\boldsymbol{p}_k\}_l \leftarrow \{\widehat{\boldsymbol{m}}_k\}_l / \left(\sqrt{\{\widehat{\boldsymbol{v}}_k\}_l} + \kappa\right)$, $\forall l \in \{1,2,\ldots,p\}$, i.e., increasing the stability of the algorithm by shortening unreliable steps. Unless the convergence criterion is reached, the algorithm updates the parameter vector and the metric based on the update rule Eq. (10). It also rescales ε_k with the learning rate schedule, resulting in smaller updates with

increasing number of optimization steps. Otherwise, if the convergence criterion is satisfied, the algorithm stops the iteration and outputs the optimal parameter vector $\boldsymbol{\theta}^*$. We suggest reinitializing qBang once the update of the Fisher information matrix becomes minute, which might appear for particularly long optimization trajectories but has not been encountered in this work.

Algorithm 2 presents a simplified version of our optimization approach, which we refer to as qBroyden. Unlike qBang, qBroyden does not incorporate momentum and variance update rules and instead utilizes only the metric to update the parameter vector at each optimization step. Consequently, qBroyden is more closely related to QNG and VarQITE than qBang.

Our framework surrounding Eq. (10) has several advantages. Firstly, the Fisher information matrix is guaranteed to be positive semi-definite [24]. With the Gauss-Newton-like update, we maintain the positive semi-definiteness property through the optimisation, see SI Section G and Martens *et al.* [46]. In fact, we apply a small regularisation to the initial QFIM to ensure that $\mathbf{B}_0$ is positive definite. This is an important feature since it can happen that the QFIM is singular, particularly in over-parameterized systems with multiple layers. Additionally, because the QFIM is not recalculated at each time step, this framework significantly reduces the necessary number of circuit evaluations. Lastly, incorporating momentum updates not only results in superior speed but also increases the stability with respect to hyperparameter changes (illustrated in Sec. 3.4).

We want to note that a potential drawback of approximating the QFIM is that the resulting algorithms technically lose theoretically ensured convergence properties of QITE [30, 31]. However, this was not an issue for all the problems studied in this work. On the contrary, qBang ensured a faster and more stable convergence.

Regarding circuit evaluations, our proposed method reduces cost and increases efficiency. Each optimization step requires $\mathcal{O}(n_{\boldsymbol{\theta}})$ circuit evaluations, which is on par with Adam due to the parameter-shift rule [38, 45]. QNG without any approximation scales as $\mathcal{O}(n_{\boldsymbol{\theta}}^2)$ due to estimating the full Fisher information matrix [76]. Our proposed optimizers, qBang and qBroyden, require as many circuit evaluations in the first step as QNG, and only $\mathcal{O}(n_{\boldsymbol{\theta}})$ circuit evaluations per subsequent optimization step. The following sections demonstrate that the most striking advantage of qBang is its efficiency.

Algorithm 1 qBang

Input: learning rates $\eta = 0.01$, $\varepsilon_0 = 0.2$
Input: decay rates $\beta_1 = 0.9$ and $\beta_2 = 0.999$
Input: convergence criterion γ
Input: PQC $U(\boldsymbol{\theta})$
Input: Initial parameter vector $\boldsymbol{\theta}_0 \in \mathbb{R}^{n_\theta}$.
Initialization: $k \leftarrow 0$, $\boldsymbol{m}_{-1} \leftarrow \mathbf{0}$, $\boldsymbol{v}_{-1} \leftarrow \mathbf{0}$, $\mathbf{B}_0^{-1}$ via QNG, QFIM or Identity
`not_converged`$\leftarrow$`true`
while `not_converged` **do**
  QC: estimate $\mathcal{L}(\boldsymbol{\theta}_k)$
  QC: estimate $\nabla\mathcal{L}(\boldsymbol{\theta}_k)$
  $\boldsymbol{m}_k \leftarrow \beta_1 \boldsymbol{m}_{k-1} + (1-\beta_1)\nabla\mathcal{L}(\boldsymbol{\theta}_k)$
  $\boldsymbol{v}_k \leftarrow \beta_2 \boldsymbol{v}_{k-1} + (1-\beta_2)\nabla\mathcal{L}(\boldsymbol{\theta}_k) \odot \nabla\mathcal{L}(\boldsymbol{\theta}_k)$
  $\widehat{\boldsymbol{m}}_k \leftarrow \boldsymbol{m}_k / \left(1-\beta_1^{k+1}\right)$
  $\widehat{\boldsymbol{v}}_k \leftarrow \boldsymbol{v}_k / \left(1-\beta_2^{k+1}\right)$
  $\{\boldsymbol{p}_k\}_l \leftarrow \{\widehat{\boldsymbol{m}}_k\}_l / \left(\sqrt{\{\widehat{\boldsymbol{v}}_k\}_l} + \kappa\right), \quad \forall l \in \{1, 2, \ldots, p\}$
  if $\|\mathbf{B}_k^{-1}\boldsymbol{p}_k\|_2 > \gamma$ **then**
    $\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k - \eta\, \mathbf{B}_k^{-1}\boldsymbol{p}_k / ((k+2)-1)^{\epsilon_0}$
    $\varepsilon_k \leftarrow \frac{\varepsilon_0}{k+1}$
    $\mathbf{B}_{k+1}^{-1} \leftarrow$ Eq. (10)
    $k \leftarrow k+1$
  else
    `not_converged`$\leftarrow$`false`
    $\boldsymbol{\theta}^* \leftarrow \underset{\{\boldsymbol{\theta}_n\}_0^k}{\operatorname{argmin}} \mathcal{L}(\boldsymbol{\theta}_n)$
  end if
end while
return $\boldsymbol{\theta}^*$

Algorithm 2 qBroyden

Input: learning rates $\eta = 0.01$, $\varepsilon_0 = 0.2$
Input: convergence criterion γ
Input: PQC $U(\boldsymbol{\theta})$
Input: Initial parameter vector $\boldsymbol{\theta}_0 \in \mathbb{R}^{n_\theta}$.
Initialization: $k \leftarrow 0$, $\mathbf{B}_0^{-1}$ via QNG, QFIM or Identity
`not_converged`$\leftarrow$`true`
while `not_converged` **do**
    QC: estimate $\mathcal{L}(\boldsymbol{\theta}_k)$
    QC: estimate $\nabla\mathcal{L}(\boldsymbol{\theta}_k)$
    if $\|\mathbf{B}_k^{-1}\nabla\mathcal{L}(\boldsymbol{\theta}_k)\|_2 > \gamma$ **then**
        $\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k - \eta\,\mathbf{B}_k^{-1}\nabla\mathcal{L}(\boldsymbol{\theta}_k)$
        $\varepsilon_k \leftarrow \frac{\varepsilon_0}{k+1}$
        $\mathbf{B}_{k+1}^{-1} \leftarrow$ Eq. (10)
        $k \leftarrow k+1$
    else
        `not_converged`$\leftarrow$`false`
        $\boldsymbol{\theta}^* \leftarrow \underset{\{\boldsymbol{\theta}_n\}_0^k}{\operatorname{argmin}}\,\mathcal{L}(\boldsymbol{\theta}_n)$
    end if
end while
return $\boldsymbol{\theta}^*$

3 Results

This section presents numerical results from noise-free simulations of the new optimizers applied to several important classes of problems. We focus only on hybrid quantum-classical algorithms, which combine quantum and classical processing. The necessary quantum circuits for this study are available on GitHub [85] and additional information is provided in the SI.

Considering that quantum circuit queries are costly, our main goal is to reduce the number of circuit evaluations to obtain the parameters encoding the ground state of the PQC. Therefore, the key metric is the number of circuit evaluations. See Section 2.3 for the scaling of the number of circuit evaluations for each optimizer. Another important metric to assess the performance of the optimization is the approximation ratio. It describes how close the energy of the optimized quantum circuit is to the ground state energy. Formally, the approximation ratio is defined as

$$r = \frac{E_{\text{opt}} - E_{\text{max}}}{E_{\text{min}} - E_{\text{max}}}, \tag{11}$$

where E_{opt} is the energy obtained after optimization, and E_{min} and E_{max} are the theoretical minimum and maximum energy values, respectively.

We compare the optimizers Adam [13], QNG [22] with the block-diagonal approximation, as well as qBroyden and qBang using either the full or block-diagonal Fisher information in the first iteration. We largely exclude VarQITE in the following due to its prohibitive cost but show results for individual trajectories in SI Section A. It should be noted that the computational overhead for VarQITE might reduce in relation to gradient estimates when using advanced sampling techniques [86]. However, the cost of simulation with sampling is considerably larger than the here employed state propagation. For QNG and VarQITE, in case the QFIM is singular, we employ a Tikhonov regularization [87] and add 10^{-7} to its diagonal. Both algorithms of qBroyden and qBang use an initial filter parameter of $\epsilon_0 = 0.2$. For QNG and Adam, we use default parameters provided in [88].

We use identical step sizes for all algorithms to ensure a fair comparison but emphasize that the optimal step size will depend on the problem and algorithm at hand. Our investigation is comprehensive, accounting for statistical features in the random initialization, but not exhaustive, given the infinite combinations of hyperparameters and VQAs.

3.1 Barren plateau circuit

We start by illustrating the performance of the newly proposed optimizers on the BP circuit introduced in Ref. [18]. This quantum circuit was initially designed to show that highly expressible circuits come with a caveat, i.e., the more freedom we give a quantum circuit, the more difficult the optimization due to vanishing gradients in the exponentially growing Hilbert space [20]. The consequence: simple gradient-based optimizers fail.

Our circuit consists of an initial fixed layer of $R_y(\pi/4)$ gates acting on 9 qubits, followed by l layers of parameterized Pauli rotations with an entangling layer of controlled-Z gates. The objective operator is $\hat{H} = \hat{Z}_1\hat{Z}_2$ with a ground state energy of -1. The relative quality of the optimization will depend on the initial configuration, i.e., drawing a meaningful conclusion for the performance of an optimizer for a given problem requires a statistical analysis. In this manuscript, we obtain the expectation value $\langle\psi(\boldsymbol{\theta})|\hat{H}|\psi(\boldsymbol{\theta})\rangle$ for a parametrization of the wavefunction which

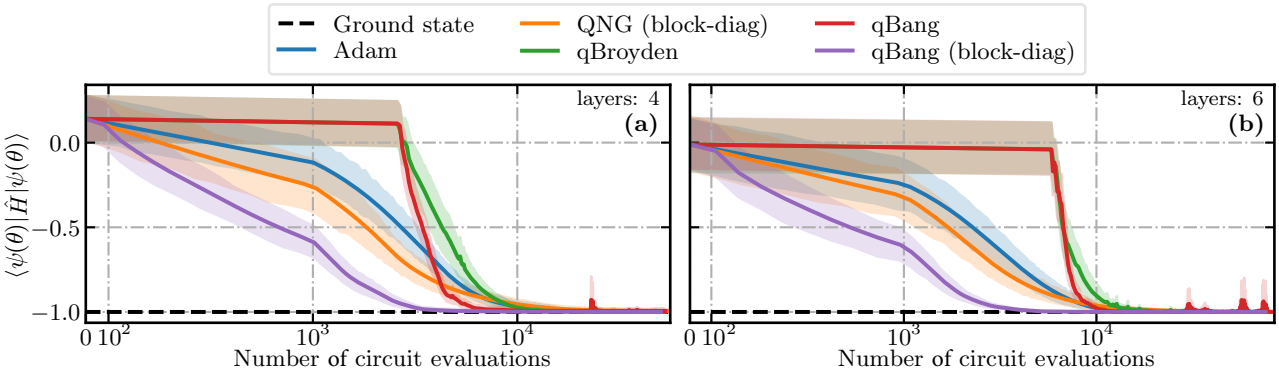

Figure 2: Comparison of optimization performance of Adam, QNG, qBroyden, and qBang in finding the ground state of the BP circuit. $\langle\psi(\boldsymbol{\theta})|\hat{H}|\psi(\boldsymbol{\theta})\rangle$ is shown as a function of the number of circuit evaluations. The step size is fixed at $\eta = 0.01$, and the results are averaged over 25 random initializations of parameters. The PQC used consists of 4 and 6 layers as depicted in subplots (a) and (b), respectively. The initial plateau in the optimization using qBroyden and qBang arises from the significant cost of initially measuring the QFIM.

is to be optimized. Our plots show the mean and variance of 25 trajectories with randomly initialized parameters (the same for all algorithms) and a step size of $\eta = 0.01$. The PQC considered has 4, 6, 8, and 10 layers, respectively. Figure 2 illustrates the performance as a function of circuit evaluations using 4 and 6 layers.

The QNG (block-diagonal) optimizer shows a moderate improvement over Adam within the initial 5000 evaluations for a small set of parameters but loses this initial advantage in the long run. qBang, on the other hand, is substantially faster. Approximating the QFIM as block-diagonal reduces the computational cost for the first iteration and explains the reduction in the required number of evaluations for the convergence of qBang (block-diag). The early plateau observed in the performance of qBroyden and qBang results from the upfront computational effort needed to estimate the QFIM. More relevant in practice is the number of circuit evaluations required to approximate the ground state accurately. To evaluate this, we determine the number of circuit evaluations necessary to reach an approximation ratio of 0.99 and present the results in Table 1. As shown in the table, qBang (block-diag) substantially outperforms Adam and QNG, requiring merely a third of the circuit evaluations.

While the BP circuit is of no practical use, it illustrates that qBang is a highly competitive optimizer when handling almost flat energy surfaces. We will briefly discuss classical optimization problems before moving on to quantum chemistry, arguably the most promising application for quantum computing to this date.

Table 1: Comparison of the number of circuit evaluations required for four optimizers to reach an approximation ratio of $r = 0.99$ for the BP circuit, with the results averaged over the 25 optimization trajectories. The PQC used range from 4, 6, 8, to 10 layers. "bd" indicates the block-diagonal approximation.

	Layers			
Optimizer	4	6	8	10
Adam	10700	10300	10200	13000
qBang	5980	9750	16900	25300
qBang (bd)	**3290**	**3490**	**4150**	**5330**
qBroyden	10300	13100	16100	25300
qBroyden (bd)	8990	11400	13800	17900
QNG (bd)	12300	17300	18500	26900

3.2 Quantum Approximate Optimization Algorithm

Classical combinatorial optimization can be just as hard as the optimization of quantum systems. QAOA represents a subclass of VQAs that handles the question if quantum computing could assist such classical combinatorial optimization.

We study the max-cut problem for which the cost (or energy) of the classical problem is mapped to an Ising Hamiltonian [39]. The Hamiltonian for the max-cut problem is encoded using eight qubits on the quantum device. The optimization performance of the different optimizers is displayed in Fig. 3 against the number of circuit evaluations. The results are averaged over five random initializations of parameters and a

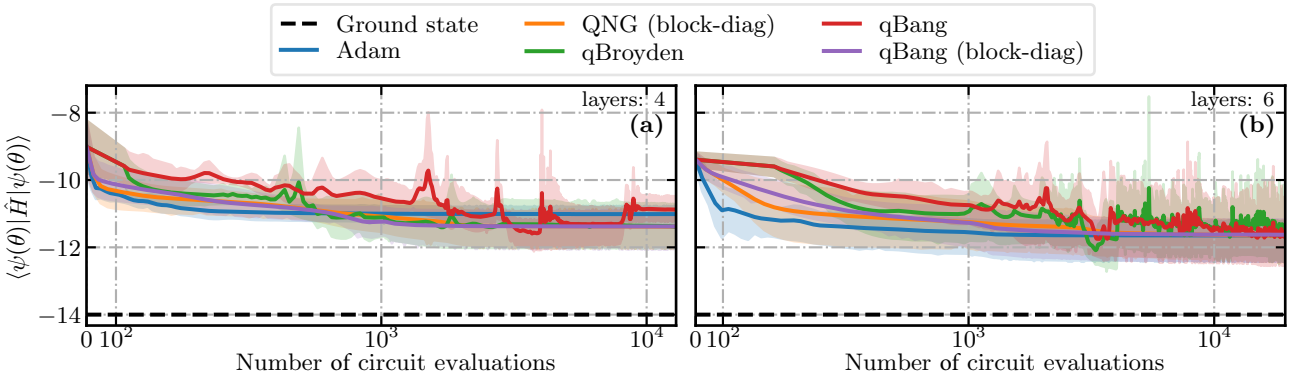

Figure 3: Ground state optimization performance of Adam, QNG with block-diagonal approximation, qBroyden with full Fisher information matrix, and qBang with full Fisher matrix and block-diagonal approximation of the QAOA circuit of an eight qubit max-cut problem instance using a PQC. The expectation value, $\langle\psi(\boldsymbol{\theta})|\hat{H}|\psi(\boldsymbol{\theta})\rangle$ is shown as a function of the number of circuit evaluations. The step size is fixed at $\eta = 0.06$, and the results are averaged over five random initializations of parameters. The PQC used consists of 4 and 6 layers as depicted in subplots (a) and (b), respectively.

step size of $\eta = 0.06$. We show the optimization trajectories for the 4- and 6-layered circuits in subplots (a) and (b), respectively. In Table 2, we compare the approximation ratios for the quantum state with the lowest expectation value, obtained by averaging over five trials for 4-, 6-, 8-, and 10-layered quantum circuits.

The optimization trajectories shown in Fig. 3 are similar in convergence behavior. One notable difference is the oscillations that qBroyden and qBang exhibit after many circuit evaluations using the full Fisher information. The oscillations result from incomplete updates of the off-diagonal elements in the Fisher information, which pushes the optimization away from the optimal direction. We elaborate on this feature in the SI Section A.1. Using the block-diagonal approximation ensures a smoother optimization. Alternatively, qBroyden and qBang could be reinitialized whenever instabilities occur.

Table 2 shows the approximation ratio averaged over five trajectories. Our proposed algorithms perform well on the 4- and 6-layered quantum circuits, while Adam outperforms all optimizers for 8- and 10-layers. Overall we observe only minor differences in convergence behavior, and the significant deviation from the optimal solution demonstrates that QAOAs face a serious challenge. It is important to note that the used circuit ansatz is likely incapable of representing a quantum state near the ground state of the classical optimization problem.

Table 2: Ground state energy approximation ratios of Adam, QNG with block-diagonal approximation, qBroyden, and qBang with full Fisher information and block-diagonal approximation for the max-cut Ising Hamiltonian. Results for PQCs with 4, 6, 8, and 10 layers are shown. The values are obtained from the quantum state with the expectation value closest to the ground state averaged over the five optimization pathways with a maximum length of 1100 optimization steps. "bd" indicates the block-diagonal approximation.

	Layers			
Optimizer	4	6	8	10
Adam	0.787	0.832	**0.896**	**0.91**
qBang	**0.829**	0.866	0.872	0.832
qBang (bd)	0.813	0.83	0.826	0.86
qBroyden	0.816	**0.89**	0.846	0.879
qBroyden (bd)	0.814	0.833	0.881	0.888
QNG (bd)	0.814	0.833	0.87	0.879

3.3 Variational Quantum Eigensolver

Solving Schrödinger's equation is challenging, yet essential to understand chemistry. In this study, we concentrate on investigating three prototypical molecular benchmark systems: hydrogen four (H_4), lithium hydride (LiH), and the water molecule (H_2O). We employed minimal basis sets (STO-6G) for all quantum chemistry problems and used a frozen core approximation for LiH and H_2O (with the 1s orbital of Li and O, respectively, frozen) [89]. To construct the quantum circuits, we used the Jordan-Wigner Fermion-to-qubit mapping and employed a hardware-efficient ansatz [35] that utilizes 8, 10, and 12 qubits for H_4, LiH, and H_2O, respectively. This ansatz

is composed of l layers, each comprising a tunable $R_y(\theta)$ gate on each qubit register, followed by a closed ring of CNOT gates. We compare the algorithm's performance with random and Hartree-Fock parameter initializations. Details of the molecular geometries and the Hartree-Fock parameter initialization can be found in the SI Section C. We used `Pennylane` [88] with the built-in `PySCF` interface [90] to setup our molecular systems and perform the Fermion-to-qubit mapping.

Our results provide insight into the feasibility and limitations of hardware-efficient circuit ansätze for preparing the ground state of molecular systems. In addition to assessing the optimization performance, we also analyze the physical soundness of the quantum states generated with the lowest overall energy. To this end, we calculate various observables, including the particle number, $\hat{N}$, the total spin projection observable, $\hat{S}_z$, and the total spin observable, $\hat{S}^2$, based on the optimized quantum state $|\psi(\boldsymbol{\theta})\rangle$.

3.3.1 Hydrogen square, H_4

We studied four hydrogen atoms, H_4, arranged in a square geometry with a side length of 2.25 Å. Figure 4 presents the mean energy as a function of the number of circuit evaluations for circuits with two and four layers. qBang requires substantially fewer circuit evaluations, qBroyden is on par with Adam and the performance of QNG is limited. The latter is likely due to the importance of off-diagonal components in the QFIM for correlated systems.

Upon further analysis of the quantum states generated by the PQCs, we find that, for all optimizers, the particle number $\langle\hat{N}\rangle$ and total spin projection $\langle\hat{S}_z\rangle$ observables are in proximity, but not in precise agreement with, the physical ground state (see Table 3). The deviations are most severe for the total spin $\langle\hat{S}^2\rangle$ and illustrate that the total energy is not the only observable of interest for the optimization in VQEs. This issue is a common challenge for hardware-efficient ansätze and stems from the choice of the circuit ansatz rather than the optimization algorithm itself (see also SI Section B.3). We verified the numerics with an equivalent Qiskit implementation providing the same hyperparameter and initial conditions leading to the same optimization trajectory.

3.3.2 Lithium hydride, LiH

We studied LiH at a bond distance of 1.59 Å with the 1s orbital of Li frozen. Figure 5 clarifies that the conclusions drawn for H_4 can be largely transferred to LiH: qBang vastly outperforms its competitors and consistently finds the best estimation for the energy closest to the ground state. Furthermore, once the optimum has been obtained, the comparably small variance of the 10 trajectories indicates a reliable optimization process. Consistent with H_4, $\langle\hat{S}^2\rangle$ challenges all optimizers (see Table 3).

3.3.3 Water, H_2O

We studied H_2O with an OH distance of 0.7 Å and with an $\angle$(HOH) of 104.48° with the 1s orbital of O frozen. Figure 6 illustrates the mean expectation value as a function of the number of circuit evaluations for quantum circuits consisting of two and four layers averaged over five trials. As before, qBang outperforms Adam and QNG. Interestingly, qBang with the full Fisher information is the only optimizer that manages to discover the exact ground state energy of the system in one of the optimization trajectories for two layers. The optimized circuits corresponding to the state with the lowest overall energy are analyzed in Table 4, showing an overall good performance of qBang and Adam.

Overall, qBang deliver accurate results for quantum chemistry applications at a discount. An important question remains: How resilient is this observation against changes in hyperparameters or noise?

3.4 Hyperparameter resilience

Hyperparameter resilience is important in ensuring robust and reliable optimization outcomes, especially in quantum chemistry, where the objective is to find a particular quantum state. A hyperparameter-resilient optimizer increases the chances of successfully finding the optimal solution and reduces the additional overhead of optimizing hyperparameters.

In Fig. 7, we investigate the effect of varying step size on the approximation ratio over the number of optimization steps in the BP circuit with 9 qubits and 5 layers. We use qBang, qBroyden, QNG with block diagonal, and Adam as the

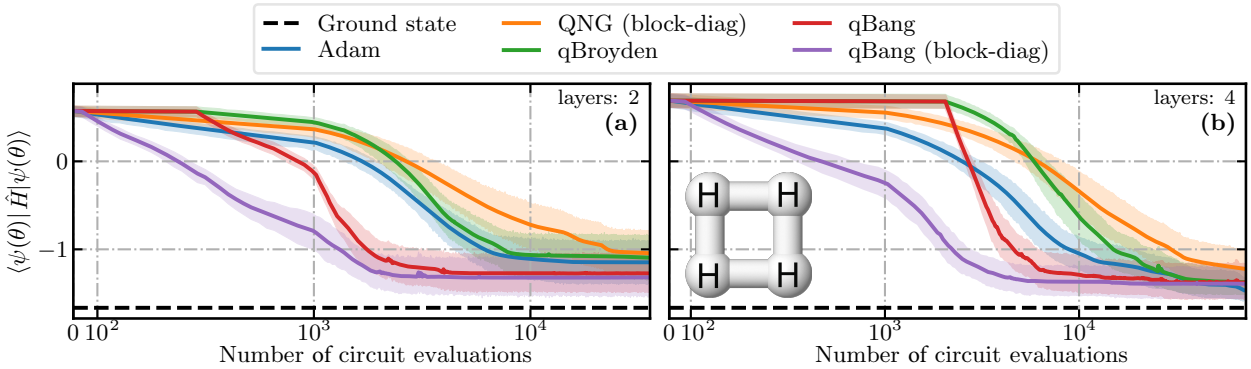

Figure 4: Comparison of optimization performance of Adam, QNG using the block-diagonal approximation, qBroyden using the full Fisher matrix, and qBang with the full Fisher information and block-diagonal approximation in finding the ground state of H_4 using a PQC. The expectation value, $\langle\psi(\boldsymbol{\theta})|\hat{H}|\psi(\boldsymbol{\theta})\rangle$ is shown as a function of the number of circuit evaluations. The step size is fixed at $\eta = 0.01$, and the results are averaged over 15 random initialization of parameters. The PQC consists of 2 and 4 layers, as shown in subplots (a) and (b), respectively. The initial plateau in the optimization using qBroyden and qBang arises from the significant cost of initially measuring the QFIM.

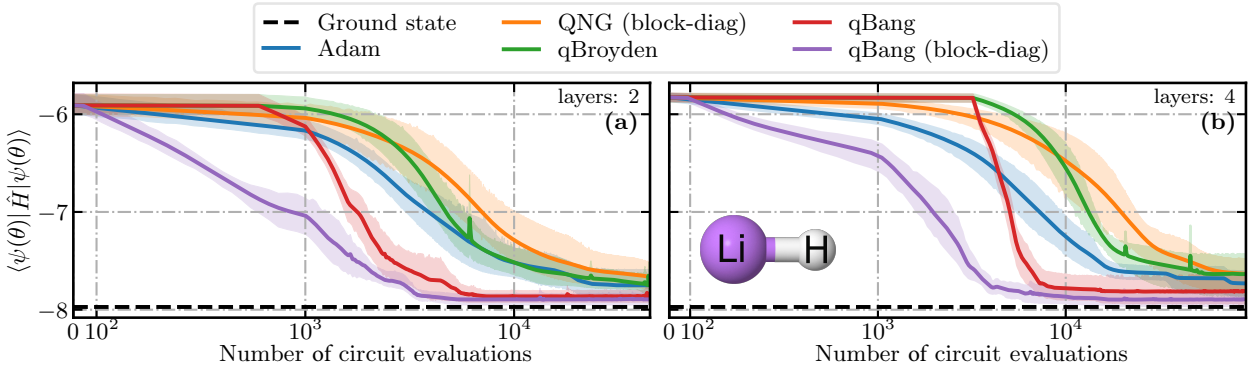

Figure 5: Comparison of optimization performance for four optimizers in finding the ground state of LiH using a PQC. The optimizers evaluated are Adam, Quantum Natural Gradient using the block-diagonal approximation, qBroyden using the full Fisher information matrix, and qBang with the full Fisher information and block-diagonal approximation. The expectation value, $\langle\psi(\boldsymbol{\theta})|\hat{H}|\psi(\boldsymbol{\theta})\rangle$ is shown as a function of the number of circuit evaluations. The step size is fixed at $\eta = 0.01$, and the results are averaged over 5 random initializations of parameters. The PQC used consists of 2 and 4 layers, as shown in subplots (a) and (b), respectively. The initial plateau in the optimization using qBroyden and qBang arises from the significant cost of initially measuring the QFIM.

Table 3: Converged optimization results for PQCs, representing H_4 and LiH. Results for H_4 are averaged over 15 optimization trajectories, while results for LiH are averaged over 10 optimization trajectories. The ground truth for each observable is shown in the column $\langle \hat{O} \rangle_\Psi$. Observables are calculated for circuits with layers ranging from 1 to 4 based on the variational quantum state with minimum expectation value along the optimization trajectory. Bold symbols indicate the optimizer that gets closest to the ground truth. The column labeled qBang shows results by starting with the full Fisher information matrix, and the column to the right labeled $F^{k=0}_{\text{block-diag}}$ are results starting with the block-diagonal approximation.

$\mathbf{H_4}$							
$\hat{O}$	$\langle \hat{O} \rangle_\Psi$	l	Adam	qBang	$F^{k=0}_{\text{block-diag}}$	qBroyden	QNG
$\hat{H}$	**-1.665**	1	**-1.08**	-1.05	-1.05	-1.03	-1.03
		2	-1.15	-1.21	**-1.25**	-1.21	-1.18
		3	**-1.37**	-1.34	-1.35	-1.35	-1.34
		4	**-1.46**	-1.42	-1.41	-1.4	-1.37
$\hat{N}$	**4**	1	**3.8**	3.67	3.62	3.6	3.59
		2	**3.93**	3.88	3.91	3.9	3.89
		3	3.83	3.84	**3.87**	**3.87**	3.86
		4	3.83	**3.9**	3.89	3.88	3.88
$\hat{S}_z$	**0**	1	-0.5	-0.5	-0.5	-0.43	**-0.39**
		2	**-0.17**	-0.19	-0.18	-0.18	-0.19
		3	-0.09	**-0.07**	-0.09	-0.14	-0.14
		4	-0.31	-0.17	-0.16	**-0.14**	**-0.14**
$\hat{S}^2$	**0**	1	**1.52**	1.73	1.81	1.63	1.54
		2	1.48	**1.39**	1.4	1.45	1.46
		3	1.79	**1.74**	1.84	1.83	1.82
		4	1.56	1.49	1.46	**1.42**	1.44
LiH							
$\hat{O}$	$\langle \hat{O} \rangle_\Psi$	l	Adam	qBang	$F^{k=0}_{\text{block-diag}}$	qBroyden	QNG
$\hat{H}$	**-7.972**	1	-7.33	-7.35	-7.35	**-7.36**	**-7.36**
		2	-7.75	-7.81	**-7.84**	-7.82	-7.79
		3	-7.66	-7.69	**-7.72**	-7.67	-7.64
		4	-7.73	-7.77	**-7.81**	-7.77	-7.74
$\hat{N}$	**2**	1	3.0	2.9	**2.87**	2.93	2.98
		2	2.2	2.1	**2.07**	**2.07**	2.09
		3	2.8	2.71	**2.61**	2.68	2.74
		4	2.5	2.5	2.44	2.44	**2.42**
$\hat{S}_z$	**0**	1	0.1	0.05	**0.03**	0.08	0.1
		2	**-0.3**	-0.35	-0.37	-0.43	-0.44
		3	**-0.0**	-0.01	-0.05	0.07	0.13
		4	0.09	0.12	0.06	**0.05**	0.06
$\hat{S}^2$	**0**	1	1.65	1.53	1.48	1.41	**1.37**
		2	1.25	1.13	**1.02**	1.14	1.25
		3	**1.6**	1.82	1.78	1.77	1.75
		4	1.13	0.93	**0.76**	0.84	0.94

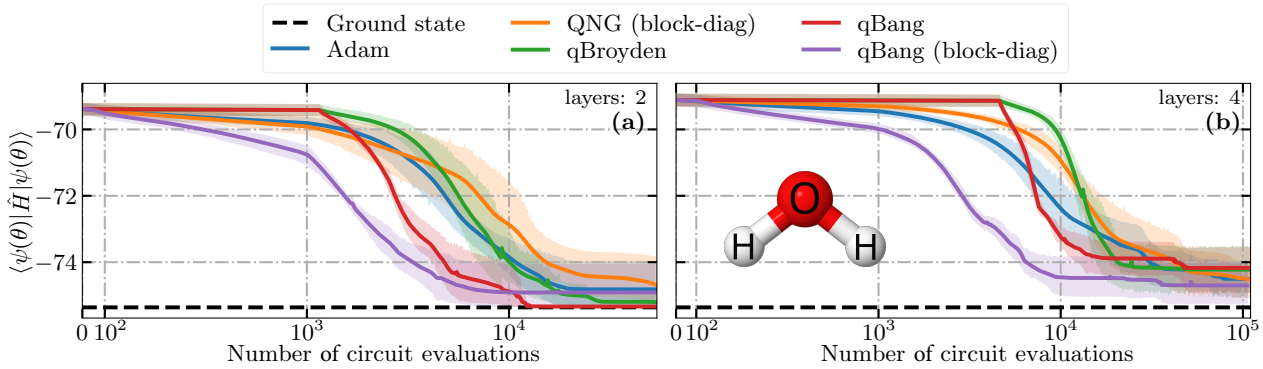

Figure 6: Comparison of optimization performance for four optimizers in finding the ground state of H_2O using a PQC. The optimizers evaluated are Adam, QNG using the block-diagonal approximation, qBroyden using the full Fisher matrix, and qBang with the full Fisher information and block-diagonal approximation. The expectation value, $\langle\psi(\boldsymbol{\theta})|H|\psi(\boldsymbol{\theta})\rangle$ is shown as a function of the number of circuit evaluations. The step size is fixed at 0.01 and the results are averaged over 5 random initializations of parameters. The PQC consists of 2 and 4 layers, as shown in subplots (a) and (b), respectively. The initial plateau in the optimization using qBroyden and qBang arises from the significant cost of initially measuring the QFIM.

optimization algorithms and optimize each circuit for 300 optimization steps. The approximation ratio, equal to one if the energy minimum is reached [see Eq. (11)], is used to evaluate the optimization performance. We show the approximation ratio plotted against the number of optimization steps for step sizes ranging from 0.01 to 0.7.

Fig. 7 demonstrates the greatest strength of Adam – its extreme resilience. Even for large step-sizes, such as 0.7, Adam remains stable and provides reliable predictions. Approximate or perturbative second order optimization methods, such as QNG and qBroyden, are prone to instabilities when using large steps. They tend to result in unreliable predictions for the local curvature which might even further amplify a large step, resulting in oscillating or divergent behaviour. Let us emphasize here that this is not a failure of second-order informed optimization but rather its approximation. Consider for example the step-reducing influence of second-order information in Newtons method for a steep harmonic potential.

Importantly, qBang can benefit from the momentum update that it inherits from Adam and achieves a resilience located between Adam and QNG/qBroyden. An even stronger resilience of qBang could be realized by unifying the gradient update with the metric update or the use of a more controlled step size depending on the local gradient and cost function, based for example on the Wolfé conditions [91]. Given the excellent performance in the previous section, we conclude that qBang is a promising optimizer that strikes the balance between low cost, high stability, speed, and accuracy.

3.5 Noise resilience

Understanding the resilience of quantum algorithms to various types of noise is crucial in the noisy intermediate-scale quantum (NISQ) era. Shot noise is one of the most fundamental contributors and arises due to the statistical nature of quantum measurements. Let us put our previous discussions in this context by considering first a simple BP circuit with 9 qubits and 6 layers, similar to the setup in Sec. 3.1. The step size is fixed at $\eta = 0.01$, and the results are averaged over 15 random initializations of parameters with 500 shots for each circuit evaluation.

Figure 8 demonstrates that all optimizers exhibit performance closely resembling that of exact state vector simulations. Among them, qBang consistently finds the solution most efficiently. We note that with shot noise, the estimate of the initial QFIM is not guaranteed to be positive semi-definite. If necessary, we ensure invertibility (and thus positive definiteness) of the initial QFIM by shifting the diagonal by the most negative eigenvalue $\lambda_{\text{min}} < 0$, as $\mathbf{F}_{\text{PD}} = \mathbf{F} + (\gamma_{\text{reg}} - \lambda_{\text{min}})\,\mathbb{1}$, see SI. Section H for details. Here, $\gamma_{\text{reg}} > 0$ is a small regularising parameter to ensure that $\mathbf{F}_{\text{PD}} \succ 0$.

Next, we revisit quantum chemistry in the form

Table 4: Converged optimization results for PQCs, representing H_2O. Results are averaged over five optimization trajectories. The ground truth for each observable is shown in the column $\langle\hat{O}\rangle_\Psi$. Observables are calculated for circuits with layers ranging from 1 to 4 based on the variational quantum state with minimum expectation value along the optimization trajectory. Bold symbols indicate the optimizer that gets closest to the ground truth. The column labeled qBang shows results by starting with the full Fisher information matrix, and the column to the right labeled $F^{k=0}_{\text{block-diag}}$ are results starting with the block-diagonal approximation.

$\hat{O}$	$\langle\hat{O}\rangle_\Psi$	l	Adam	BANG	$F^{k=0}_{\text{block-diag}}$	qBroyden	QNG
$\hat{H}$	**-75.36**	1	**-73.45**	-73.23	-73.15	-73.34	-73.36
		2	-74.82	**-75.08**	-75.02	-75.07	-75.0
		3	-73.59	-74.01	-74.04	-74.11	**-74.18**
		4	**-74.5**	-74.34	-74.46	-74.4	-74.44
$\hat{N}$	**8**	1	**7.8**	7.3	7.13	7.35	7.44
		2	7.6	**7.8**	**7.8**	7.79	7.74
		3	7.3	7.55	**7.6**	7.58	7.56
		4	7.9	**7.95**	7.87	7.75	7.76
$\hat{S}_z$	**0**	1	**-0.1**	-0.55	-0.7	-0.52	-0.48
		2	-0.2	-0.1	-0.1	-0.09	**-0.07**
		3	0.15	0.12	0.2	0.13	**0.09**
		4	0.05	**0.0**	-0.04	-0.01	0.02
$\hat{S}^2$	**0**	1	2.95	3.28	3.38	2.89	**2.86**
		2	0.5	**0.25**	0.28	0.26	0.32
		3	2.63	1.99	1.9	1.66	**1.53**
		4	1.53	1.46	1.32	**1.27**	1.29

of the H_4 circuit featuring 2 layers, discussed in Sec. 3.3.1. Circuit evaluations are performed using 500 shots and the results are averaged over 5 random initializations. We add the Simultaneous Perturbation Stochastic Approximation (SPSA) [23] optimizer, often used in a noisy circuit setting, to our comparison. All optimizers are run for 700 steps, with the exception of SPSA, which is run for 50000 steps. The step size is fixed at $\eta = 0.01$. Figure 9 illustrates how qBang outperforms Adam, while SPSA is failing to find the minimum. Surprisingly, the performance of qBang is even better when affected by noise, likely due to a slightly larger effective step when PD is enforced. Individual trajectories are presented in SI Section A.2. We can expect the improved performance of qBang to be thus of practical relevance for NISQ devices.

SPSA is a representative of a stochastic approach to optimization, closely related to random walk algorithms, and we refer the reader to Ref. [23, 92] for a detailed discussion and possible improvements. The isolated example shown here is of anecdotal evidence and does not allow to draw any conclusion about the superiority of stochastic or gradient-based approaches. We are indeed convinced that a synergistic approach could be the most promising.

4 Conclusion

Quantum computing has developed into a vibrant research domain, promising nothing less than a revolution. If this ambitious target can be met depends largely on the availability of fault-tolerant hardware and efficient algorithmic design. VQAs, merging quantum evaluations on short circuits with classical optimization of the parameterized state, are a promising framework for the use of near-term quantum computing resources. However, associated energy landscapes often feature sizeable flat areas that are challenging to maneuver. Here, we have introduced qBang and qBroyden, curvature-informed gradient-based algorithms that perform better than previous approaches for relevant quantum circuits while requiring comparably few evaluations on the QPU. The reduction in quantum evaluations is achieved by performing rank-1 updates to the Fisher information matrix. Additionally, qBang utilizes a momentum-based update rule, providing an additional boost in performance and resilience to changes in hyperparameters. We provide access to qBang and qBroyden

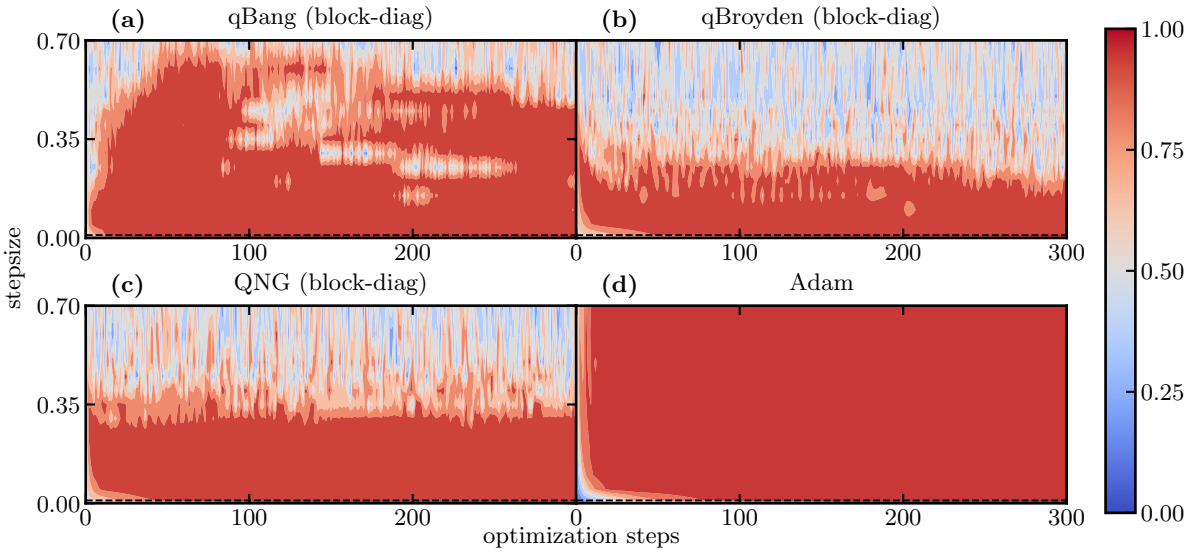

Figure 7: Dependence of convergence behavior on the learning rate by comparing the effects of different step sizes on the optimization process. Four optimization algorithms, including qBang, qBroyden, QNG with block-diagonal approximation, and Adam, are evaluated with step sizes ranging from 0.01 to 0.7. The optimization performance is assessed using the approximation ratio, which equals one if the energy minimum is reached (see Equation Eq. (11)). A dotted line at a step size of 0.01 is included to facilitate comparison with other simulations.

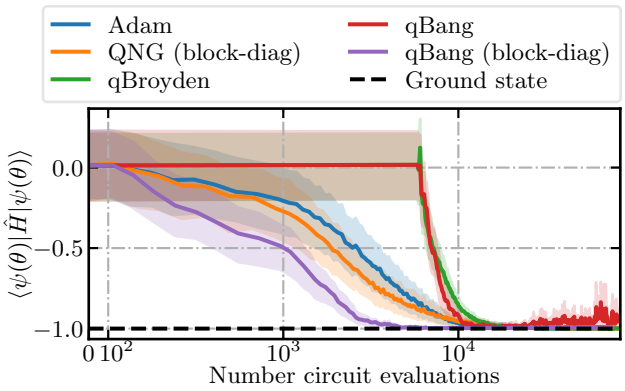

Figure 8: Comparison of optimization performance of Adam, QNG, qBroyden, and qBang in finding the ground state of the BP circuit under the influence of shot noise. $\langle\psi(\boldsymbol{\theta})|\hat{H}|\psi(\boldsymbol{\theta})\rangle$ is shown as a function of the number of circuit evaluations. The step size is fixed at $\eta = 0.01$, and the results are averaged over 15 random initializations of parameters. The PQC used consists of 6 layers. For each evaluation 500 shots are used. The initial plateau in the optimization using qBroyden and qBang arises from the significant cost of initially measuring the QFIM.

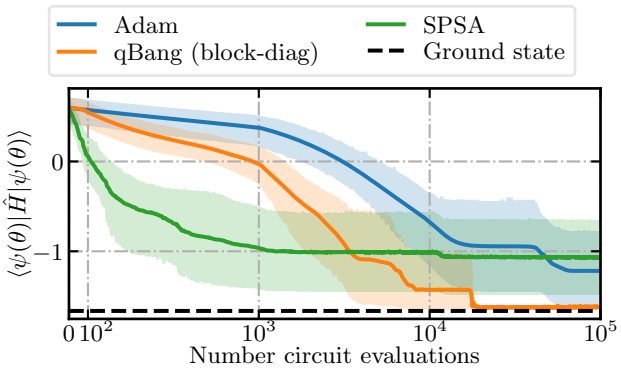

Figure 9: Comparison of optimization performance of SPSA, Adam, and qBang with the block-diagonal approximation in finding the ground state of H_4 using a 2-layer PQC. The expectation value, $\langle\psi(\boldsymbol{\theta})|\hat{H}|\psi(\boldsymbol{\theta})\rangle$ is shown as a function of the number of circuit evaluations. The step size is fixed at $\eta = 0.01$, and 500 shots are used for each evaluation. The results are averaged over 5 random initializations. Individual trajectories are presented in SI Section A.2.

via the freely accessible repository [93].

Our benchmarks, including QNG and Adam, are evaluated on a broad range of VQAs. First, we demonstrated for a set of BP circuits [18] that qBang is able to tackle flat energy landscapes efficiently. Second, we investigate classical optimization on QAOA circuits in the form of the max-cut problem, resulting in an overall underwhelming performance of all optimizers. Third, we moved on to quantum chemistry, arguably the most promising application for quantum computing. The associated VQEs have been investigated for three chemical compounds, namely H_4, LiH, and H_2O, where qBang is consistently more efficient than its competitors. Lastly, we illustrate that qBang, i.e., the combination of qBroyden and Adam, does indeed lead to a more noise- and hyper-parameter-resilient optimizer than QNG or qBroyden itself. qBang is an efficient and capable optimizer, yet the strongest aspect of our work is that it inspires a new generation of optimizers – qBang representing a first step in an evolutionary process. Such an evolution will be fostered by understanding the consequences of locality, complexity, and entanglement on the existence of BPs [94, 95].

With the increasing number of qubits and their connectivity, the number of quantum Ansatz parameters will grow, resulting in increasing pressure on the classical optimizers. With this in mind, we suggest using qBang as a "convergence starter" for optimization problems that involve a sizeable number of Ansatz layers. One potential approach is to optimize the first few layers and then keep those optimized layers with their parameters as an initial guess for the next few layers to optimize. This process can be repeated recursively until all layers are optimized and could significantly reducing the number of optimization steps required to find an acceptable ground-state energy. For a last refinement, one could use the VarQITE algorithm or restart the qBang algorithm by wiping the memory. Furthermore, the Fisher information matrix encodes information about the degree of linear dependence, i.e., it can be used to maximize the efficiency of additional layers and improve stability by controlling overparametrization [96]. To this end, it should be noted that an application to relevant problems with real-world devices remains a considerable challenge.

Acknowledgments

We thank Anton Frisk Kockum, Mats Granath, Leo Laine, Davide Castaldo, and Göran Johansson for insightful discussions. This work was supported by the Swedish Research Council (VR) through Grant No. 2016-06059 and the computational resources provided by the Swedish National Infrastructure for Computing at Chalmers Centre for Computational Science and Engineering partially funded by the Swedish Research Council through grant agreement no. 2018-05973. D.F. and R.S.J. acknowledge the Knut and Alice Wallenberg (KAW) Foundation for funding through the Wallenberg Centre for Quantum Technology (WACQT). W.D. and C.S. acknowledge funding from the Horizon Europe research and innovation program of the European Union under the Marie Skłodowska-Curie grant agreement no. 101062864 and 101065117. Partially funded by the European Union. Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union or REA. Neither the European Union nor the granting authority can be held responsible for them.

References

[1] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles. "Variational quantum algorithms". Nature Reviews Physics **3**, 625–644 (2021).

[2] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W.-K. Mok, S. Sim, L.-C. Kwek, and A. Aspuru-Guzik. "Noisy intermediate-scale quantum algorithms". Reviews of Modern Physics **94**, 015004 (2022).

[3] J. Tilly, H. Chen, S. Cao, D. Picozzi, K. Setia, Y. Li, E. Grant, L. Wossnig, I. Rungger, G. H. Booth, and J. Tennyson. "The Variational Quantum Eigensolver: A review of methods and best practices". Physics Reports **986**, 1–128 (2022).

[4] F. Arute et al. "Quantum supremacy using a programmable superconducting processor.". Nature **574**, 505–510 (2019).

[5] C. D. Bruzewicz, J. Chiaverini, R. Mc-

Connell, and J. M. Sage. "Trapped-ion quantum computing: Progress and challenges". Applied Physics Reviews **6**, 021314 (2019).

[6] A. J. Daley, I. Bloch, C. Kokail, S. Flannigan, N. Pearson, M. Troyer, and P. Zoller. "Practical quantum advantage in quantum simulation". Nature **607**, 667–676 (2022).

[7] S. Bravyi, O. Dial, J. M. Gambetta, D. Gil, and Z. Nazario. "The future of quantum computing with superconducting qubits". Journal of Applied Physics **132**, 160902 (2022).

[8] J. Preskill. "Quantum computing in the NISQ era and beyond". Quantum **2**, 79 (2018).

[9] A. Peruzzo, J. McClean, P. Shadbolt, M. H. Yung, X. Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien. "A variational eigenvalue solver on a photonic quantum processor". Nature Communications **5** (2014).

[10] D. Wecker, M. B. Hastings, and M. Troyer. "Progress towards practical quantum variational algorithms". Phys. Rev. A **92**, 042303 (2015).

[11] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik. "The theory of variational hybrid quantum-classical algorithms". New Journal of Physics **18**, 023023 (2016).

[12] S. Endo, Z. Cai, S. C. Benjamin, and X. Yuan. "Hybrid quantum-classical algorithms and quantum error mitigation". Journal of the Physical Society of Japan **90**, 032001 (2021).

[13] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization" (2017). arXiv:1412.6980.

[14] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. "Quantum circuit learning". Physical Review A **98**, 032309 (2018).

[15] L. Banchi and G. E. Crooks. "Measuring analytic gradients of general quantum evolution with the stochastic parameter shift rule". Quantum **5**, 386 (2021).

[16] M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, and N. Killoran. "Evaluating analytic gradients on quantum hardware". Physical Review A **99**, 032331 (2019).

[17] L. D'Alessio, Y. Kafri, A. Polkovnikov, and M. Rigol. "From quantum chaos and eigenstate thermalization to statistical mechanics and thermodynamics". Advances in Physics **65**, 239–362 (2016).

[18] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven. "Barren plateaus in quantum neural network training landscapes". Nature Communications **9**, 4812 (2018).

[19] Z. Holmes, K. Sharma, M. Cerezo, and P. J. Coles. "Connecting ansatz expressibility to gradient magnitudes and barren plateaus". PRX Quantum **3**, 010313 (2022).

[20] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles. "Cost function dependent barren plateaus in shallow parametrized quantum circuits". Nature Communications **12**, 1791 (2021).

[21] S. Wang, E. Fontana, M. Cerezo, K. Sharma, A. Sone, L. Cincio, and P. J. Coles. "Noise-induced barren plateaus in variational quantum algorithms". Nature Communications **12** (2021).

[22] J. Stokes, J. Izaac, N. Killoran, and G. Carleo. "Quantum Natural Gradient". Quantum **4**, 269 (2020).

[23] J. Gacon, C. Zoufal, G. Carleo, and S. Woerner. "Simultaneous perturbation stochastic approximation of the quantum Fisher information". Quantum **5**, 567 (2021).

[24] J. Liu, H. Yuan, X.-M. Lu, and X. Wang. "Quantum Fisher information matrix and multiparater estimation". Journal of Physics A: Mathematical and Theoretical **53**, 023001 (2020).

[25] D. Wierichs, C. Gogolin, and M. Kastoryano. "Avoiding local minima in variational quantum eigensolvers with the natural gradient optimizer". Physical Review Research **2**, 043246 (2020).

[26] B. Koczor and S. C. Benjamin. "Quantum natural gradient generalized to noisy and nonunitary circuits". Phys. Rev. A **106**, 062416 (2022).

[27] J. L. Beckey, M. Cerezo, A. Sone, and P. J. Coles. "Variational quantum algorithm for estimating the quantum Fisher information". Physical Review Research **4**, 013083 (2022).

[28] J. Gacon, J. Nys, R. Rossi, S. Woerner, and G. Carleo. "Variational quantum time evo-

lution without the quantum geometric tensor". Phys. Rev. Res. **6**, 013143 (2024).

[29] C. G. Broyden. "The convergence of a class of double-rank minimization algorithms 1. General considerations". IMA Journal of Applied Mathematics **6**, 76–90 (1970).

[30] M. Motta, C. Sun, A. T. K. Tan, M. J. O. Rourke, E. Ye, A. J. Minnich, F. G. S. L. Brandao, and G. K.-L. Chan. "Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution". Nature Physics **16**, 205–210 (2020).

[31] S. McArdle, T. Jones, S. Endo, Y. Li, S. C. Benjamin, and X. Yuan. "Variational ansatz-based quantum simulation of imaginary time evolution". npj Quantum Information **5**, 75 (2019).

[32] X. Yuan, S. Endo, Q. Zhao, Y. Li, and S. Benjamin. "Theory of variational quantum simulation". Quantum **3**, 191 (2019).

[33] C. Cao, Z. An, S.-Y. Hou, D. L. Zhou, and B. Zeng. "Quantum imaginary time evolution steered by reinforcement learning". Communications Physics **5**, 57 (2022).

[34] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta. "Supervised learning with quantum-enhanced feature spaces". Nature **567**, 209–212 (2019).

[35] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta. "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets". Nature **549**, 242–246 (2017).

[36] E. Farhi, J. Goldstone, and S. Gutmann. "A Quantum Approximate Optimization Algorithm" (2014). arXiv:1411.4028.

[37] S. Sim, P. D. Johnson, and A. Aspuru-Guzik. "Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms". Advanced Quantum Technologies **2**, 1900070 (2019).

[38] D. Wierichs, J. Izaac, C. Wang, and C. Y.-Y. Lin. "General parameter-shift rules for quantum gradients". Quantum **6**, 677 (2022).

[39] A. Lucas. "Ising formulations of many NP problems". Frontiers in Physics **2**, 1–14 (2014).

[40] S. Hadfield, Z. Wang, B. O'Gorman, E. G. Rieffel, D. Venturelli, and R. Biswas. "From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz". Algorithms **12**, 34 (2019).

[41] M. Svensson, M. Andersson, M. Grönkvist, P. Vikstål, D. Dubhashi, G. Ferrini, and G. Johansson. "A Heuristic Method to solve large-scale Integer Linear Programs by combining Branch-and-Price with a Quantum Algorithm" (2021). arXiv:2103.15433.

[42] W. Lavrijsen, A. Tudor, J. Müller, C. Iancu, and W. de Jong. "Classical optimizers for noisy intermediate-scale quantum devices". In 2020 IEEE International Conference on Quantum Computing and Engineering (QCE). Pages 267–277. (2020).

[43] Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. D. Sawaya, S. Sim, L. Veis, and A. Aspuru-Guzik. "Quantum chemistry in the age of quantum computing". Chemical Reviews **119**, 10856–10915 (2019).

[44] V. Lordi and J. M. Nichol. "Advances and opportunities in materials science for scalable quantum computing". MRS Bulletin **46**, 589–595 (2021).

[45] G. E. Crooks. "Gradients of parameterized quantum gates using the parameter-shift rule and gate decomposition" (2019). quant-ph:1905.13311.

[46] J. Martens. "New insights and perspectives on the natural gradient method". Journal of Machine Learning Research **21**, 1–76 (2020). url: `https://www.jmlr.org/papers/v21/17-678.html`.

[47] J. Martens and I. Sutskever. "Training deep and recurrent networks with Hessian-free optimization". Pages 479–535. Springer Berlin Heidelberg. (2012).

[48] D. F. Shanno. "Conditioning of quasi-Newton methods for function minimization". Mathematics of Computation **24**, 647–656 (1970).

[49] R. Fletcher. "A new approach to variable metric algorithms". The Computer Journal **13**, 317–322 (1970).

[50] D. Goldfarb. "A family of variable-metric methods derived by variational means". Mathematics of Computation **24**, 23–26 (1970).

[51] S. Ruder. "An overview of gradient descent optimization algorithms" (2016). arXiv:1609.04747.

[52] G. C. Wick. "Properties of Bethe-Salpeter wave functions". Phys. Rev. **96**, 1124–1134 (1954).

[53] T. Tsuchimochi, Y. Ryo, S. L. Ten-no, and K. Sasasako. "Improved algorithms of quantum imaginary time evolution for ground and excited states of molecular systems". Journal of Chemical Theory and Computation (2023).

[54] W. von der Linden. "A quantum Monte Carlo approach to many-body physics". Physics Reports **220**, 53–162 (1992).

[55] D. M. Ceperley. "Path integrals in the theory of condensed helium". Rev. Mod. Phys. **67**, 279–355 (1995).

[56] N. Trivedi and D. M. Ceperley. "Ground-state correlations of quantum antiferromagnets: A Green-function Monte Carlo study". Phys. Rev. B **41**, 4552–4569 (1990).

[57] K. Guther, R. J. Anderson, N. S. Blunt, N. A. Bogdanov, D. Cleland, N. Dattani, W. Dobrautz, K. Ghanem, P. Jeszenszki, N. Liebermann, et al. "NECI: N-Electron Configuration Interaction with an emphasis on state-of-the-art stochastic methods". The Journal of Chemical Physics **153**, 034107 (2020).

[58] A. McLachlan. "A variational solution of the time-dependent Schrodinger equation". Molecular Physics **8**, 39–44 (1964).

[59] C. Zoufal, D. Sutter, and S. Woerner. "Error bounds for variational quantum time evolution". Phys. Rev. Appl. **20**, 044059 (2023).

[60] G. Fubini. "Sulla teoria delle funzioni automorfe e delle loro trasformazioni". Annali di Matematica Pura ed Applicata **14**, 33–67 (1908).

[61] E. Study. "Kürzeste wege im komplexen gebiet". Mathematische Annalen **60**, 321–378 (1905).

[62] Y. Yao, P. Cussenot, R. A. Wolf, and F. Miatto. "Complex natural gradient optimization for optical quantum circuit design". Phys. Rev. A **105**, 052402 (2022).

[63] F. Wilczek and A. Shapere. "Geometric phases in physics". World Scientific Publishing. (1989).

[64] L. Hackl, T. Guaita, T. Shi, J. Haegeman, E. Demler, and J. I. Cirac. "Geometry of variational methods: dynamics of closed quantum systems". SciPost Phys. **9**, 048 (2020).

[65] S. Zhou and L. Jiang. "An exact correspondence between the quantum Fisher information and the Bures metric" (2019). arXiv:1910.08473.

[66] V. Giovannetti, S. Lloyd, and L. Maccone. "Advances in quantum metrology". Nature Photonics **5**, 222–229 (2011).

[67] D. Petz and C. Sudár. "Geometries of quantum states". Journal of Mathematical Physics **37**, 2662–2673 (1996).

[68] J. P. Provost and G. Vallee. "Riemannian structure on manifolds of quantum states". Communications in Mathematical Physics **76**, 289–301 (1980).

[69] C.-Y. Park and M. J. Kastoryano. "Geometry of learning neural quantum states". Physical Review Research **2**, 023232 (2020).

[70] S. L. Braunstein and C. M. Caves. "Statistical distance and the geometry of quantum states". Phys. Rev. Lett. **72**, 3439–3443 (1994).

[71] P. Facchi, R. Kulkarni, V. Man'ko, G. Marmo, E. Sudarshan, and F. Ventriglia. "Classical and quantum Fisher information in the geometrical formulation of quantum mechanics". Physics Letters A **374**, 4801–4803 (2010).

[72] S.-I. Amari. "Neural learning in structured parameter spaces: natural Riemannian gradient". In Proceedings of the 9th International Conference on Neural Information Processing Systems. Pages 127—133. NIPS'96. MIT Press (1996).

[73] S.-i. Amari. "Natural gradient works efficiently in learning". Neural Computation **10**, 251–276 (1998).

[74] S.-i. Amari and S. Douglas. "Why natural gradient?". In Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP

'98 (Cat. No.98CH36181). Volume 2, pages 1213–1216. (1998).

[75] S.-i. Amari, H. Park, and K. Fukumizu. "Adaptive method of realizing natural gradient learning for multilayer perceptrons". Neural Computation **12**, 1399–1409 (2000).

[76] J. J. Meyer. "Fisher information in noisy intermediate-scale quantum applications". Quantum **5**, 539 (2021).

[77] P. Huembeli and A. Dauphin. "Characterizing the loss landscape of variational quantum circuits". Quantum Science and Technology **6**, 025011 (2021).

[78] E. Grant, L. Wossnig, M. Ostaszewski, and M. Benedetti. "An initialization strategy for addressing barren plateaus in parametrized quantum circuits". Quantum **3**, 214 (2019).

[79] I. O. Sokolov, W. Dobrautz, H. Luo, A. Alavi, and I. Tavernelli. "Orders of magnitude increased accuracy for quantum many-body problems on quantum computers via an exact transcorrelated method". Phys. Rev. Res. **5**, 023174 (2023).

[80] W. Dobrautz, I. O. Sokolov, K. Liao, P. L. Ríos, M. Rahm, A. Alavi, and I. Tavernelli. "Ab initio transcorrelated method enabling accurate quantum chemistry on near-term quantum hardware" (2023). arXiv:2303.02007.

[81] T. R. Bromley, J. M. Arrazola, S. Jahangiri, J. Izaac, N. Quesada, A. D. Gran, M. Schuld, J. Swinarton, Z. Zabaneh, and N. Killoran. "Applications of near-term photonic quantum computers: software and algorithms". Quantum Science and Technology **5**, 034010 (2020).

[82] H. Park, S.-i. Amari, and K. Fukumizu. "Adaptive natural gradient learning algorithms for various stochastic models". Neural Networks **13**, 755—-764 (2000).

[83] S.-i. Amari. "Information geometry and its applications". Springer. (2016).

[84] S. Dash, F. Vicentini, M. Ferrero, and A. Georges. "Efficiency of neural quantum states in light of the quantum geometric tensor" (2024). arXiv:2402.01565.

[85] D. Fitzek, R. S. Jonsson, W. Dobrautz, and C. Schäfer (2023). code: davidfitzek/qflow.

[86] B. van Straaten and B. Koczor. "Measurement cost of metric-aware variational quantum algorithms". PRX Quantum **2**, 030324 (2021).

[87] A. N. Tikhonov, A. V. Goncharsky, V. V. Stepanov, and A. G. Yagola. "Numerical methods for the solution of ill-posed problems". Springer Dordrecht. (1995).

[88] V. Bergholm, J. Izaac, M. Schuld, et al. "PennyLane: Automatic differentiation of hybrid quantum-classical computations" (2018). arXiv:1811.04968.

[89] T. Helgaker, P. Jørgensen, and J. Olsen. "Molecular electronic-structure theory". John Wiley & Sons. (2000).

[90] Q. Sun, X. Zhang, S. Banerjee, P. Bao, et al. "Recent developments in the PySCF program package". The Journal of Chemical Physics **153**, 024109 (2020).

[91] J. Nocedal and S. J. Wright. "Numerical optimization". Springer Science+Business Media. (2006).

[92] J. M. Kübler, A. Arrasmith, L. Cincio, and P. J. Coles. "An Adaptive Optimizer for Measurement-Frugal Variational Algorithms". Quantum **4**, 263 (2020).

[93] D. Fitzek, R. S. Jonsson, W. Dobrautz, and C. Schäfer (2023). code: davidfitzek/qbang.

[94] M. Ragone, B. N. Bakalov, F. Sauvage, A. F. Kemper, C. O. Marrero, M. Larocca, and M. Cerezo. "A unified theory of barren plateaus for deep parametrized quantum circuits" (2023). arXiv:2309.09342.

[95] E. Fontana, D. Herman, S. Chakrabarti, N. Kumar, R. Yalovetzky, J. Heredge, S. H. Sureshbabu, and M. Pistoia. "The adjoint is all you need: Characterizing barren plateaus in quantum ansätze" (2023). arXiv:2309.07902.

[96] M. Larocca, N. Ju, D. García-Martín, P. J. Coles, and M. Cerezo. "Theory of overparametrization in quantum neural networks". Nature Computational Science **3**, 542–551 (2023).

[97] Y. Du, M.-H. Hsieh, T. Liu, and D. Tao. "Expressive power of parametrized quantum circuits". Phys. Rev. Res. **2**, 033125 (2020).

[98] L. Funcke, T. Hartung, K. Jansen, S. Kühn, and P. Stornati. "Dimensional expressivity analysis of parametric quantum circuits". Quantum **5**, 422 (2021).

[99] Y. Du, Z. Tu, X. Yuan, and D. Tao. "Ef-

ficient measure for the expressivity of variational quantum algorithms". Phys. Rev. Lett. **128**, 080506 (2022).

[100] R. D'Cunha, T. D. Crawford, M. Motta, and J. E. Rice. "Challenges in the use of quantum computing hardware-efficient ansätze in electronic structure theory". The Journal of Physical Chemistry A (2023).

[101] H. Shima. "The geometry of Hessian structures". World Scientific. (2007).

[102] L. Campos Venuti and P. Zanardi. "Quantum critical scaling of the geometric tensors". Phys. Rev. Lett. **99**, 095701 (2007).

[103] M. Bukov, D. Sels, and A. Polkovnikov. "Geometric speed limit of accessible many-body state preparation". Phys. Rev. X **9**, 011034 (2019).

[104] M. Kolodrubetz, D. Sels, P. Mehta, and A. Polkovnikov. "Geometry and non-adiabatic response in quantum and classical systems". Physics Reports **697**, 1–87 (2017).

[105] S. Pancharatnam. "Generalized theory of interference, and its applications". Proceedings of the Indian Academy of Sciences - Section A **44**, 247–262 (1956).

[106] M. V. Berry. "Quantal phase factors accompanying adiabatic changes". Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences **392**, 45–57 (1984).

[107] J. Broeckhove, L. Lathouwers, E. Kesteloot, and P. V. Leuven. "On the equivalence of time-dependent variational principles". Chemical Physics Letters **149**, 547–550 (1988).

[108] S. Sorella. "Green function Monte Carlo with stochastic reconfiguration". Phys. Rev. Lett. **80**, 4558–4561 (1998).

[109] S. Sorella and L. Capriotti. "Green function Monte Carlo with stochastic reconfiguration: An effective remedy for the sign problem". Phys. Rev. B **61**, 2599–2612 (2000).

[110] G. Mazzola, A. Zen, and S. Sorella. "Finite-temperature electronic simulations without the Born-Oppenheimer constraint". The Journal of Chemical Physics **137**, 134112 (2012).

Appendix

A Single trajectories including QITE

In this section, we compare the performance of qBang, qBroyden, QNG, and Adam optimizers, including QNG using the full quantum Fisher information matrix (QFIM) at each step. We consider a barren plateau (BP) circuit with 4 layers and 9 qubits, resulting in 36 tunable parameters. We optimize for 700 steps, resulting in varying circuit evaluations since the QFIM requires n_θ^2 circuit evaluations while approximations such as diagonal or block-diagonal approximation require only $n_\theta + l$ circuit evaluations, where n_θ is the number of variational parameters and l is the number of layers in the circuit. QNG using the QFIM is equivalent, up to a constant factor, to VarQITE [31]. QNG, qBang, and qBroyden require the QFIM in the first step, explaining the initial plateau in the number of circuit evaluations compared to Adam or the approximated versions. All optimizers, except for QNG with the block-diagonal approximation, converge to the exact ground state solution. The results in Fig. 10 show that a single estimate of the QFIM, in combination with an appropriate cost-efficient metric update, is sufficient to speed up convergence to the desired ground state.

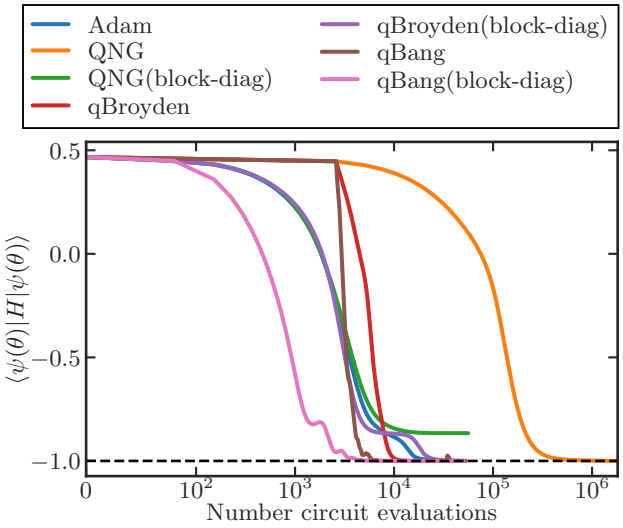

Figure 10: Comparison of optimization performance of Adam, QNG, qBroyden, and qBang in finding the ground state of the BP circuit. $\langle\psi(\boldsymbol{\theta})|\hat{H}|\psi(\boldsymbol{\theta})\rangle$ is shown as a function of the number of circuit evaluations. The step size is fixed at $\eta = 0.01$. The PQCs used consist of 4 layers. All optimizers perform 700 steps, which results in a wide range of circuit evaluations due to the expensive estimation of the Fisher information. The initial plateau in the optimization using QNG, qBroyden and qBang arises from the significant cost of initially measuring the QFIM.

A.1 Why updating the metric is important (ablation study)

In this subsection, we perform an ablation study to investigate the impact of the update rule formula on optimization performance. We use a BP circuit with 9 qubits and 6 layers and average over 10 random parameter initializations.

We show in Fig. 11 that, for the first iterations, both algorithms perform similarly, but in the long run, without a metric update, oscillations appear in the system, leading to no convergence of the optimization. To understand this behavior, let us recall that the Fisher information is a measure of how much a parametrized state changes under a change of a parameter [76]. This information can be understood as an adaptive step size for each parameter to optimize. However, since the energy landscape changes during optimization, we need to adjust the Fisher information to ensure proper

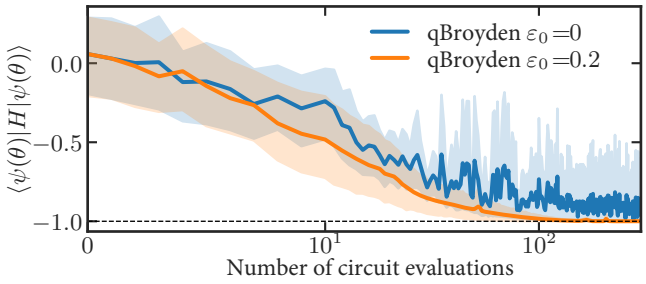

Figure 11: Effect of metric update on the optimization performance in a 6-layer, 9-qubit BP circuit. The performance of qBroyden is compared for $\varepsilon_0 = 0$ and $\varepsilon_0 = 0.2$. When $\varepsilon_0 = 0$, the update rule Eq. (10) is not used. For both settings, the algorithms are initialized with the full Fisher information matrix. Results are averaged over 10 random parameter initializations with 300 optimization steps each.

convergence. As shown in Figure 11, if we do not correct the metric, oscillations start after a few optimization steps when the energy landscape has undergone a sufficient change and is no longer described by the initial QFIM. On the other hand, the quasi-Newton updates to the initial metric ensure that the gradient descent is more consistent and qBroyden find the ground state quickly.

The update rule is thus crucial and provides the necessary correction to adjust the curvature of the Fisher information matrix based on the current point in the energy landscape. This has two significant advantages. First, it reduces the number of circuit queries required, and second, it simplifies the algorithm's execution on the hardware because we only need to estimate the Fisher information once on the quantum device.

In summary, the ablation study in Fig. 11 shows that correcting the metric is essential to avoid oscillations and ensure convergence of the optimization process.

A.2 Analysis of H_4 optimization trajectories under shot noise

Revisiting the H_4 circuit with 2 layers, as discussed in Section 3.5 of the main document, we now shift our focus from averaged results to an examination of individual optimization trajectories. This approach provides a more granular view of the optimizer performance under shot noise conditions. Each circuit evaluation is performed using 500 shots, and we observe the behavior across 5 random initializations.

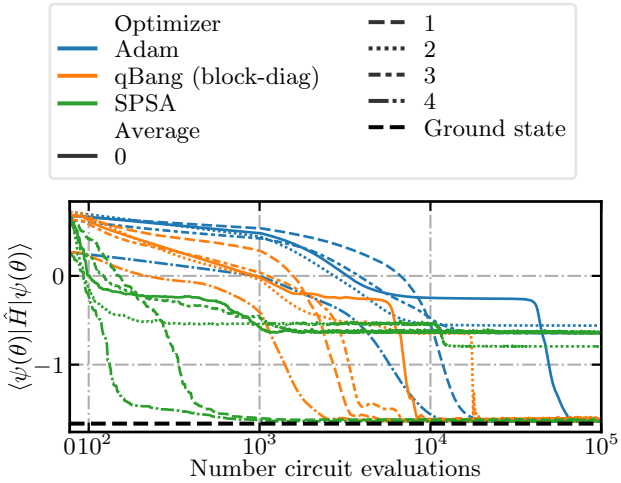

Figure 12: Individual optimization trajectories for the H_4 circuit with a 2-layer PQC. The expectation value, $\langle\psi(\boldsymbol{\theta})|\hat{H}|\psi(\boldsymbol{\theta})\rangle$, is shown for each circuit evaluation. The step size is fixed at $\eta = 0.01$. Each line represents a separate optimization run, illustrating the variability among trajectories.

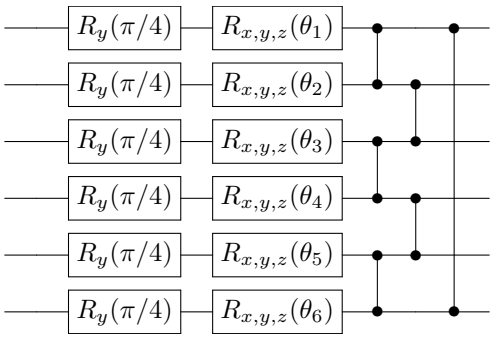

Figure 13: The BP circuit ansatz. The ansatz consists of an initial layer of $R_y(\pi/4)$ gates followed by l layers of parameterized Pauli rotations and a controlled-Z entangling layer, initialized in the state $|0\rangle^n$ for all n qubit registers.

In the analysis, represented in Fig. 12, qBang demonstrates reliable performance in finding the ground state and outperforms the Adam and Simultaneous Perturbation Stochastic Approximation (SPSA) optimizer. Notably, SPSA, despite running for 50000 steps, struggles to locate the minimum in several cases.

B Circuit layouts and Hamiltonians

This section collects all the circuit ansätze and Hamiltonian descriptions used for the benchmarks. All of the circuits are built with l layers. The more layers the larger the expressivity of the circuit which allows for potentially more accurate solutions but also increases the linear dependence of parameters. All circuits are optimized in a closed-loop with a classical optimization algorithm to minimize $\langle\psi(\boldsymbol{\theta})|\,\hat{H}\,|\psi(\boldsymbol{\theta})\rangle$, where $\psi(\boldsymbol{\theta})$ describes the circuit ansatz.

B.1 Barren plateau circuit

BPs are a major obstacle in quantum computing, hindering its potential for solving complex problems [11, 20]. The BP circuit is an example of this phenomenon and utilizes the objective operator $\hat{H} = \hat{Z}_1\hat{Z}_2$ with a ground state energy of -1. The circuit is initialized in the state $|0\rangle^n$ and consists of an initial fixed layer of $R_y(\pi/4)$ gates acting on n qubits, followed by l layers of parameterized Pauli rotations with an entangling layer of controlled-Z gates, as shown in Figure 13. This circuit is a critical benchmark for understanding and addressing the BP problem in quantum computing.

B.2 Quantum approximate optimization algorithm circuit ansatz

The Quantum Approximate Optimization Algorithm (QAOA) is a quantum algorithm that can be used to solve combinatorial optimization problems. One such problem is the max-cut problem, which involves partitioning a set of vertices in a graph into two disjoint subsets such that the number of edges between the subsets is maximized [36].

The max-cut problem is mapped onto a quantum optimization problem by constructing a cost Hamiltonian $\hat{H}_C$ that encodes the objective function of the max-cut problem. The cost Hamiltonian is defined as follows:

$$\hat{H}_C = \sum_{(i,j)\in E} \frac{1}{2}(\hat{\mathbb{1}} - \hat{Z}_i\hat{Z}_j), \tag{12}$$

where E is the set of edges in the graph, and Z_i and Z_j are the Pauli Z operators acting on the qubits corresponding to vertices i and j, respectively. The cost Hamiltonian penalizes states in which neighboring vertices are in the same subsets since the corresponding edge contributes 1 to the energy in these states.

The quantum circuit uses two non-commuting operators, the cost Hamiltonian and the mixing Hamiltonian, to evolve the system towards states that optimize the cost function. The mixing Hamiltonian

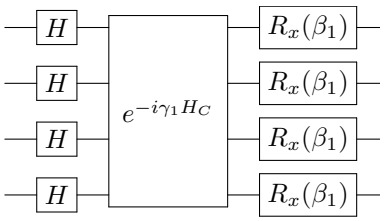

Figure 14: The QAOA circuit ansatz. It is composed of alternating layers of the cost Hamiltonian and the mixing Hamiltonian. The circuit is initialized in the state $|0\rangle^n$, where n is the number of qubits required by the cost Hamiltonian. The parameters of the circuit are optimized to maximize the expected value of the cost function.

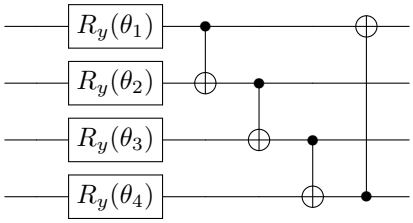

Figure 15: The hardware efficient circuit ansatz is composed of l layers of parametrized single qubit R_y rotations and a ring of CNOT gates to entangle the qubits. The circuit is applied to n qubits, with the parameters optimized to minimize the energy of the molecular system.

is typically a sum of Pauli X operators, acting as a "driver" that moves the system away from the initial state and encourages exploration of different states.

Figure 14 shows a QAOA circuit ansatz with one layer, applying the cost and mixing Hamiltonians. The circuit is initialized in the state $|0\rangle^n$, which is transformed into the uniform superposition state $|+\rangle^n$ via the Hadamard gate. The QAOA provides an approximation to the optimal solution, with the quality of the approximation expected to improve as the number of layers l is increased.

B.3 Chemistry applications

We employed minimal basis sets (STO-6G) for all quantum chemistry problems and used a frozen core approximation for LiH and H_2O (with the 1s orbital of Li and O, respectively, frozen) [89]. We used a hardware-efficient ansatz (HEA) [35] that utilizes 8, 10, and 12 qubits for H_4, LiH, and H_2O, respectively. This ansatz is composed of l layers, each comprising a tunable $R_y(\theta)$ gate on each qubit register, followed by a closed ring of CNOT gates. A 1-layer motif of the HEA for 4 qubits can be seen in Fig. 15. In the following, we list the geometries of all the studied molecular problems (in the xyz-format and atomic units):

Listing 1: H_4 geometry in xyz-format and atomic units

```
4
*
H       2.1213      2.1213      0.0
H       2.1213     -2.1213      0.0
H      -2.1213      2.1213      0.0
H      -2.1213     -2.1213      0.0
```

Listing 2: LiH geometry in xyz-format and atomic units

```
2
*
Li      0.0     0.0     0.0
H       0.0     0.0     3.0
```

Listing 3: H_2O geometry in xyz-format and atomic units

```
3
*
O    0.0         0.0         0.0
H    0.8081      1.0437      0.0
H    0.8081     -1.0437      0.0
```

We provide a python implementation of the circuits and Hamiltonians used in this work in [85].

B.3.1 Hardware-efficient R_y Ansatz

HEAs, like the R_y Ansatz shown in Fig. 15, are commonly used in quantum computing studies of chemical and physical systems. It is, however, not trivial and thus an active field of research how increasing the number of layers affects the "expressivity" – how well $|\psi(\boldsymbol{\theta})\rangle$ can approximate the target $|\Psi\rangle$ – of a HEA [37, 97, 98, 99, 100]. This effect can be seen in the slow convergence of the total energy of H_4 with the number of ansatz layers, see Fig. 15. Nevertheless, we chose to study HEA in this work since (a) they are desirable to use as they lead to smaller errors due to hardware noise [35]. However, especially because it was proven that the gradient exponentially vanishes for deep, randomly initialized HEA [18, 19].

C Initialization using Hartree-Fock parameters

We present the performance starting from the Hartree-Fock parameter initialization in Fig. 16. We compare the optimization performance of four different optimizers, namely Adam, Quantum natural gradient (QNG) with block-diag approximation, qBroyden with full QFIM and qBang with block-diag and full QFIM, for finding the ground state of H_4 using a variational quantum circuit. The step size for each optimizer is set to 0.01. We employ a parameterized quantum circuit (PQC) with varying numbers of layers, from 1 to 4, to explore the impact of circuit depth on the optimization performance of each optimizer. To ensure the robustness of our results, we perform 15 independent optimization runs, each with a randomly perturbed Hartree-Fock parameter initialization. Overall we see stable convergence behavior for the chosen circuit ansatz. All optimizers converge to the same minimum.

D Collection of Algorithms

This section summarizes all the optimization algorithms introduced in this work.

qBroyden is a quasi-Newton method that approximates the QFIM matrix using rank-one updates. In each iteration, the inverse QFIM is updated using an updating rule that depends on the gradient and parameter differences between the current and previous iterations. Algorithm 2 presents the pseudo-code for qBroyden.

qBang is an extension of qBroyden that incorporates both the approximation of the QFIM and momentum. In each iteration, the gradients are first normalized using the adaptive moment estimation (Adam) method, and then a preconditioned gradient step is taken using the inverse QFIM. Similar to qBroyden, qBang can also incorporate QNG, QFIM, or the identity matrix as a preconditioner. Algorithm 1 presents the pseudo-code for qBang.

Momentum QNG combines momentum optimization with QNG. In each iteration, we utilize an Adam [13] inspired update for the momentum and then take a natural gradient step by using both the momentum and the QNG approximation of the QFIM. Algorithm 5 presents the pseudo-code for Momentum QNG.

A Python implementation for all three optimization algorithms can be found in [93].

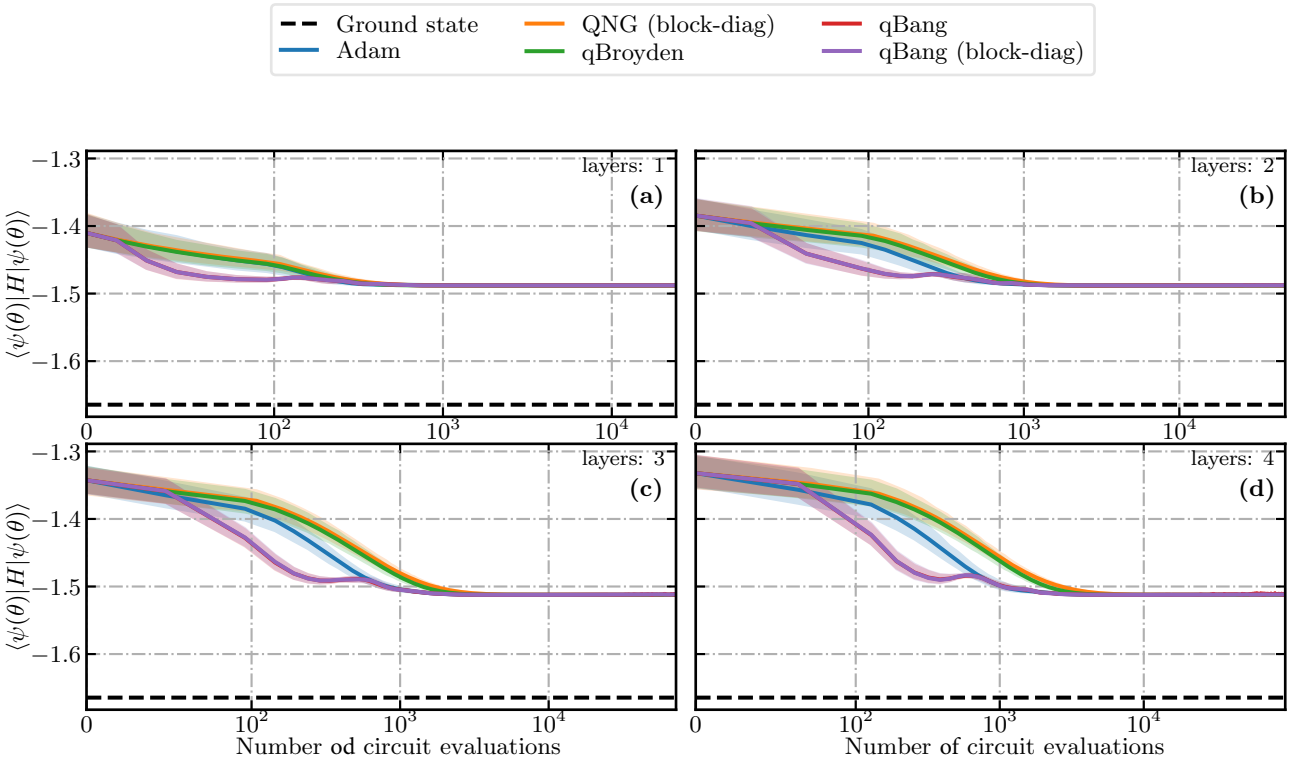

Figure 16: Comparing the optimization performance for the four optimizers, Adam, Quantum natural gradient (QNG) with block-diag approximation, qBroyden with full QFIM and qBang with block-diag and full QFIM for finding the ground state of H_4 with a variational quantum circuit. The step size is set to 0.01. We average over 15 randomly perturbed HF-parameter initializations. We use a PQC with 1, 2, 3, and 4 layers.

Algorithm 3 qBroyden

Input: learning rates $\eta = 0.01$, $\varepsilon_0 = 0.2$
Input: convergence criterion γ
Input: PQC $U(\boldsymbol{\theta})$ with initial parameter vector $\boldsymbol{\theta}_0 \in \mathbb{R}^p$.
Initialization: $k \leftarrow 0$, $\mathbf{B}_0^{-1}$ via QNG, QFIM or Identity
`not_converged`$\leftarrow$`true`
while `not_converged` **do**
    QC: estimate $\mathcal{L}(\boldsymbol{\theta}_k)$
    QC: estimate $\nabla\mathcal{L}(\boldsymbol{\theta}_k)$
    if $\|\mathbf{B}_k^{-1}\nabla\mathcal{L}(\boldsymbol{\theta}_k)\|_2 > \gamma$ **then**
        $\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k - \eta\,\mathbf{B}_k^{-1}\nabla\mathcal{L}(\boldsymbol{\theta}_k)$
        $\varepsilon_k \leftarrow \frac{\varepsilon_0}{k+1}$
        $\mathbf{B}_{k+1}^{-1} \leftarrow$ Eq. (10)
        $k \leftarrow k+1$
    else
        `not_converged`$\leftarrow$`false`
        $\boldsymbol{\theta}^* \leftarrow \underset{\{\boldsymbol{\theta}_n\}_0^k}{\operatorname{argmin}}\,\mathcal{L}(\boldsymbol{\theta}_n)$
    end if
end while
return $\boldsymbol{\theta}^*$

Algorithm 4 qBang

Input: learning rates $\eta = 0.01$, $\varepsilon_0 = 0.2$
Input: decay rates $\beta_1 = 0.9$ and $\beta_2 = 0.999$
Input: convergence criterion γ
Input: PQC $U(\boldsymbol{\theta})$ with initial parameter vector $\boldsymbol{\theta}_0 \in \mathbb{R}^p$.
Initialization: $k \leftarrow 0$, $\boldsymbol{m}_{-1} \leftarrow \mathbf{0}$, $\boldsymbol{v}_{-1} \leftarrow \mathbf{0}$, $\mathbf{B}_0^{-1}$ via QNG, QFIM or Identity
`not_converged`$\leftarrow$`true`
while `not_converged` **do**
    QC: estimate $\mathcal{L}(\boldsymbol{\theta}_k)$
    QC: estimate $\nabla\mathcal{L}(\boldsymbol{\theta}_k)$
    $\boldsymbol{m}_k \leftarrow \beta_1 \boldsymbol{m}_{k-1} + (1-\beta_1)\,\nabla\mathcal{L}(\boldsymbol{\theta}_k)$
    $\boldsymbol{v}_k \leftarrow \beta_2 \boldsymbol{v}_{k-1} + (1-\beta_2)\,\nabla\mathcal{L}(\boldsymbol{\theta}_k) \odot \nabla\mathcal{L}(\boldsymbol{\theta}_k)$
    $\widehat{\boldsymbol{m}}_k \leftarrow \boldsymbol{m}_k / \left(1-\beta_1^{k+1}\right)$
    $\widehat{\boldsymbol{v}}_k \leftarrow \boldsymbol{v}_k / \left(1-\beta_2^{k+1}\right)$
    $\{\boldsymbol{p}_k\}_l \leftarrow \{\widehat{\boldsymbol{m}}_k\}_l / \left(\sqrt{\{\widehat{\boldsymbol{v}}_k\}_l} + \kappa\right),\ \forall l \in \{1, 2, \ldots, p\}$
    if $\|\mathbf{B}_k^{-1}\boldsymbol{p}_k\|_2 > \gamma$ **then**
        $\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k - \eta\,\mathbf{B}_k^{-1}\boldsymbol{p}_k$
        $\varepsilon_k \leftarrow \frac{\varepsilon_0}{k+1}$
        $\mathbf{B}_{k+1}^{-1} \leftarrow$ Eq. (10)
        $k \leftarrow k+1$
    else
        `not_converged`$\leftarrow$`false`
        $\boldsymbol{\theta}^* \leftarrow \underset{\{\boldsymbol{\theta}_n\}_0^k}{\operatorname{argmin}}\,\mathcal{L}(\boldsymbol{\theta}_n)$
    end if
end while
return $\boldsymbol{\theta}^*$

Algorithm 5 Momentum QNG

Input: learning rates $\eta = 0.01$, $\varepsilon_0 = 0.2$
Input: decay rates $\beta_1 = 0.9$ and $\beta_2 = 0.999$
Input: convergence criterion γ
Input: PQC $U(\boldsymbol{\theta})$ with initial parameter vector $\boldsymbol{\theta}_0 \in \mathbb{R}^p$.
Initialization: $k \leftarrow 0$, $\boldsymbol{m}_{-1} \leftarrow \mathbf{0}$, $\boldsymbol{v}_{-1} \leftarrow \mathbf{0}$
`not_converged`$\leftarrow$`true`
while `not_converged` **do**
  QC: estimate $\mathcal{L}(\boldsymbol{\theta}_k)$
  QC: estimate $\nabla\mathcal{L}(\boldsymbol{\theta}_k)$
  QC: estimate $\mathbf{B}_k$
  $\boldsymbol{m}_k \leftarrow \beta_1 \boldsymbol{m}_{k-1} + (1-\beta_1)\nabla\mathcal{L}(\boldsymbol{\theta}_k)$
  $\boldsymbol{v}_k \leftarrow \beta_2 \boldsymbol{v}_{k-1} + (1-\beta_2)\nabla\mathcal{L}(\boldsymbol{\theta}_k) \odot \nabla\mathcal{L}(\boldsymbol{\theta}_k)$
  $\widehat{\boldsymbol{m}}_k \leftarrow \boldsymbol{m}_k / \left(1-\beta_1^{k+1}\right)$
  $\widehat{\boldsymbol{v}}_k \leftarrow \boldsymbol{v}_k / \left(1-\beta_2^{k+1}\right)$
  $\{\boldsymbol{p}_k\}_l \leftarrow \{\widehat{\boldsymbol{m}}_k\}_l / \left(\sqrt{\{\widehat{\boldsymbol{v}}_k\}_l} + \kappa\right),\ \forall l \in \{1, 2, \ldots, p\}$
  if $\|\mathbf{B}_k^{-1}\boldsymbol{p}_k\|_2 > \gamma$ **then**
    $\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k - \eta\, \mathbf{B}_k^{-1}\boldsymbol{p}_k$
    $k \leftarrow k+1$
  else
    `not_converged`$\leftarrow$`false`
    $\boldsymbol{\theta}^* \leftarrow \underset{\{\boldsymbol{\theta}_n\}_0^k}{\operatorname{argmin}} \mathcal{L}(\boldsymbol{\theta}_n)$
  end if
end while
return $\boldsymbol{\theta}^*$

E Relation between Fisher information and Hessian

For certain classes of classical optimization problems, the natural gradient method is equivalent to the Newton method. Here, we describe a class of problems where the Fisher information matrix (FIM) and Hessian are related. This relationship is well known in the literature, see, e.g., Ref. [101].

Let the random variable $X \in \mathcal{D}_X$ be distributed according to the probability density function $p(X;\boldsymbol{\theta})$, where the distribution is parametrized by the continuous parameter vector $\boldsymbol{\theta}$. Through the Cramér-Rao lower bound, the FIM describes how well $\boldsymbol{\theta}$ can be estimated, ideally, from observations of X. The FIM is defined as

$$\mathbf{I}_{i,j} = \mathrm{E}_X\left[\left(\partial_{\theta_i} \ln p(X;\boldsymbol{\theta})\right)\left(\partial_{\theta_j} \ln p(X;\boldsymbol{\theta})\right)\right], \tag{13}$$

or $\mathbf{I}_{i,j} = \int_{\mathcal{D}_X} \mathrm{d}X\, \frac{(\partial_{\theta_i} p(X;\boldsymbol{\theta}))(\partial_{\theta_j} p(X;\boldsymbol{\theta}))}{p(X;\boldsymbol{\theta})}$. A required condition of regularity permits us to exchange the order of integration and differentiation[1] and the FIM can then be described with the second order derivatives, as $\mathbf{I}_{i,j} = -\mathrm{E}_X\left[\frac{\partial^2}{\partial_{\theta_i}\partial_{\theta_j}} \ln p(X;\boldsymbol{\theta})\right]$.

Let us assume we are dealing with a stochastic optimization problem, where the task is to minimize some loss function L. That is, we want to minimize the expectation over X of some parametrized error function $l(X;\boldsymbol{\theta})$, as $L = \mathrm{E}_X[l(X;\boldsymbol{\theta})]$. Newton-based optimization involves the Hessian of L, which has

[1] Specifically, the required condition is that $\int_{\mathcal{D}_x} \mathrm{d}X \partial_{\theta_i}\partial_{\theta_j} p(X;\boldsymbol{\theta}) = 0$, which is satisfied if $\mathcal{D}_X$ is independent of $\boldsymbol{\theta}$.

elements $\mathbf{H}_{i,j} = \frac{\partial^2}{\partial_{\theta_i}\partial_{\theta_j}} L$, or

$$\mathbf{H}_{i,j} = \frac{\partial^2}{\partial_{\theta_i}\partial_{\theta_j}} \mathrm{E}_X[l(X;\boldsymbol{\theta})] \tag{14}$$

$$= \int_{\mathcal{D}_X} \mathrm{d}X \, \frac{\partial^2}{\partial_{\theta_i}\partial_{\theta_j}} l(X;\boldsymbol{\theta}) p(X;\boldsymbol{\theta}) \tag{15}$$

because we require that we can exchange order of integration and differentiation. Assume now that the error function can be written as $l(X;\boldsymbol{\theta}) = b(\boldsymbol{\theta}) - \ln c(X;\boldsymbol{\theta})$, for some functions b and c. If $l(X;\boldsymbol{\theta}) \geq 0$ for all $X, \boldsymbol{\theta}$, this implies $0 \leq b(\boldsymbol{\theta})$ and $0 < c(X;\boldsymbol{\theta}) \leq 1$. As a consequence,

$$\mathbf{H}_{i,j} = \int_{\mathcal{D}_X} \mathrm{d}X \, \frac{\partial^2}{\partial_{\theta_i}\partial_{\theta_j}} b(\boldsymbol{\theta}) p(X;\boldsymbol{\theta}) \tag{16}$$

$$- \int_{\mathcal{D}_X} \mathrm{d}X \, \frac{\partial^2}{\partial_{\theta_i}\partial_{\theta_j}} \ln c(X;\boldsymbol{\theta}) p(X;\boldsymbol{\theta}). \tag{17}$$

Using $\partial_{\theta_i} \int p(X;\boldsymbol{\theta})) = 0$, the Hessian reduces to the FIM in the particular case that

$$c(X;\boldsymbol{\theta}) = p(X;\boldsymbol{\theta}), \tag{18}$$

$$\frac{\partial^2}{\partial_{\theta_i}\partial_{\theta_j}} b(\boldsymbol{\theta}) = \int_{\mathcal{D}_X} \mathrm{d}X \, \ln p(X;\boldsymbol{\theta}) \frac{\partial^2}{\partial_{\theta_i}\partial_{\theta_j}} p(X;\boldsymbol{\theta}), \tag{19}$$

i.e., the Hessian and Fisher information matrix overlap exactly. In practice, this means that a class of problems where the natural gradient method is equivalent to the Newton method are those where the probability density function is exponential in the error function, i.e., $p(X;\boldsymbol{\theta}) = \exp(b(\boldsymbol{\theta}) - l(X;\boldsymbol{\theta})))$. The connection between Fisher information and Hessian has been utilized before in the domain of neural network optimization with Gaussian statistics [73, 83].

The above relation for the classical Fisher information matrix and Hessian takes for variational quantum algorithms the form $\mathcal{L} = \int \mathrm{d}X \, p(X;\boldsymbol{\theta}) l(X;\boldsymbol{\theta})$ with probability density function $p(X;\boldsymbol{\theta}) = \Psi^*(X;\boldsymbol{\theta})\Psi(X;\boldsymbol{\theta})$ and energy density $l(X;\boldsymbol{\theta}) = \Psi^*(X;\boldsymbol{\theta})\hat{H}\Psi(X;\boldsymbol{\theta})/p(X;\boldsymbol{\theta})$.

F Properties of the approximate metric

For the optimization algorithms we have introduced, the update rule

$$\mathbf{B}_{k+1} = (1-\varepsilon_k)\mathbf{B}_k + \varepsilon_k \nabla\mathcal{L}_k \nabla\mathcal{L}_k^\top. \tag{20}$$

is applied to iterate on the metric. If the initial matrix $\mathbf{B}_0$ is positive semi-definite ($\mathbf{B}_0 \succeq 0$), the update rule preserves this property for all $\mathbf{B}_k$. To see this, first assume $\mathbf{B}_k \succeq 0$. Then it holds that $(1-\varepsilon_k)\mathbf{B}_k \succeq 0$ for all $\varepsilon_k \in (0,1)$. Next, $\varepsilon_k \nabla\mathcal{L}_k \nabla\mathcal{L}_k^\top \succeq 0$ for all $\varepsilon_k > 0$, because

$$\boldsymbol{x}^\top \nabla\mathcal{L}_k \nabla\mathcal{L}_k^\top \boldsymbol{x} = \langle \boldsymbol{x}, \nabla\mathcal{L}_k \rangle \langle \nabla\mathcal{L}_k, \boldsymbol{x} \rangle \tag{21}$$

$$= \langle \nabla\mathcal{L}_k, \boldsymbol{x} \rangle^2 \geq 0 \tag{22}$$

for all $\boldsymbol{x}$. The sum of two matrices that are positive semi-definite is again positive semi-definite. Additionally, if we initialise $\mathbf{B}_0 \succ 0$, we preserve $\mathbf{B}_k \succ 0$ for all k. Consequently, it follows that $\mathbf{B}_k^{-1}$ exists and is positive definite for all k.

To provide further intuition for the algorithm, we study the long-term behavior of the metric under the update rule. Each step taken in the parameter space is defined by the vector $\Delta_k = \mathbf{B}_k^{-1}\nabla\mathcal{L}(\boldsymbol{\theta}_k)$. Now, we insert in the Δ_{k+1} explicitly the expression

$$\mathbf{B}_{k+1}^{-1} = \left[\mathbb{1} - \frac{\varepsilon_k \mathbf{B}_k^{-1}\nabla\mathcal{L}_k\nabla\mathcal{L}_k^\top}{1-\varepsilon_k(1-\nabla\mathcal{L}_k^\top\mathbf{B}_k^{-1}\nabla\mathcal{L}_k)}\right]\frac{\mathbf{B}_k^{-1}}{1-\varepsilon_k} \tag{23}$$

to get

$$\Delta_{k+1} = \left[\mathbb{1} - \frac{\varepsilon_k \mathbf{B}_k^{-1}\nabla\mathcal{L}(\boldsymbol{\theta}_k)\nabla\mathcal{L}(\boldsymbol{\theta}_k)^\top}{1+\varepsilon_k(\nabla\mathcal{L}(\boldsymbol{\theta}_k)^\top\mathbf{B}_k^{-1}\nabla\mathcal{L}(\boldsymbol{\theta}_k)-1)}\right]\frac{\mathbf{B}_k^{-1}\nabla\mathcal{L}(\boldsymbol{\theta}_{k+1})}{1-\varepsilon_k} \tag{24}$$

$$= \left[\mathbf{B}_k^{-1} - \frac{\varepsilon_k \Delta_k\Delta_k^\top}{1+\varepsilon_k(\Delta_k^\top\nabla\mathcal{L}(\boldsymbol{\theta}_k)-1)}\right]\frac{\nabla\mathcal{L}(\boldsymbol{\theta}_{k+1})}{1-\varepsilon_k}. \tag{25}$$

Since $\lim_{k\to\infty}\varepsilon_k = 0$, for sufficiently large k the effective step is $\Delta_k \approx \mathbf{B}_{k-1}^{-1}\nabla\mathcal{L}(\boldsymbol{\theta}_k)$. Let us denote the second term inside the parenthesis of Eq. (25) by $\gamma_k = \varepsilon_k\Delta_k\Delta_k^\top/(1+\varepsilon_k(\Delta_k^\top\nabla\mathcal{L}(\boldsymbol{\theta}_k)-1))$ and refer to it as the *innovation* at each step. Expanding from the initial point and defining $\varepsilon_{-1}=0$, the generic step can be written

$$\Delta_k = \left[\mathbf{B}_0^{-1} - \boldsymbol{\Gamma}_k\right]\frac{\nabla\mathcal{L}(\boldsymbol{\theta}_k)}{\prod_{m=0}^{k-1}(1-\varepsilon_m)}, \tag{26}$$

where $\boldsymbol{\Gamma}_k = \sum_{m=0}^{k-1}\gamma_m\prod_{n=0}^{m}(1-\varepsilon_{n-1})$ is the matrix of corrections to the metric picked up by the innovations from the first $k-1$ steps. We have that $\prod_{k=0}^{n}(1-\frac{\varepsilon_0}{k+1}) = \frac{(1-\varepsilon_0)}{\Gamma(2-\varepsilon_0)}\cdot\frac{\Gamma(n+2-\varepsilon_0)}{\Gamma(n+2)}$ and that $\frac{\Gamma(n+2-\varepsilon_0)}{\Gamma(n+2)} \sim n^{-\varepsilon_0}$ as $n\to\infty$. Since $\varepsilon_k = \varepsilon_0/(k+1)$, the innovations are attenuated $\propto k^{-1}$, and, for some number of steps $k' \gg 1$, the innovations can be considered negligible. In this regime, where $k > k'$, the step taken is $\Delta_k \propto (k-1)^{\varepsilon_0}\left[\mathbf{B}_0^{-1} - \boldsymbol{\Gamma}_{k'}\right]\nabla\mathcal{L}(\boldsymbol{\theta}_k)$, where the approximate metric $\mathbf{B}_0^{-1}-\boldsymbol{\Gamma}_{k'}$ can be considered constant.

This behavior invites a possible modification to the algorithms, where, if convergence has not been achieved after k' steps, the metric is reinitialized at the current parameters by computing the full FIM matrix at $\boldsymbol{\theta}_{k'}$ and the algorithm restarted.

G Connection of VarQITE and QNG

As stated in the main text, there is a close relationship between the QFIM, $\mathbf{F}$, and the Fubini-Study metric, $\mathbf{A}$, which is given by

$$A_{ij} = \mathrm{Re}\left\{\left\langle\partial_{\theta_i}\Phi|\partial_{\theta_j}\Phi\right\rangle - \langle\partial_{\theta_i}\Phi|\Phi\rangle\left\langle\Phi|\partial_{\theta_j}\Phi\right\rangle\right\}, \tag{27}$$

where, $\partial_{\theta_i} \equiv \frac{\partial}{\partial\theta_i}$. The Fubini-Study metric [60, 61, 62, 63, 64], is the metric of parametrized **pure** quantum states $|\Phi(\boldsymbol{\theta})\rangle$. $\mathbf{A}$ can be expressed as the real part of a more general quantum geometric tensor (QGT) [102, 63, 103, 104]

$$G_{ij} = \left\langle\partial_{\theta_i}\Phi|\partial_{\theta_j}\Phi\right\rangle - \langle\partial_{\theta_i}\Phi|\Phi\rangle\left\langle\Phi|\partial_{\theta_j}\Phi\right\rangle, \tag{28}$$

whose imaginary part corresponds to the Berry geometrical phase [105, 106, 71, 63].

For pure states – as we consider exclusively in this work – the Fubini-Study metric (in matrix form) is (up to a factor of 4) equivalent to the QFIM [65, 24, 66, 67], i.e., $\mathbf{F} = 4\mathbf{A}$. The factor of 4 could, however, be absorbed by a change of variables [71] or in the time-step $\delta\tau = \frac{\eta}{4}$ as we did in the main text. Thus we use the terms Fubini-study metric/QFIM and variables $\mathbf{A}$ and $\mathbf{F}$ interchangeably in the main text.

The matrices $\mathbf{A}$ and $\mathbf{F}$ describe the geometry of the parameter space rather than the energy landscape. The second term of Eq. (8) resolves a possible arbitrary overall phase mismatch between $|\Phi(\boldsymbol{\theta}(\tau))\rangle$ and the target state $|\Psi(\tau)\rangle$ along the imaginary time propagation [59, 32]. Using different variational principles (time-dependent/Dirac-Frenkel) [32, 107] yields slightly different equations for the metric and gradient resulting in possibly complex values of the parameters $\boldsymbol{\theta}$ (see Ref. number [32] for details). As $\boldsymbol{\theta}$ usually refers to real-valued angles of rotational gates in a PQC, solving

Eq. (3) from the main text using McLachlan's variational principle is preferred in the VarQITE setting, as it ensures real-valued solutions for $\frac{\partial \boldsymbol{\theta}}{\partial \tau}$. If $|\Phi\rangle$ and $\partial_{\theta_i}|\Phi\rangle$ are real (not to be confused with real parameters), the second term in Eq. (8) vanishes, due to the normalization of $|\Phi\rangle$, $\langle\Phi|\Phi\rangle = 1$

$$\langle \partial_{\theta_i}\Phi|\Phi\rangle + \langle\Phi|\partial_{\theta_i}\Phi\rangle = \partial_{\theta_i} 1 = 0. \tag{29}$$

Due to the above-mentioned relation between the Fubini-Study metric and QFIM, $\mathbf{F} = 4\mathbf{A}$, Eq. (6) from the main text reveals that QNG is equivalent to VarQITE when the energy of the system is used as the cost function, $\mathcal{L} = \langle \hat{H} \rangle$, and $\eta = 4\delta\tau$.

Additionally, VarQITE is closely related to the stochastic reconfiguration (SR) method of Sorella [108, 109, 110], which is a second-order iterative approximation to the "classical" ITE.

H Ensuring Positive Definiteness of the Quantum Fisher Information Matrix

In the presence of noise, especially shot noise, the method used to estimate the QFIM may produce a matrix that is not positive semi-definite. This is problematic as it could adversely affect the optimization process, potentially leading to unstable or divergent behavior. To mitigate this issue, we employ a diagonal loading to ensure that the QFIM remains positive definite (PD). The method is straightforward but crucial for the robustness of our optimization algorithms. We first compute the eigenvalues of the QFIM. If the matrix has any negative eigenvalues, we identify the most negative one, say $\lambda_{\min}$. We then add $(\gamma_{\text{reg}} - \lambda_{\min})$ times the identity matrix to the QFIM, where γ_{reg} is a small regularising parameter. Mathematically, this can be expressed as:

$$\mathbf{F}_{\text{PD}} = \begin{cases} \mathbf{F} + (\gamma_{\text{reg}} - \lambda_{\min})\mathbb{1} & \text{if } \lambda_{\min} < 0, \\ \mathbf{F} & \text{otherwise.} \end{cases}$$

Here, $\mathbf{F}$ is the original QFIM and $\mathbb{1}$ is the identity matrix of the same dimension as $\mathbf{F}$.