Adaptive Observation Cost Control for Variational Quantum Eigensolvers

Christopher J. Anders [*1 2] **Kim A. Nicoli** [*3 4] **Bingting Wu** [2] **Naima Elosegui** [1 2] **Samuele Pedrielli** [2]
Lena Funcke [3 4] **Karl Jansen** [5] **Stefan Kühn** [5] **Shinichi Nakajima** [1 2 6]

Abstract

The objective to be minimized in the variational quantum eigensolver (VQE) has a restricted form, which allows a specialized sequential minimal optimization (SMO) that requires only a few observations in each iteration. However, the SMO iteration is still costly due to the observation noise—one *observation* at a point typically requires averaging over hundreds to thousands of repeated quantum *measurement shots* for achieving a reasonable noise level. In this paper, we propose an adaptive cost control method, named *subspace in confident region* (SubsCoRe), for SMO. SubsCoRe uses the Gaussian process (GP) surrogate, and requires it to have low uncertainty over the subspace being updated, so that optimization in each iteration is performed with guaranteed accuracy. The adaptive cost control is performed by first setting the required accuracy according to the progress of the optimization, and then choosing the minimum number of measurement shots and their distribution such that the required accuracy is satisfied. We demonstrate that SubsCoRe significantly improves the efficiency of SMO, and outperforms the state-of-the-art methods.

1. Introduction

Introduced a decade ago by Peruzzo et al. (2014), the variational quantum eigensolver (VQE) (McClean et al., 2016) has become a widespread hybrid quantum-classical algorithm for approximating the ground-state of a given Hamiltonian. It utilizes a quantum device to efficiently prepare a trial state in the form of a parameterized ansatz circuit and to measure the expectation value of the Hamiltonian, i.e., the energy, in the given ansatz. Subsequently, based on the measurement outcome, a classical optimizer is used to compute a new set of parameters that likely lowers the energy. Running the feedback loop between the quantum device and the classical optimizer until convergence, the parametric ansatz encodes an approximation for the ground-state of the Hamiltonian, and the final energy measurement provides an estimate for the ground-state energy.

VQEs may offer the potential to overcome computational challenges in many fields of science, including quantum chemistry, biology, and material science (Kandala et al., 2017; Grimsley et al., 2019; Bauer et al., 2020; Fedorov & Gelfand, 2021; Ollitrault et al., 2020), where accurately identifying the ground-state energy of complex molecules or materials is intractable for classical computers. By leveraging the principles of quantum mechanics, VQEs may efficiently solve such problems, enabling advancements in drug discovery, materials design, and other fields (Cao et al., 2018). Furthermore, applications of VQEs to other computationally challenging tasks, e.g., high-energy physics calculations (Humble et al., 2022; Di Meglio et al., 2023) including lattice field theory simulations (Funcke et al., 2023), have been recently investigated. We refer readers to Tilly et al. (2022) for a comprehensive review.

As is typical for any type of quantum computing protocols, VQEs suffer from two types of noise—hardware noise and shot noise. The hardware noise is caused by the imperfect quantum circuit that systematically or stochastically distorts the observations, while the shot noise is due to the intrinsic (and therefore unavoidable) stochastic nature of quantum computing. Throughout the paper, we do not consider the hardware noise[*] and focus on the *shot noise*, i.e., statistical uncertainties due to a finite number of repeated measurements (shots) to estimate expected values of observables on the quantum computer. We treat the shot noise as i.i.d. *observation noise* when regression is applied to the measurements.

[*]Equal contribution [1]Berlin Institute for the Foundations of Learning and Data (BIFOLD) [2]Technische Universität Berlin, Germany [3]Transdisciplinary Research Area (TRA) Matter, University of Bonn, Germany [4]Helmholtz Institute for Radiation and Nuclear Physics (HISKP), University of Bonn, Germany [5]CQTA, Deutsches Elektronen-Synchrotron (DESY), Zeuthen, Germany [6]RIKEN Center for AIP, Japan. Correspondence to: <anders/nakajima@tu-berlin.de>, <knicoli@uni-bonn.de>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

[*]Recent progress in quantum hardware development has substantially reduced the hardware noise, see, e.g., Bluvstein et al. (2023).

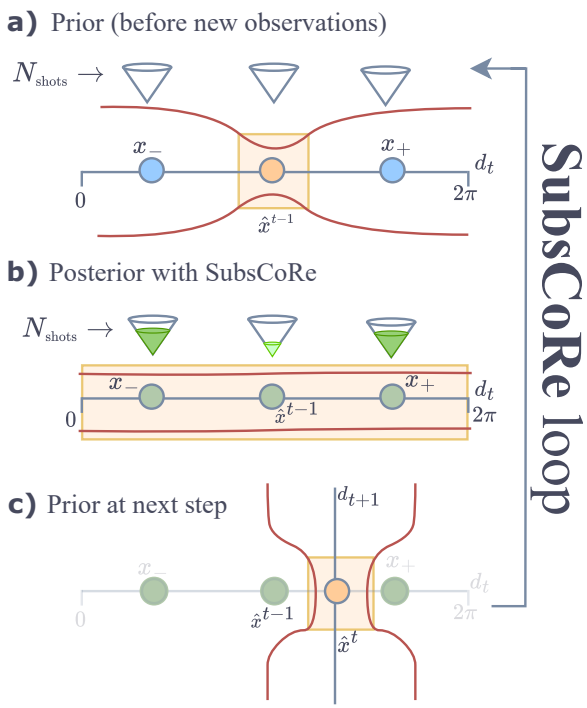

Figure 1. Illustration of SubsCoRe. (a) After iteration $t-1$, the function value at the best point $\widehat{\boldsymbol{x}}^{t-1}$, which is in the CoRe (yellow rectangle) due to the previous observations, is predicted by the GP with sufficiently low uncertainty (red curves). Two equidistant shifts (blue points) at $\widehat{\boldsymbol{x}}^{t-1} \pm \frac{2\pi}{3}\boldsymbol{e}_{d_t}$ are chosen, yet not measured, along the line parallel to the d_t axis at the current step. (b) SubsCoRe finds the minimum number of measurement shots for measuring the points $\{\widehat{\boldsymbol{x}}^{t-1}, \boldsymbol{x}_\pm\}$ such that the entire line will be included in the CoRe, i.e., the *posterior uncertainty* at any point on the line is smaller than the required threshold. The three points are measured (green circles), with the middle point $\widehat{\boldsymbol{x}}^{t-1}$ typically requiring fewer shots because of its lower prior uncertainty. (c) Using the trigonometric polynomial regression, the minimum $\widehat{\boldsymbol{x}}^t$ (orange point) along the line (where the GP mean function is fully identified by predicting *any* three points) is computed along with the corresponding energy, thus becoming the starting point for the next iteration.

For quantum circuits that consist of parametrized rotation gates and non-parametric unitary gates (e.g., the Hadamard gate and the Controlled-Z gate), it is known that the objective function in VQE is a low-order trigonometric polynomial along each axis, and therefore can be fit exactly by a linear model with low degrees of freedom (Nakanishi et al., 2020). Based on this property, Nakanishi et al. (2020) proposed an efficient sequential minimal optimization (SMO) (Platt, 1998)—often referred to as Nakanishi-Fuji-Todo (NFT)—which requires only a few (two in the most typical case) observations in each iteration of the 1D subspace optimization. Nicoli et al. (2023a) introduced the corresponding VQE kernel, and enhanced NFT with Gaussian processes (GP). Specifically, they applied Bayesian optimization (BO) to find the best locations to be observed in each SMO iteration. To make use of the property that only a few points determine the complete functional form in the 1D subspace, they introduced the notion of confident region (CoRe)—the region on which the uncertainty of the GP prediction is lower than a threshold—and proposed the expected maximum improvement over CoRe (EMICoRe) acquisition function to evaluate how much the CoRe extends and how much improvement is expected within the CoRe after new points are observed. BO with EMICoRe successfully enhanced NFT, proving the usefulness of GP for VQE (Nicoli et al., 2023a).

In this paper, we enhance NFT in an orthogonal way—introducing adaptive observation cost control. Due to the shot noise, each *observation*, i.e., each evaluation of the cost function value at a single point, consists of hundreds to thousands of *measurement shots*—a repeated quantum computation from the initial state preparation to the final quantum state observation, often also referred to as *readouts*. The computational cost on the quantum computer is largely dominated by this quantum process and should, therefore, be measured by the total number of shots. This implies that one can improve the computational efficiency by adapting the number of shots according to the progress of optimization, e.g., saving shots in the early phase where the best value rapidly decreases, and using more shots in the converging phase where accurate prediction guides the optimizer towards the very bottom of the objective, i.e., the ground-state. Such strategies adapting the number of shots have been proposed and applied to gradient-based methods. For example, Tamiya & Yamasaki (2022) proposed the stochastic gradient line BO (SGLBO) approach, where the number of shots is set according to the norm of the estimated gradient, and showed that SGLBO outperforms NFT with a fixed number of shots. An adaptive-shots strategy for NFT—where the gradient is not estimated—is yet to be established and is the primary goal of this paper.

Our approach, named *subspace in confident region* (SubsCoRe), distributes the minimum number of shots in each SMO iteration so that the uncertainty in the updated subspace is lower than a threshold. In other words, we search for the minimum number of shots, i.e., minimize the observation cost, such that the optimized subspace is a subset of the CoRe (see Figure 1). Although identifying the optimal distribution of thousands of shots seems intractable, our theory implies that we do not need to distribute the shots thinly over the subspace. More specifically, observing $1 + 2V_d$ equidistant points, where V_d is the number of gates parameterized by the d-th variable, with equally divided shots results in a min-max optimal uniform posterior uncertainty. Based on our theory, we propose a few variants of SubsCoRe and show their state-of-the-art performance.

Related Work Some of the recently proposed optimization techniques for VQEs rely on special properties of the VQE objective, i.e., the parameter shift rule (PSR) (Mitarai et al., 2018; Schuld et al., 2019) and the low-order trigonometric polynomial form along each axis (Nakanishi et al., 2020), which have been shown to be equivalent (Nicoli et al., 2023a). The PSR is used for efficient gradient estimation (Mitarai et al., 2018), while the trigonometric polynomial form allows for efficient SMO, i.e., NFT (Nakanishi et al., 2020). Vanilla BO with a GP surrogate was also applied to VQE (Iannelli & Jansen, 2021; Tamiya & Yamasaki, 2022; Nicoli et al., 2023a). Tamiya & Yamasaki (2022) combined stochastic gradient descent and 1D BO with an adaptive number of measurement shots, which outperforms NFT with a *fixed* number of shots. Nicoli et al. (2023a) introduced the VQE kernel and the EMICoRe acquisition function, and enhanced NFT with BO. Although the usefulness of GP for VQE has already been reported, to the best of our knowledge, no prior work has used it for observation cost control. In the more general BO context, works dealing with heteroscedastic noise by incorporating the variance in the acquisition function for BO and active learning have also been proposed (Mickaël Binois & Ludkovski, 2019; Makarova et al., 2021).

Difference between EMICoRe and SubsCoRe All approaches analyzed in our numerical experiments, i.e, the NFT, EMICoRe, SGLBO and the newly proposed SubsCoRe, differ substantially from each other. While the difference between SubsCoRe and SGLBO is apparent, here we elaborate on the crucial differences between EMICoRe and SubsCoRe. A key feature of both SubsCoRe and EMICoRe is that both approaches rely on the notion of Confident Region (CoRe) and on the VQE kernel introduced by Nicoli et al. (2023a). However, besides this, the objective and the optimization of both approaches substantially differ. EMICoRe uses a special acquisition function (based on the CoRe) to select the best points to measure at the next iteration of SMO, and the observations are performed at a fixed number of shots (repeated measurements on a quantum computer). Namely, it tries to minimize the number of SMO steps required to converge with a fixed number of shots for each observed point. This approach is fundamentally different from SubsCoRe, which focuses on minimizing the total quantum computing budget, i.e., the total number of shots, regardless of the number of classical optimization steps. We stress that in VQEs the crucial bottleneck is the number of operations performed on the quantum device, and, therefore, minimizing the total number of shots as the "observation cost" is more appropriate than minimizing the SMO steps required for convergence. In addition, SubsCoRe relies on the min-max optimality of the equidistant measurements under the "subspace in the CoRe" requirement, and thus observes fixed points in each SMO step without using acquisition functions. SubsCoRe adapts the number of shots based on how the CoRe extends after the measurements, and, therefore, the necessary number of shots is inherently tied to the confidence of the GP. More specifically, our algorithm chooses the number of measurement shots at each observed point such that the posterior variance is within a given threshold, which is adapted during the optimization. To summarize, both the methodology and the objective of EMICoRe and SubsCoRe are fundamentally different, despite both relying on the notion of CoRe. Furthermore, we note that both approaches are not mutually exclusive and can be combined. We defer such a study to future work.

2. Background

2.1. Gaussian Process (GP) Regression with Heteroscedastic Observation Noise

Assume that we aim to learn an unknown function $f^*(\cdot) : \mathcal{X} \mapsto \mathbb{R}$ from the training data $\boldsymbol{X} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_N) \in \mathcal{X}^N, \boldsymbol{y} = (y_1, \dots, y_N)^\top \in \mathbb{R}^N, \boldsymbol{\sigma} = (\sigma_1^2, \dots, \sigma_N^2) \in \mathbb{R}_{++}^N$ that fulfills

$$y_n = f^*(\boldsymbol{x}_n) + \varepsilon_n, \qquad \varepsilon_n \sim \mathcal{N}_1(y_n; 0, \sigma_n^2), \tag{1}$$

where $\mathcal{N}_D(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the D-dimensional Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. With the Gaussian process (GP) prior

$$p(f(\cdot)) = \mathrm{GP}(f(\cdot); 0(\cdot), k(\cdot, \cdot)), \tag{2}$$

where $0(\cdot)$ and $k(\cdot, \cdot)$ are the prior zero-mean and the kernel (covariance) functions, respectively, the posterior distribution of the function values $\boldsymbol{f}' = (f(\boldsymbol{x}_1'), \dots, f(\boldsymbol{x}_M'))^\top \in \mathbb{R}^M$ at arbitrary test points $\boldsymbol{X}' = (\boldsymbol{x}'_1, \dots, \boldsymbol{x}'_M) \in \mathcal{X}^M$ is given as

$$p(\boldsymbol{f}'|\boldsymbol{X}, \boldsymbol{y}) = \mathcal{N}_M(\boldsymbol{f}'; \boldsymbol{\mu}'_{|\boldsymbol{X},\boldsymbol{\sigma},\boldsymbol{y}}, \boldsymbol{S}'_{|\boldsymbol{X},\boldsymbol{\sigma}}), \quad \text{where} \tag{3}$$

$$\boldsymbol{\mu}'_{|\boldsymbol{X},\boldsymbol{\sigma},\boldsymbol{y}} = \boldsymbol{K}'^\top \left(\boldsymbol{K} + \mathbf{Diag}(\boldsymbol{\sigma})\right)^{-1} \boldsymbol{y}, \tag{4}$$

$$\boldsymbol{S}'_{|\boldsymbol{X},\boldsymbol{\sigma}} = \boldsymbol{K}'' - \boldsymbol{K}'^\top \left(\boldsymbol{K} + \mathbf{Diag}(\boldsymbol{\sigma})\right)^{-1} \boldsymbol{K}' \tag{5}$$

(Rasmussen & Williams, 2006). Here $\mathbf{Diag}(\boldsymbol{v})$ is the diagonal matrix with $\boldsymbol{v}$ specifying the diagonal entries, and $\boldsymbol{K} = k(\boldsymbol{X}, \boldsymbol{X}) \in \mathbb{R}^{N \times N}, \boldsymbol{K}' = k(\boldsymbol{X}, \boldsymbol{x}') \in \mathbb{R}^{N \times M}$, and $\boldsymbol{K}'' = k(\boldsymbol{x}', \boldsymbol{x}'') \in \mathbb{R}^{M \times M}$ are the train, train-test, and test kernel matrices, respectively, where $k(\boldsymbol{X}, \boldsymbol{X}')$ denotes the kernel matrix evaluated at each column of $\boldsymbol{X}$ and $\boldsymbol{X}'$ such that $(k(\boldsymbol{X}, \boldsymbol{X}'))_{n,m} = k(\boldsymbol{x}_n, \boldsymbol{x}_m)$. We also denote the posterior as $p(f(\cdot)|\boldsymbol{X}, \boldsymbol{y}) = \mathrm{GP}(f(\cdot); \mu_{|\boldsymbol{X},\boldsymbol{\sigma},\boldsymbol{y}}(\cdot), s_{|\boldsymbol{X},\boldsymbol{\sigma}}(\cdot, \cdot))$ with the posterior mean $\mu_{|\boldsymbol{X},\boldsymbol{\sigma},\boldsymbol{y}}(\cdot)$ and covariance $s_{|\boldsymbol{X},\boldsymbol{\sigma}}(\cdot, \cdot)$ functions, e.g., $\mu_{|\boldsymbol{X},\boldsymbol{\sigma},\boldsymbol{y}}(\boldsymbol{x}') = \boldsymbol{\mu}'_{|\boldsymbol{X},\boldsymbol{\sigma},\boldsymbol{y}} \in \mathbb{R}$ and $s_{|\boldsymbol{X},\boldsymbol{\sigma}}(\boldsymbol{x}', \boldsymbol{x}') = \boldsymbol{S}'_{|\boldsymbol{X},\boldsymbol{\sigma}} \in \mathbb{R}_{++}$ for a single test point $\boldsymbol{X}' = (\boldsymbol{x}')$.

2.2. Variational Quantum Eigensolvers

The VQE (Peruzzo et al., 2014; McClean et al., 2016) is a hybrid quantum-classical computing protocol for estimating the ground-state energy of a given quantum Hamiltonian for a Q-qubit system. A quantum computer is used to prepare a trial state $|\psi_0\rangle$ transformed via a parametric quantum circuit $G(\boldsymbol{x})$, which depends on D angular parameters $\boldsymbol{x} \in \mathcal{X} = [0, 2\pi)^D$. The final state $|\psi_{\boldsymbol{x}}\rangle$ is thus generated by applying $D'(\geq D)$ *quantum gate operations*, $G(\boldsymbol{x}) = G_{D'} \circ \cdots \circ G_1$, to the initial trial state $|\psi_0\rangle$, i.e., $|\psi_{\boldsymbol{x}}\rangle = G(\boldsymbol{x})|\psi_0\rangle$. All gates $\{G_{d'}\}_{d'=1}^{D'}$ are unitary operators, parameterized by at most one variable x_d. Let $d(d') : \{1, \ldots, D'\} \mapsto \{1, \ldots, D\}$ be the mapping specifying which one of the variables $\{x_d\}$ parameterizes the d'-th gate. We consider parametric gates of the form $G_{d'}(x) = U_{d'}(x_{d(d')}) = \exp\left(-ix_{d(d')}P_{d'}/2\right)$, where $P_{d'}$ is an arbitrary sequence of the Pauli operators $\{\mathbf{1}_q, \sigma_q^X, \sigma_q^Y, \sigma_q^Z\}_{q=1}^Q$ acting on each qubit at most once. This general structure covers both single-qubit gates, such as $R_X(x) = \exp\left(-i\theta\sigma_q^X\right)$, and entangling gates acting on multiple qubits simultaneously, such as $R_{XX}(x) = \exp\left(-ix\sigma_{q_1}^X \circ \sigma_{q_2}^X\right)$ and $R_{ZZ}(x) = \exp\left(-ix\sigma_{q_1}^Z \circ \sigma_{q_2}^Z\right)$ for $q_1 \neq q_2$, commonly realized in trapped-ion quantum hardware setups (Kielpinski et al., 2002; Debnath et al., 2016).

The quantum computer is used to evaluate the energy of the resulting quantum state $|\psi_{\boldsymbol{x}}\rangle$ by observing

$$y = f^*(\boldsymbol{x}) + \varepsilon, \qquad \text{where}$$
$$f^*(\boldsymbol{x}) = \langle\psi_{\boldsymbol{x}}|H|\psi_{\boldsymbol{x}}\rangle = \langle\psi_0|G(\boldsymbol{x})^\dagger H G(\boldsymbol{x})|\psi_0\rangle, \tag{6}$$

where $\dagger$ denotes the Hermitian conjugate. For each observation, multiple measurement shots, denoted by N_{shots}, are acquired to suppress the variance $\sigma^{*2}(N_{\text{shots}})$ of the noise ε in evaluating the expectation value of the Hamiltonian, i.e., the energy of the quantum system.[†] Since the observation y is the sum of many random variables, it approximately follows a Gaussian distribution, according to the central limit theorem. The Gaussian likelihood (1) therefore approximates the observation y well if $f(\boldsymbol{x}) \approx f^*(\boldsymbol{x})$ and $\sigma^2 \approx \sigma^{*2}(N_{\text{shots}})$. Using the noisy estimates of $f^*(\boldsymbol{x})$ obtained from the quantum computer, a protocol running on a classical computer is used to solve the following minimization problem:

$$\min_{\boldsymbol{x}\in[0,2\pi)^D} f^*(\boldsymbol{x}), \tag{7}$$

thus finding the minimizer $\widehat{\boldsymbol{x}}$, i.e., the optimal parameters for the (rotational) quantum gates.

Let V_d be the number of gates parameterized by x_d, i.e., $V_d = |\{d' \in \{1, \ldots D'\}; d = d(d')\}|$, and

$$\boldsymbol{\psi}_\gamma(\theta) = (\gamma, \sqrt{2}\cos\theta, \sqrt{2}\cos 2\theta, \ldots, \sqrt{2}\cos V_d\theta, \sqrt{2}\sin\theta, \sqrt{2}\sin 2\theta, \ldots, \sqrt{2}\sin V_d\theta)^\top \in \mathbb{R}^{1+2V_d} \tag{8}$$

be the (1-dimensional) V_d-th order Fourier basis for arbitrary $\gamma > 0$. Nakanishi et al. (2020) found that the VQE objective function $f^*(\cdot)$ in Eq. (6) with any[‡] $G(\cdot)$, H, and $|\psi_0\rangle$ can be exactly expressed as

$$f^*(\boldsymbol{x}) = \boldsymbol{b}^\top \mathbf{vec}\left(\otimes_{d=1}^D \boldsymbol{\psi}_\gamma(x_d)\right) \tag{9}$$

for some $\boldsymbol{b} \in \mathbb{R}^{\prod_{d=1}^D(1+2V_d)}$, where $\otimes$ and $\mathbf{vec}(\cdot)$ denote the tensor product and the vectorization operator for a tensor, respectively. Based on this property, the NFT (Nakanishi et al., 2020) method performs SMO (Platt, 1998), where the optimum in the 1D subspace in each iteration is analytically estimated from only $1 + 2V_d$ observations.

Nakanishi-Fuji-Todo (NFT) Algorithm Let $\{\boldsymbol{e}_d\}_{d=1}^D$ be the standard basis. NFT is initialized with a random point $\widehat{\boldsymbol{x}}^0$ with a first observation $\widehat{y}^0 = f^*(\widehat{\boldsymbol{x}}^0) + \varepsilon_0$, and iterates the following procedure: for each iteration step t,

1. Select an axis $d \in \{1, \ldots, D\}$ sequentially and observe the objective $\boldsymbol{y} \in \mathbb{R}^{2V_d}$ at $2V_d$ points $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{2V_d}) = \{\widehat{\boldsymbol{x}}^{t-1} + \alpha_w \boldsymbol{e}_d\}_{w=1}^{2V_d} \in \mathbb{R}^{D\times 2V_d}$ along the axis d.[§] Here $\boldsymbol{\alpha} \in [0, 2\pi)^{2V_d}$ is such that $\alpha_w \neq 0$, $\alpha_{w'} \neq \alpha_w$, for all w and $w' \neq w$.
2. Apply the 1D trigonometric polynomial regression $\widetilde{f}(\theta) = \widetilde{\boldsymbol{b}}^\top \boldsymbol{\psi}_1(\theta)$ to the $2V_d$ new observations $\boldsymbol{y}$, together with the previous best estimated score $\widehat{y}^{t-1}$, and analytically compute the new optimum $\widehat{\boldsymbol{x}}^t = \widehat{\boldsymbol{x}}^{t-1} + \widehat{\theta}\boldsymbol{e}_d$, where $\widehat{\theta} = \operatorname{argmin}_\theta \widetilde{f}(\theta)$.
3. Update the best score by $\widehat{y}^t = \widetilde{f}(\widehat{\theta})$.

Note that if the observation noise is negligible, i.e., $y \approx f^*(\boldsymbol{x})$, each step of NFT reaches the global optimum in the 1D subspace along the chosen axis d for any choice of $\boldsymbol{\alpha}$, and thus performs SMO exactly. Otherwise, errors can be accumulated in the best score $\widehat{y}^t$, and therefore an additional measurement may need to be performed at $\widehat{\boldsymbol{x}}^t$ after a certain iteration interval.

Inspired by the trigonometric polynomial form (9), Nicoli

[†]When the Hamiltonian consists of N_{og} groups of non-commuting operators, each of which needs to be measured separately, N_{shots} denotes the number of shots *per operator group*. Therefore, the number of shots *per observation* is $N_{\text{og}} \times N_{\text{shots}}$. Throughout the paper, we use the total number of shots per operator group, i.e., the cumulative sum of N_{shots} over all observations, to evaluate the observation cost.

[‡]Any circuit consisting of parametrized rotation gates and non-parametric unitary gates as stated in the introduction.

[§]With slight abuse of notation, we use the set notation to specify the column vectors of a matrix, i.e., $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) = \{\boldsymbol{x}_n\}_{n=1}^N$.

et al. (2023a) used the corresponding VQE kernel

$$k_\gamma(\boldsymbol{x}, \boldsymbol{x}') = \sigma_0^2 \prod_{d=1}^{D} \left(\frac{\gamma^2 + 2\sum_{v=1}^{V_d} \cos\big(v(x_d - x'_d)\big)}{\gamma^2 + 2V_d} \right), \quad (10)$$

which is decomposed as $k_\gamma(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\phi}_\gamma(\boldsymbol{x})^\top \boldsymbol{\phi}_\gamma(\boldsymbol{x}')$ for $\boldsymbol{\phi}_\gamma(\boldsymbol{x}) = \frac{\sigma_0}{(\gamma^2+2V_d)^{D/2}} \mathbf{vec}\left(\otimes_{d=1}^{D} \boldsymbol{\psi}_\gamma(x_d)\right)$, for GP regression, and enhanced NFT with BO, using the notion of CoRe

$$\mathcal{Z}_{|\boldsymbol{X},\boldsymbol{\sigma}}(\kappa^2) = \left\{ \boldsymbol{x} \in \mathcal{X}; s_{|\boldsymbol{X},\boldsymbol{\sigma}}(\boldsymbol{x}, \boldsymbol{x}) \leq \kappa^2 \right\}, \quad (11)$$

i.e., the region in which the uncertainty of the GP prediction is lower than a threshold κ. Specifically, Nicoli et al. (2023a) introduced the EMICoRe acquisition function to find the best observation points, i.e., $\boldsymbol{\alpha}$ in Step 1 of NFT, such that the maximum expected improvement within CoRe is maximized. Note that the kernel parameter γ^2 controls the smoothness of the function, i.e., suppressing the interaction terms when $\gamma^2 > 1$. When $\gamma^2 = 1$, the Fourier basis (8) is orthonormal, and the VQE kernel (10) is proportional to the product of Dirichlet kernels (Rudin, 1964).

3. Subspace in Confident Region (SubsCoRe)

We propose an adaptive observation cost control method, named *subspace in confident region* (SubsCoRe), for NFT, where the notion of CoRe (Nicoli et al., 2023a) for GP plays an essential role. Specifically, in each NFT iteration t, SubsCoRe uses the minimum total number of measurement shots and distributes them optimally, so that the subspace optimization is performed with the required accuracy κ_t^2. This is guaranteed by requiring that the entire subspace $\mathcal{S}_d(\widehat{\boldsymbol{x}}) = \{\widehat{\boldsymbol{x}} + \alpha' \boldsymbol{e}_d; \alpha' \in [0, 2\pi)\}$ to be updated is a subset of the CoRe, i.e.,

$$\mathcal{S}_d(\widehat{\boldsymbol{x}}) \subseteq \mathcal{Z}_{|(\boldsymbol{X},\breve{\boldsymbol{X}}),(\boldsymbol{\sigma},\breve{\boldsymbol{\sigma}})}(\kappa_t^2), \quad (12)$$

when the GP is trained on the augmented data $(\boldsymbol{X}, \breve{\boldsymbol{X}}), (\boldsymbol{\sigma}, \breve{\boldsymbol{\sigma}})$ with new observation points $\breve{\boldsymbol{X}}, \breve{\boldsymbol{\sigma}}$. Let $\overline{N}_{\text{shot}}^t$ be the total number of shots we use in the t-th iteration. In one extreme, we can allocate each single shot to an arbitrary position, i.e., choose $\boldsymbol{\alpha} \in [0, 2\pi)^{\overline{N}_{\text{shot}}^t}$ and set $\breve{\boldsymbol{X}} = \{\widehat{\boldsymbol{x}} + \alpha_w \boldsymbol{e}_d\}_{w=1}^{\overline{N}_{\text{shot}}^t} \in \mathbb{R}^{D \times \overline{N}_{\text{shot}}^t}$ and $\breve{\boldsymbol{\sigma}} = \sigma^{*2}(1) \mathbf{1}_{\overline{N}_{\text{shot}}^t}$, where $\sigma^{*2}(1)$ is the shot noise with a single measurement shot, and $\mathbf{1}_N$ denotes the all-one vector in the N-dimensional space. Here, we note that the observation noise variance is defined as a function of N_{shots} to be

$$\sigma^{*2}(N_{\text{shots}}) = \frac{\sigma^{*2}}{N_{\text{shots}}},$$

where σ^{*2} is the observation noise variance for single-shot measurements. The optimum number of shots and their distribution are given by solving

$$\min_{\overline{N}_{\text{shot}}, \boldsymbol{\alpha} \in [0,2\pi)^{\overline{N}_{\text{shot}}^t}} \overline{N}_{\text{shot}} \text{ s.t. } \mathcal{S}_d(\widehat{\boldsymbol{x}}) \subseteq \mathcal{Z}_{|(\boldsymbol{X},\breve{\boldsymbol{X}}(\boldsymbol{\alpha})),(\boldsymbol{\sigma},\breve{\boldsymbol{\sigma}})}(\kappa_t^2). \quad (13)$$

The following theory implies that we do not need to solve such a huge ($\overline{N}_{\text{shot}} \sim 10^3$)-dimensional problem, and guides us to simple approximate solutions (the proof is given in Appendix C):

Theorem 3.1. *Assume that, for an arbitrary $\widehat{\boldsymbol{x}} \in [0, 2\pi)^D$, we observed the function values $\boldsymbol{y} \in \mathbb{R}^{1+2V_d}$ at the equidistant points $\boldsymbol{X} = \{\widehat{\boldsymbol{x}} + \alpha_w \boldsymbol{e}_d; \alpha_w = \frac{2w}{1+2V_d}\pi\}_{w=0}^{2V_d}$ along the d-th axis with the same observation noise variance σ^2. Then, the posterior variance of GP with the VQE kernel at the test point $\boldsymbol{x}' = \widehat{\boldsymbol{x}} + \alpha' \boldsymbol{e}_d$ for arbitrary $\alpha' \in [0, 2\pi)$ is*

$$s_{\boldsymbol{X}}(\boldsymbol{x}', \boldsymbol{x}') = \frac{\sigma^2 \left((\gamma^2+2V_d)^2 \frac{\sigma^2}{\sigma_0^2} + (1+2V_d)^2 \gamma^2 \right)}{\left((\gamma^2+2V_d) \frac{\sigma^2}{\sigma_0^2} + 1 + 2V_d \right) \left((\gamma^2+2V_d) \frac{\sigma^2}{\sigma_0^2} + (1+2V_d)\gamma^2 \right)}. \quad (14)$$

This theorem assumes that we do not use the previous observed points $\boldsymbol{X}$, and only use new $1 + 2V_d$ observed points (including the point at $\widehat{\boldsymbol{x}}$ or $\alpha_0 = 0$).

Notably, the posterior variance does not depend on α', and is therefore uniform along the 1D subspace. Figure 2 depicts the GP uncertainty with equidistant and non-equidistant observed points for $V_d = 1$ (left) and $V_d = 3$ (right). The left plot shows that, in terms of the worst uncertainty within the subspace, the optimal choice[¶] for the $V_d = 1$ case is not $\boldsymbol{\alpha} = (0, \pi/2, 3\pi/2)^\top = (0, \pi/2, -\pi/2)^\top$—the setting used in the original NFT (Nakanishi et al., 2020) —but the equidistant $\boldsymbol{\alpha} = (0, 2\pi/3, 4\pi/3)^\top = (0, 2\pi/3, -2\pi/3)^\top$—the setting used for the baseline NFT in Nicoli et al. (2023a).

One might have naturally expected the uniform uncertainty at the observed equidistant points $\{\widehat{\boldsymbol{x}} + \frac{2w}{1+2V_d}\pi \boldsymbol{e}_d\}_{w=0}^{2V_d}$ from the viewpoint of the Fourier analysis: GP with the VQE kernel is equivalent to Bayesian linear regression with the Fourier basis (8), and therefore the equidistant observations result in a regularized version of the discrete Fourier transform (see Appendix C.1). However, the uniform uncertainty between the equidistant points is not trivially expected, because the posterior variance (5) has a quadratic form of $\boldsymbol{K}'$, which can contain $2V_d$-th order components (i.e., twice higher frequency than the highest frequency of the trigonometric polynomial (9)), as observed in the non-equidistant

[¶]Here the "optimality" is with respect to the minimization (13) of the total number of shots that makes the entire updated subspace within the CoRe. For a fixed observation budget, the equidistant observed points are proved to give the uniform uncertainty, which is min-max optimal, i.e., it minimizes the maximum uncertainty. The minimum observation cost can be obtained by adjusting the total number of shots, while keeping the optimized observed points, so that the uniform uncertainty (14) matches the required accuracy κ. Note that this does not necessarily mean that the equidistant measurements are optimal for the overall VQE performance. We defer further analysis and a full ablation study relating the optimality to the choice of the shift to future work, e.g., by combining EMICoRe and SubsCoRe.

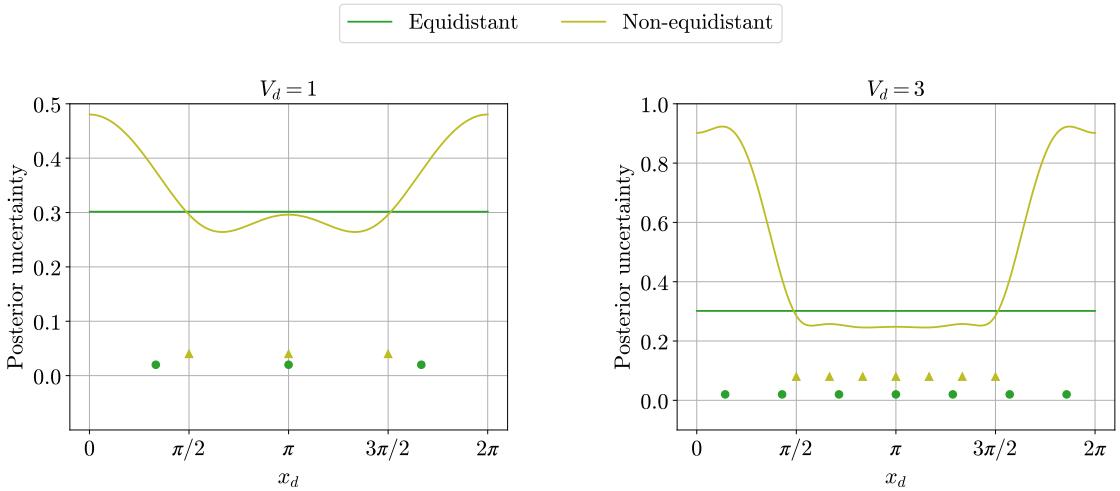

Figure 2. Uncertainty of GP with the VQE kernel trained on equidistant (darker green, round markers) and non-equidistant (olive, triangle markers) observation points. The left and right plots show the $V_d = 1$ and $V_d = 3$ cases, respectively. The green and olive solid lines refer to the GP posterior uncertainty obtained after observing $1 + 2V_d$ points with an equidistant and non-equidistant spacing. For instance, in the $V_d = 1$ case, the observations along a one-dimensional subspace (parallel to the d-axis) are performed at $x_d = \pi - \alpha, \pi, \pi + \alpha$ with α being $\frac{\pi}{2}$ (olive) and $\frac{2\pi}{3}$ (green) in the non-equidistant and equidistant cases, respectively. Following our theory, observing $1 + 2V_d$ equidistant points leads to uniform posterior uncertainty.

case in Figure 2. One might have expected that the middle points between the neighboring observed points would have higher uncertainty, a belief disproved by Theorem 3.1.

An important practical implication of Theorem 3.1 is that we can achieve uniform uncertainty—implying the min-max optimality—by observing $1 + 2V_d$ equidistant points with $\overline{N}^t_{\text{shot}}/(1 + 2V_d)$ shots each, which drastically simplifies the problem (13). Furthermore, it is crucial for GP with cubic complexity not to use too many training samples with large observation noise, although the classical computation cost is not counted in this paper. The following corollary implies an even simpler alternative.

Corollary 3.2. *The predictive uncertainty is upper-bounded by the observation noise variance, i.e.,*

$$s_{\boldsymbol{X}}(\boldsymbol{x}', \boldsymbol{x}') < \sigma^2.$$

Based on the theory above, we propose variants of SubsCoRe, all of which observe the fixed $1 + 2V_d$ equidistant points $\breve{\boldsymbol{X}} = \{\widehat{\boldsymbol{x}}^{t-1} + \alpha_w \boldsymbol{e}_d; \alpha_w = \frac{2w}{1+2V_d}\pi\}_{w=0}^{2V_d}$ with adapted observation noise variance $\breve{\boldsymbol{\sigma}} \in \mathbb{R}^{1+2V_d}_{++}$. Ignoring the rounding error, the total number of shots is proportional to $\overline{N}_{\text{shot}} \propto (\|\breve{\boldsymbol{\sigma}}\|_1)^{-1}$. Therefore, under our fixed choice of $\breve{\boldsymbol{X}}$, the SubsCoRe problem (13) reduces to

$$\max_{\breve{\boldsymbol{\sigma}} \in \mathbb{R}^{1+2V_d}_{++}} \|\breve{\boldsymbol{\sigma}}\|_1 \text{ s.t. } \mathcal{S}_d(\widehat{\boldsymbol{x}}) \subseteq \mathcal{Z}_{(\boldsymbol{X}, \breve{\boldsymbol{X}}), (\boldsymbol{\sigma}, \breve{\boldsymbol{\sigma}})}(\kappa_t^2). \quad (15)$$

The following variants give approximate feasible solutions.

SubsCoRe-Bound We set $\breve{\boldsymbol{\sigma}} = \kappa_t^2 \mathbf{1}_{1+2V_d}$, which satisfies the SubsCoRe requirement, i.e., the constraint in Eq. (15), according to Corollary 3.2.

SubsCoRe-Center We set $\breve{\boldsymbol{\sigma}} = (\breve{\sigma}_0^2, \breve{\sigma}^2, \ldots, \breve{\sigma}^2)^\top$, i.e., we tie the entries except for the variance $\breve{\sigma}_0^2$ at the previous best point $\widehat{\boldsymbol{x}}^{t-1}$, and solve the SubsCoRe problem (15) by a 2D search with respect to $\breve{\sigma}_0^2$ and $\breve{\sigma}^2$.

SubsCoRe-Bound ignores the existence of previous training data $\boldsymbol{X}$, and therefore gives a loose bound. On the other hand, SubsCoRe-Center makes use of the fact that GP already has low uncertainty ($\leq \kappa_{t-1}^2$) at the current best point $\widehat{\boldsymbol{x}}^{t-1}$ after the previous iteration, and adjust the observation accuracy $\breve{\sigma}_0^2$ at $\widehat{\boldsymbol{x}}^{t-1}$ and the accuracy $\breve{\sigma}^2$ at the other points separately. Therefore, SubsCoRe-Center is more efficient than SubsCoRe-Bound. The difference in efficiency is expected to be more prominent in the converging phase, where the best point moves only slightly in each iteration, and therefore, the GP can already posses a low uncertainty over the subspace to be updated before new observations are acquired.

Once we specify the necessary observation noise $\breve{\boldsymbol{\sigma}}$, the number of shots is chosen such that $\sigma^{*2}(N_{\text{shot}}) = \breve{\sigma}_w^2$ for each observed point. Below, we describe the whole procedure of SubsCoRe-Center, which equips NFT with the adaptive number of shots control. For further details on the algorithm we defer the reader to Appendix D.2. Throughout the manuscript, SubsCoRe will always implicitly refer to SubsCoRe-Center unless specified otherwise.

SubsCoRe Algorithm SubsCoRe is initialized with a random point $\boldsymbol{X}^0 = (\widehat{\boldsymbol{x}}^0)$ and a first observation $\boldsymbol{y}^0 = \widehat{y}^0 =$

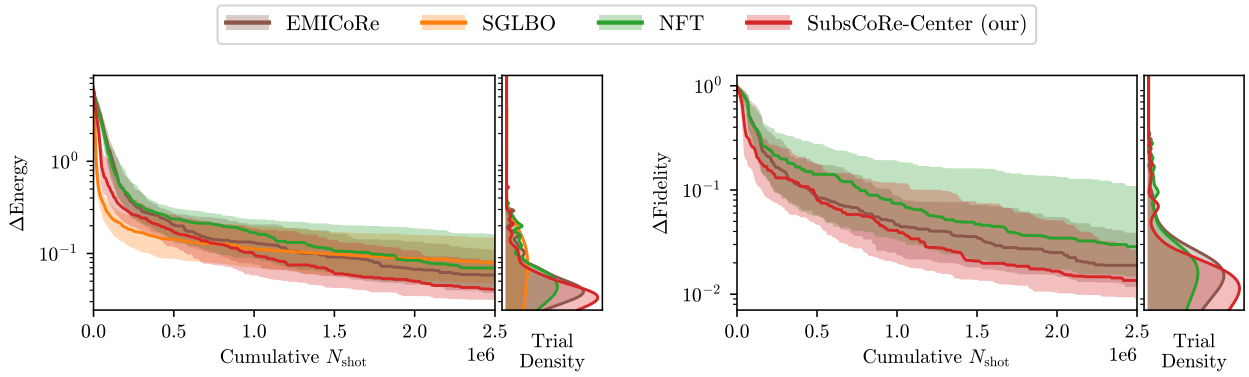

Figure 3. Energy (left) and fidelity (right), as the difference from the ground-truth (see Eqs. (18) and (19)), achieved by our SubsCoRe-Center and the baselines, NFT, SGLBO, and EMICoRe for the Ising model with $(Q, L) = (5, 3)$. Both plots are on a logarithmic scale, and the horizontal axis indicates the cumulative number of shots (per operator group) as the total quantum computation cost. The fidelity by SGLBO is not shown since the original code does not store the optimal parameters $\widehat{\boldsymbol{x}}$ to reproduce the quantum state $|\psi_{\widehat{\boldsymbol{x}}}\rangle$. However, given its slower convergence in terms of energy (left), we expect that the achieved fidelity is also worse than our SubsCoRe-Center.

$f^*(\widehat{\boldsymbol{x}}^0) + \varepsilon_0$ with observation noise $\breve{\sigma}^2 = \kappa_0^2$. Then, it iterates the following procedure: in each iteration step t,

1. Select an axis $d \in \{1, \ldots, D\}$ sequentially and observe the objective $\breve{\boldsymbol{y}} \in \mathbb{R}^{1+2V_d}$ at the $1 + 2V_d$ equidistant points $\breve{\boldsymbol{X}} = \{\widehat{\boldsymbol{x}}^{t-1} + \alpha_w \boldsymbol{e}_d; \alpha_w = \frac{2w}{1+2V_d}\pi\}_{w=0}^{2V_d}$ along the d-th axis. Here, the observation variances $\breve{\boldsymbol{\sigma}}$ are determined by SubsCoRe-Center (or SubsCoRe-Bound), using the CoRe threshold κ_t^2, as well as the training data $\boldsymbol{X}^{t-1}, \boldsymbol{\sigma}^{t-1}$ up to the previous iteration.
2. Train GP with updated training data $\boldsymbol{X}^t = (\boldsymbol{X}^{t-1}, \breve{\boldsymbol{X}}), \boldsymbol{\sigma}^t = (\boldsymbol{\sigma}^{t-1}, \breve{\boldsymbol{\sigma}}), \boldsymbol{y}^t = (\boldsymbol{y}^{t-1}, \breve{\boldsymbol{y}})$.
3. Analytically find the minimum of the GP mean
$$\widehat{\boldsymbol{x}}^t = \operatorname*{argmin}_{\boldsymbol{x}'} \mu_{|\boldsymbol{X}^t, \boldsymbol{\sigma}^t, \boldsymbol{y}^t}(\boldsymbol{x}')$$
in the 1D subspace $\boldsymbol{x}' = \widehat{\boldsymbol{x}}^{t-1} + \alpha' \boldsymbol{e}_d$ by applying the trigonometric polynomial regression to the GP predictions at $1 + 2V_d$ equidistant points.
4. Update the best score by $\widehat{y}^t = \mu_{|\boldsymbol{X}^t, \boldsymbol{\sigma}^t, \boldsymbol{y}^t}(\widehat{\boldsymbol{x}}^t)$.
5. Update the CoRe threshold for the next iteration by
$$\kappa^{t+1} = \max(C_0, -C_1 \cdot \mathrm{Slope}(\{\widehat{y}^{t'}\}_{t'=t-T_{\mathrm{Ave}}}^{t})). \tag{16}$$

In Eq. (16), $\mathrm{Slope}(\cdot)$ estimates the progress of optimization by linear regression to the T_{Ave} recent best values, and $C_0, C_1 \geq 0$ are the hyperparameters controlling the lower bound and linear relation between κ and the slope, respectively. In our experiments we set C_0 so that the number of shots is at most 1024 per data point (and per operator group), which is a typical value achievable on current quantum hardwares. Upper-bounding the number of shots allows us to keep the observation cost under control in the later stage of optimization, where the energy improvement, and thus the threshold κ, tends to be small. We set $C_1 = 1$.

4. Experiment

4.1. Setup

We demonstrate the performance of our adaptive cost control method, following the same experimental setup as in Nicoli et al. (2023a). Our Python implementation based on Qiskit (Abraham et al., 2019), one of the most widely used library for developing and simulating quantum programs, is available at https://github.com/angler-vqe/subscore.

Hamiltonian and Quantum Circuit We focus on the quantum Heisenberg Hamiltonian with open boundary conditions,

$$H = -\sum_{i \in \{X,Y,Z\}} \left[\sum_{j=1}^{Q-1} (J_i \sigma_j^i \sigma_{j+1}^i) + \sum_{j=1}^{Q} h_i \sigma_j^i \right], \tag{17}$$

where $\{\sigma_j^i\}_{i \in \{X,Y,Z\}}$ are the Pauli operators acting on the j-th qubit. The Heisenberg Hamiltonian is a widely-studied benchmark for assessing VQE performance (Tilly et al., 2022) because it is of high practical relevance—for example, many lattice field theories can be represented as generalized spin chains (Di Meglio et al., 2023; Funcke et al., 2023) —and the true ground-state wave function can be analytically computed for a small number Q of qubits. For the quantum circuit, we adopt the commonly used L-layered `Efficient SU(2)` circuit with open boundary conditions (see Nicoli et al. (2023a) for more details).

Evaluation Metrics We evaluate our methods with two metrics: the best achieved *true energy* $f^*(\widehat{\boldsymbol{x}})$ for $f^*(\cdot)$ defined in Eq. (6), and the *fidelity* $\langle \psi_{\mathrm{GS}} | \psi_{\widehat{\boldsymbol{x}}} \rangle \in [0, 1]$, which is the inner product between the true ground-state wave function $|\psi_{\mathrm{GS}}\rangle$ and the wave function $|\psi_{\widehat{\boldsymbol{x}}}\rangle$ corresponding to the optimized parameters $\widehat{\boldsymbol{x}}$. For both metrics, we plot the

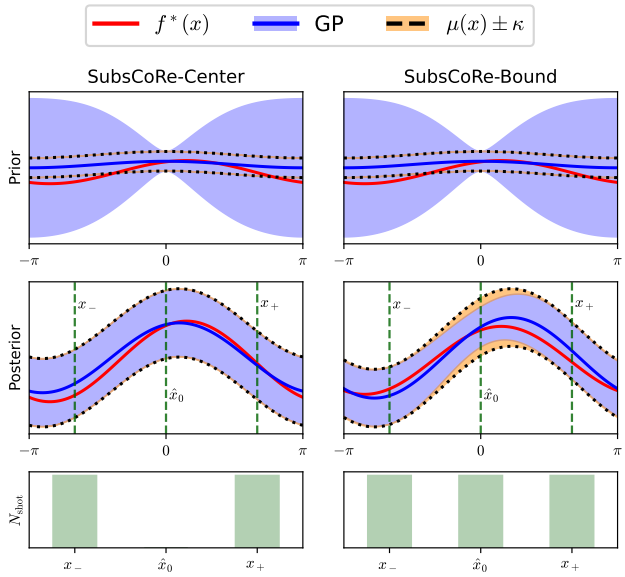

Figure 4. Prior (top) and posterior (middle) GP with SubsCoRe-Center (left) and SubsCoRe-Bound (right) in an SMO step. The red solid lines are the true function $f^*(\boldsymbol{x})$, and the blue solid lines and shadows are the GP mean and uncertainty, respectively. The vertical lines represent the previous best point $\widehat{x}_0$ and the equidistant shifts ($x_\pm$) as visualized in the cartoon in Figure 1. The bottom row reports the number of shots required by SubsCoRe-Center and SubsCoRe-Bound at each observed point. Before the new observations (i.e., prior), the previous best point $\widehat{x}_0$ is already in the CoRe, i.e., the uncertainty of GP is within the CoRe requirement (shown as black dashed lines). After the new observations (i.e., posterior), the entire subspace is in the CoRe. We see the gap (highlighted in orange) between the CoRe requirement and the posterior uncertainty with SubsCoRe-Bound. This is because SubsCoRe-Bound relies on a looser bound and assigns many shots to the previous best point $\widehat{x}_0$, on which the prior GP was already trained accurately (see bottom-right plot).

difference (the smaller is the better) from the ground-truth optimum, i.e.,

$$\Delta\text{Energy} = f^*(\widehat{\boldsymbol{x}}) - \langle\psi_{\text{GS}}|H|\psi_{\text{GS}}\rangle, \tag{18}$$

$$\begin{aligned}\Delta\text{Fidelity} &= \langle\psi_{\text{GS}}|\psi_{\text{GS}}\rangle - \langle\psi_{\text{GS}}|\psi_{\widehat{\boldsymbol{x}}}\rangle \\ &= 1 - \langle\psi_{\text{GS}}|\psi_{\widehat{\boldsymbol{x}}}\rangle,\end{aligned} \tag{19}$$

in the log scale. Here, $|\psi_{\text{GS}}\rangle$ and $\langle\psi_{\text{GS}}|H|\psi_{\text{GS}}\rangle$ are the ground-state wave function and the true ground-state energy, respectively, both of which are computed analytically. As the quantum computation cost, we count the total number of measurement shots *per operator group* (see the footnote 2 in Section 2.2), i.e., the cumulative sum of N_{shot} over all observations in the whole optimization process.

Baseline Methods We compare our SubsCoRe approach to the state-of-the-art baselines, NFT (Nakanishi et al., 2020) (with $\alpha = \pm\frac{2\pi}{3}$ shifts), SGLBO (Tamiya & Yamasaki, 2022), and EMICoRe (Nicoli et al., 2023a). SGLBO is a gradient-based method equipped with adaptive shot control, while NFT and EMICoRe are SMO-based approaches for which no adaptive shot control technique was available before our approach. Therefore, we evaluate NFT and EMICoRe with a fixed number of shots $N_{\text{shot}} = 1024$, i.e., the same setting as in Nicoli et al. (2023a).

Further details on the algorithms, including the parameter setting, and the experiments are given in Appendices D and E, respectively.

4.2. Performance Evaluation

We evaluate our SubsCoRe and the baseline methods on the Ising Hamiltonian, a special case of the Heisenberg Hamiltonian (17) with the coupling parameters set to $J_X = -1$, $J_Y = J_Z = 0, h_X = h_Y = 0$, $h_Z = -1$, with the $(L = 3)$-layered quantum circuit with $(Q = 5)$-qubits. Figure 3 shows the achieved energy (18) and the fidelity (19)—as the difference from the ground truth—with the cumulative number of measurement shots in the horizontal axis. On the right-hand side of each plot, we show the trial density after $2.5 \cdot 10^6$ measurement shots are used. We applied the kernel density estimation over the 100 independent seeded trials to produce the trial density. We observe that SubsCoRe-Center outperforms the baselines with statistical significance (p-value < 0.05 according to the Wilcoxon signed-rank test), thus proving the usefulness of our adaptive cost control for SMO. We show the evolution of the number of measurement shots taken by SubsCoRe-Center as a function of the SMO steps in Appendix E.2.

Figure 4 shows the trained GP with SubsCoRe-Center (left) and SubsCoRe-Bound (right), prior (top) and posterior (middle) to the acquisition of the new observations for an SMO step. We see that the posterior uncertainty with SubsCoRe-Center is almost uniform and just inside the CoRe requirement, i.e., $\mu(\boldsymbol{x}) \pm \kappa$. Crucially, fine-tuning the number of measurement shots (bottom) allows not to consume shots at the previous best point $\widehat{x}_0$, which was already in the CoRe for the prior GP (see the top plot). On the other hand, SubsCoRe-Bound, which relies on a looser bound, allocates shots also to the previous best point, making some regions too accurate (highlighted by orange). This explains the inferior efficiency of SubsCoRe-Bound, which will be shown in Figure 5 in Section 4.3.

4.3. Discussion

GP vs. Suffix Averaging The SGLBO baseline is not only equipped with adaptive shot control but also with *suffix averaging* (SA), i.e., it takes the average of the best point trail over the last, e.g., 10% iterations, i.e., $\widehat{\boldsymbol{x}} = \frac{1}{0.1T}\sum_{t=0.9T}^{T}\widehat{\boldsymbol{x}}^t$. The idea behind SA is that, in the converging phase, the best point can fluctuate around the optimum because of the noisy observations, and therefore,

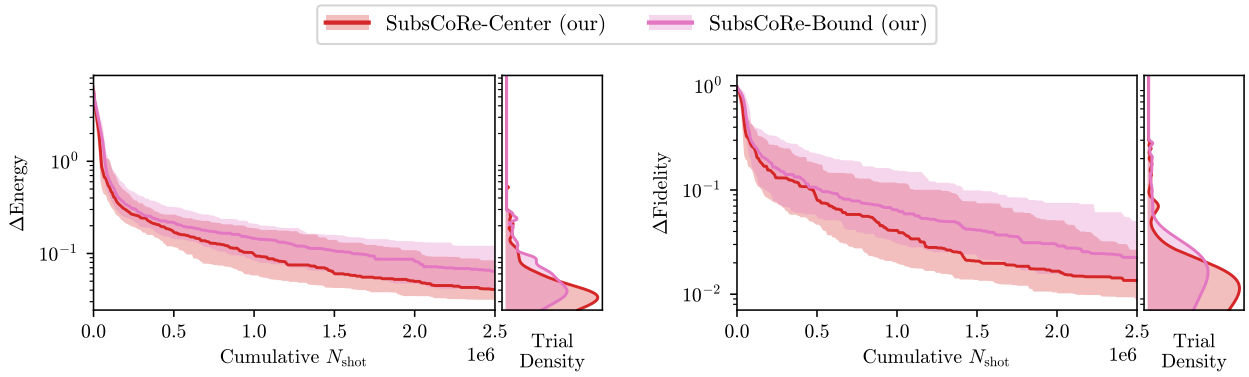

Figure 5. Energy (left) and fidelity (right), as the difference from the ground-truth (see Eqs. (18) and (19)), achieved by SubsCoRe-Center and SubsCoRe-Bound. The inferior efficiency of SubsCoRe-Bound is due to the fact that it relies on a looser bound.

taking the average over the optimization steps can better align the best point to the true optimum. Significant improvement by using SA was reported in Tamiya & Yamasaki (2022). We argue that our SubsCoRe enjoys this effect in a statistically more pronounced way as it uses the GP surrogate trained not only on the new observations but also on those from the previous steps. We emphasize that, while SGLBO also uses a GP surrogate, it is not trained on previous observations, and therefore, previous observations are only leveraged through the SA heuristics.

SubsCoRe-Center vs. SubsCoRe-Bound As discussed in Section 4.2, the SubsCoRe-Bound variant does not consider the contribution from the previous observations. A consequence can be observed in Figure 5, where SubsCoRe-Bound is outperformed by SubsCoRe-Center, and the difference increases in the later phase of the optimization.

5. Conclusion

The intrinsic probabilistic nature of measurements is one of the major drawbacks of quantum algorithms, to which noise-resilient statistical techniques are expected to make significant contributions. Our *subspace in confident region* (SubsCoRe) approach for variational quantum eigensolvers (VQEs) has shown to improve the optimization efficiency by dynamically controlling the observation costs, and thus reducing the quantum computation budget without sacrificing the accuracy of the optimization. Here, the *uncertainty estimation*, a key implement in Bayesian statistics, plays an essential role. Specifically, harnessing the notion of the confident region (CoRe) of Gaussian processes (GP), SubsCoRe minimizes the observation costs while guaranteeing the required accuracy for prediction. Our theory, built upon the unique physical property of VQEs and the analytic form of GP regression, proved the uniform uncertainty of GP predictions trained on equidistant observations, leading to simple algorithms that approximate the optimal cost distribution. The developed SubsCoRe outperforms the state-of-the-art methods with and without adaptive cost control. Our analysis also clarified the relation between GP with the VQE kernel and the Fourier analysis, which may facilitate further development of efficient VQE algorithms. In future work, we plan to run experiments on real quantum devices, and thus benchmark our approaches in the presence of quantum hardware noise. Another future plan is to apply our methods to different Hamiltonians with high practical relevance in quantum chemistry and lattice field theory.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning and Quantum Computing. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

Acknowledgements

The authors thank the reviewers for their constructive comments and discussion for improving the paper. This work was supported by the German Ministry for Education and Research (BMBF) under the grant BIFOLD24B, the Einstein Research Unit Quantum Project (ERU-2020-607), the European Union's HORIZON MSCA Doctoral Networks programme project AQTIVATE (101072344), the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) as part of the CRC 1639 NuMeriQS – project no. 511713970, the European Union's Horizon Europe Framework Programme (HORIZON) under the ERA Chair scheme with grant agreement no. 101087126, and the Ministry of Science, Research and Culture of the State of Brandenburg within the Centre for Quantum Technologies and Applications (CQTA).

References

Abraham, H. et al. Qiskit: An Open-source Framework for Quantum Computing. *Zenodo*, 2019. doi: 10.5281/zenodo.2562111.

Bauer, B., Bravyi, S., Motta, M., and Chan, G. K.-L. Quantum algorithms for quantum chemistry and quantum materials science. *Chemical Reviews*, 120(22):12685–12717, 2020.

Bluvstein, D., Evered, S. J., Geim, A. A., Li, S. H., Zhou, H., Manovitz, T., Ebadi, S., Cain, M., Kalinowski, M., Hangleiter, D., et al. Logical quantum processor based on reconfigurable atom arrays. *Nature*, pp. 1–3, 2023.

Cao, Y., Romero, J., and Aspuru-Guzik, A. Potential of quantum computing for drug discovery. *IBM Journal of Research and Development*, 62(6):6–1, 2018.

Debnath, S., Linke, N. M., Figgatt, C., Landsman, K. A., Wright, K., and Monroe, C. Demonstration of a small programmable quantum computer with atomic qubits. *Nature*, 536(7614):63–66, 2016. doi: 10.1038/nature18648. URL https://doi.org/10.1038/nature18648.

Di Meglio, A. et al. Quantum Computing for High-Energy Physics: State of the Art and Challenges. Summary of the QC4HEP Working Group, 7 2023.

Fedorov, A. and Gelfand, M. Towards practical applications in quantum computational biology. *Nature Computational Science*, 1(2):114–119, 2021.

Funcke, L., Hartung, T., Jansen, K., and Kühn, S. Review on Quantum Computing for Lattice Field Theory. *PoS*, LATTICE2022:228, 2023. doi: 10.22323/1.430.0228.

Grimsley, H. R., Economou, S. E., Barnes, E., and Mayhall, N. J. An adaptive variational algorithm for exact molecular simulations on a quantum computer. *Nature communications*, 10(1):3007, 2019.

Humble, T. S., Delgado, A., Pooser, R., Seck, C., Bennink, R., Leyton-Ortega, V., Wang, C. C. J., Dumitrescu, E., Morris, T., Hamilton, K., Lyakh, D., Date, P., Wang, Y., Peters, N. A., Evans, K. J., Demarteau, M., McCaskey, A., Nguyen, T., Clark, S., Reville, M., Di Meglio, A., Grossi, M., Vallecorsa, S., Borras, K., Jansen, K., and Krücker, D. Snowmass white paper: Quantum computing systems and software for high-energy physics research, March 2022.

Iannelli, G. and Jansen, K. Noisy bayesian optimization for variational quantum eigensolvers. *ArXiv e-prints*, 2021.

Kandala, A., Mezzacapo, A., Temme, K., Takita, M., Brink, M., Chow, J. M., and Gambetta, J. M. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *nature*, 549(7671):242–246, 2017.

Kielpinski, D., Monroe, C., and Wineland, D. J. Architecture for a large-scale ion-trap quantum computer. *Nature*, 417(6890):709–711, 2002. doi: 10.1038/nature00784. URL https://doi.org/10.1038/nature00784.

Makarova, A., Usmanova, I., Bogunovic, I., and Krause, A. Risk-averse heteroscedastic bayesian optimization. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 17235–17245. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/8f97d1d7e02158a83ceb2c14ff5372cd-Paper.pdf.

McClean, J. R., Romero, J., Babbush, R., and Aspuru-Guzik, A. The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics*, 18(2):023023, 2016.

Mickaël Binois, Jiangeng Huang, R. B. G. and Ludkovski, M. Replication or exploration? sequential design for stochastic simulation experiments. *Technometrics*, 61(1):7–23, 2019. doi: 10.1080/00401706.2018.1469433. URL https://doi.org/10.1080/00401706.2018.1469433.

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

Nakanishi, K. M., Fujii, K., and Todo, S. Sequential minimal optimization for quantum-classical hybrid algorithms. *Phys. Rev. Res.*, 2:043158, 2020. doi: 10.1103/PhysRevResearch.2.043158. URL https://link.aps.org/doi/10.1103/PhysRevResearch.2.043158.

Nicoli, K., Anders, C. J., Funcke, L., Hartung, T., Jansen, K., Kühn, S., Müller, K.-R., Stornati, P., Kessel, P., and Nakajima, S. Physics-informed bayesian optimization of variational quantum circuits. In *Advances in Neural Information Processing Systems (NeurIPS2023)*, 2023a.

Nicoli, K. A., Anders, C. J., et al. Emicore: Expected maximum improvement over confident regions. https://github.com/angler-vqe/emicore, 2023b.

Ollitrault, P. J., Kandala, A., Chen, C.-F., Barkoutsos, P. K., Mezzacapo, A., Pistoia, M., Sheldon, S., Woerner, S.,

Gambetta, J. M., and Tavernelli, I. Quantum equation of motion for computing molecular excitation energies on a noisy quantum processor. *Phys. Rev. Res.*, 2:043140, Oct 2020. doi: 10.1103/PhysRevResearch.2.043140. URL https://link.aps.org/doi/10.1103/PhysRevResearch.2.043140.

Peruzzo, A., McClean, J., Shadbolt, P., et al. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1):4213, 2014. doi: 10.1038/ncomms5213. URL https://doi.org/10.1038/ncomms5213.

Platt, J. Sequential minimal optimization : A fast algorithm for training support vector machines. *Microsoft Research Technical Report*, 1998.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, USA, 2006.

Rudin, W. *Principles of Mathematical Analysis*. McGraw-Hill, 1964.

Schuld, M., Bergholm, V., Gogolin, C., et al. Evaluating analytic gradients on quantum hardware. *Phys. Rev. A*, 99(3):032331, 2019.

Tamiya, S. and Yamasaki, H. Stochastic gradient line bayesian optimization for efficient noise-robust optimization of parameterized quantum circuits. *npj Quantum Information*, 8(1):90, 2022. doi: 10.1038/s41534-022-00592-6. URL https://doi.org/10.1038/s41534-022-00592-6.

Tilly, J., Chen, H., Cao, S., Picozzi, D., Setia, K., Li, Y., Grant, E., Wossnig, L., Rungger, I., Booth, G. H., et al. The variational quantum eigensolver: a review of methods and best practices. *Physics Reports*, 986:1–128, 2022.

A. Limitations

The limitations of our approach are inherently related to those of VQEs. In fact, applications of the proposed method on current quantum devices are mostly obstructed by hardware limitations. In addition to the probabilistic nature of the measurement process, which we tackled in this paper, other types of noise resulting from imperfect isolation from the environment and limited experimental control still significantly affect current quantum hardware. In light of this, the SubsCoRe approach we proposed in this paper in principle does not suffer from crucial limitations per se, but indeed will also be affected by the same bottlenecks of VQEs. Nonetheless, we envision that leveraging machine learning techniques and Bayesian statistics, in particular BO and GP regression in this context, is particularly promising as it might allow us to cope better with the inherent probabilistic nature of such hybrid quantum-classical algorithms.

B. Theoretical Details

C. Proofs of Theorem 3.1 and Corollary 3.2

We use basic properties of the discrete Fourier transform, derived by the *root of unity*, i.e., $\xi_L = e^{2\pi i/L}$. Since

$$\sum_{l=0}^{L-1} \xi_L^{lw} = \frac{1-\xi_L^{wL}}{1-\xi_L^{w}} = 0 \qquad \text{for} \qquad w = 1, \dots, L-1,$$

we have

$$\sum_{v=0}^{2V_d} e^{vw\frac{2\pi i}{1+2V_d}} = 0 \qquad \text{for} \qquad w = 1, \dots, 2V_d.$$

Comparing the real and imaginary parts, we have

$$\sum_{v=0}^{2V_d} \cos\left(vw\frac{2\pi}{1+2V_d}\right) = \begin{cases} 1+2V_d & \text{for} \quad w=0, \\ 0 & \text{for} \quad w = 1, \dots, 2V_d, \end{cases} \tag{20}$$

$$\sum_{v=0}^{2V_d} \sin\left(vw\frac{2\pi}{1+2V_d}\right) = 0 \qquad \text{for} \qquad w = 0, \dots, 2V_d. \tag{21}$$

From the symmetry,

$$\sum_{v=1}^{V_d} \cos\left(vw\frac{2\pi}{1+2V_d}\right) = \sum_{v=V_d+1}^{2V_d} \cos\left(vw\frac{2\pi}{1+2V_d}\right), \tag{22}$$

and thus

$$1 + 2\sum_{v=1}^{V_d} \cos\left(vw\frac{2\pi}{1+2V_d}\right) = 0 \qquad \text{for} \qquad w = 1, \dots, 2V_d. \tag{23}$$

By using the basic equations (20)–(23), we will derive the predictive variance of GP in the following.

The kernel matrix for the training points $\{\widehat{\boldsymbol{x}} + \frac{2w}{1+2V_d}\pi \boldsymbol{e}_d\}_{w=0}^{2V_d}$ is Toeplitz as

$$\boldsymbol{K} = \sigma_0^2 \begin{pmatrix} \tau_0 & \tau_1 & \tau_2 & \cdots & \tau_{2V_d-1} & \tau_{2V_d} \\ \tau_1 & \tau_0 & \tau_1 & & & \tau_{2V_d-1} \\ \tau_2 & \tau_1 & \tau_0 & & & \\ \vdots & & & \ddots & & \\ \tau_{2V_d-1} & & & & \tau_0 & \tau_1 \\ \tau_{2V_d} & \tau_{2V_d-1} & & & \tau_1 & \tau_0 \end{pmatrix} \in \mathbb{R}^{(1+2V_d)\times(1+2V_d)},$$

where

$$\tau_w = \frac{\gamma^2 + 2\sum_{v=1}^{V_d} \cos\left(\frac{2vw}{1+2V_d}\pi\right)}{\gamma^2 + 2V_d}.$$

The test kernel components for a test point at $\boldsymbol{x}' = \widehat{\boldsymbol{x}} + \alpha' \boldsymbol{e}_d$ for arbitrary $\alpha' \in [0, 2\pi)$ are

$$\boldsymbol{k}' = \sigma_0^2 \begin{pmatrix} \rho_0 \\ \rho_1 \\ \vdots \\ \rho_{2V_d} \end{pmatrix},$$
$$k'' = \sigma_0^2,$$

where

$$\rho_w = \frac{\gamma^2 + 2\sum_{v=1}^{V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right)}{\gamma^2 + 2V_d}.$$

Eq. (23) implies that

$$\tau_w = \begin{cases} 1 & \text{for} \quad w = 0, \\ \frac{\gamma^2-1}{\gamma^2+2V_d} & \text{for} \quad w = 1, \ldots, 2V_d, \end{cases}$$

and therefore

$$\boldsymbol{K} = \sigma_0^2 \left(\frac{1+2V_d}{\gamma^2+2V_d} \boldsymbol{I}_{1+2V_d} + \frac{\gamma^2-1}{\gamma^2+2V_d} \mathbf{1}_{1+2V_d} \mathbf{1}_{1+2V_d}^\top \right),$$

where $\boldsymbol{I}_N$ and $\mathbf{1}_N$ denote the N-dimensional identity matrix and the N-dimensional vector with all entries equal to one, respectively. The matrix inversion lemma gives

$$\begin{aligned}
&\left(\boldsymbol{K} + \sigma^2 \boldsymbol{I}_{1+2V_d}\right)^{-1} \\
&\quad = \frac{1}{\sigma_0^2} \left(\left(\sigma^2/\sigma_0^2 + \frac{1+2V_d}{\gamma^2+2V_d} \right) \boldsymbol{I}_{1+2V_d} + \frac{\gamma^2-1}{\gamma^2+2V_d} \mathbf{1}_{1+2V_d} \mathbf{1}_{1+2V_d}^\top \right)^{-1} \\
&\quad = \frac{1}{\sigma_0^2} \left(\sigma^2/\sigma_0^2 + \frac{1+2V_d}{\gamma^2+2V_d} \right)^{-1} \left(\boldsymbol{I}_{1+2V_d} + \frac{\gamma^2-1}{(\gamma^2+2V_d)\sigma^2/\sigma_0^2 + 1 + 2V_d} \mathbf{1}_{1+2V_d} \mathbf{1}_{1+2V_d}^\top \right)^{-1} \\
&\quad = \frac{1}{\sigma_0^2} \left(\sigma^2/\sigma_0^2 + \frac{1+2V_d}{\gamma^2+2V_d} \right)^{-1} \\
&\qquad \left(\boldsymbol{I}_{1+2V_d} - \frac{\gamma^2-1}{(\gamma^2+2V_d)\sigma^2/\sigma_0^2 + 1 + 2V_d} \mathbf{1}_{1+2V_d} \left(1 + \frac{(1+2V_d)(\gamma^2-1)}{(\gamma^2+2V_d)\sigma^2/\sigma_0^2 + 1 + 2V_d} \right)^{-1} \mathbf{1}_{1+2V_d}^\top \right) \\
&\quad = \frac{1}{\sigma_0^2} \frac{\gamma^2+2V_d}{(\gamma^2+2V_d)\sigma^2/\sigma_0^2 + 1 + 2V_d} \left(\boldsymbol{I}_{1+2V_d} - \left(\frac{(\gamma^2+2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2}{\gamma^2-1} \right)^{-1} \mathbf{1}_{1+2V_d} \mathbf{1}_{1+2V_d}^\top \right) \\
&\quad = \frac{1}{\sigma_0^2} a (\boldsymbol{I}_{1+2V_d} + b \mathbf{1}_{1+2V_d} \mathbf{1}_{1+2V_d}^\top),
\end{aligned} \tag{24}$$

where

$$a = \frac{\gamma^2+2V_d}{(\gamma^2+2V_d)\sigma^2/\sigma_0^2 + 1 + 2V_d},$$
$$b = -\frac{\gamma^2-1}{(\gamma^2+2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2}.$$

By using Eqs. (20) and (21), we have

$$
\begin{aligned}
\|\boldsymbol{k}'\|^2 &= \sigma_0^4 \sum_{w=0}^{2V_d} \rho_w^2 \\
&= \sigma_0^4 \sum_{w=0}^{2V_d} \left(\frac{\gamma^2 + 2\sum_{v=1}^{V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right)}{\gamma^2 + 2V_d} \right)^2 \\
&= \frac{\sigma_0^4}{(\gamma^2+2V_d)^2} \sum_{w=0}^{2V_d} \left\{ \gamma^4 + 4\gamma^2 \sum_{v=1}^{V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right) \right. \\
&\qquad \left. +4 \sum_{v=1}^{V_d} \sum_{v'=1}^{V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right) \cos\left(v'\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right) \right\} \\
&= \frac{\sigma_0^4}{(\gamma^2+2V_d)^2} \sum_{w=0}^{2V_d} \left\{ \gamma^4 + 4\gamma^2 \sum_{v=1}^{V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right) \right. \\
&\qquad \left. +2 \sum_{v=1}^{V_d} \sum_{v'=1}^{V_d} \left(\cos\left((v+v')\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right) + \cos\left((v-v')\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right)\right) \right\} \\
&= \frac{\sigma_0^4}{(\gamma^2+2V_d)^2} \sum_{w=0}^{2V_d} \left\{ \gamma^4 + 4(\gamma^2-1) \sum_{v=1}^{V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right) \right. \\
&\qquad \left. + 2 \sum_{v=1}^{V_d} \sum_{v'=0}^{V_d} \left(\cos\left((v+v')\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right) + \cos\left((v-v')\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right)\right) \right\} \\
&= \frac{\sigma_0^4}{(\gamma^2+2V_d)^2} \left\{ \gamma^4(1+2V_d) + 4(\gamma^2-1) \sum_{v=1}^{V_d} \sum_{w=0}^{2V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right) \right. \\
&\qquad \left. + 2 \sum_{v=1}^{V_d} \sum_{v'=0}^{V_d} \sum_{w=0}^{2V_d} \left(\cos\left((v+v')\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right) + \cos\left((v-v')\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right)\right) \right\} \\
&= \frac{\sigma_0^4}{(\gamma^2+2V_d)^2} \left\{ \gamma^4(1+2V_d) + 4(\gamma^2-1) \sum_{v=1}^{V_d} \sum_{w=0}^{2V_d} \cos\frac{2wv\pi}{1+2V_d} \cos(v\alpha') \right. \\
&\qquad + 2 \sum_{v=1}^{V_d} \sum_{v'=0}^{V_d} \sum_{w=0}^{2V_d} \left(\cos\frac{2w(v+v')\pi}{1+2V_d} \cos((v+v')\alpha') + \sin\frac{2w(v+v')\pi}{1+2V_d} \sin((v+v')\alpha') \right. \\
&\qquad \left. \left. + \cos\frac{2w(v-v')\pi}{1+2V_d} \cos((v-v')\alpha') + \sin\frac{2w(v-v')\pi}{1+2V_d} \sin((v-v')\alpha') \right) \right\} \\
&= \frac{\sigma_0^4}{(\gamma^2+2V_d)^2} \left\{ \gamma^4(1+2V_d) + 2\sum_{v=1}^{V_d} (1+2V_d) \right\} \\
&= \frac{\sigma_0^4}{(\gamma^2+2V_d)^2} \left\{ \gamma^4(1+2V_d) + 2V_d(1+2V_d) \right\} \\
&= \sigma_0^4 \frac{(1+2V_d)(\gamma^4+2V_d)}{(\gamma^2+2V_d)^2},
\end{aligned} \tag{25}
$$

and

$$
\|\boldsymbol{k}'\|_1 = \boldsymbol{k}'^{\top} \mathbf{1}_{1+2V_d} = \sigma_0^2 \sum_{w=0}^{2V_d} \frac{\gamma^2 + 2\sum_{v=1}^{V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right)}{\gamma^2 + 2V_d}
$$

$$
\begin{aligned}
&= \sigma_0^2 \frac{\gamma^2(1+2V_d) + 2\sum_{v=1}^{V_d}\sum_{w=0}^{2V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right)}{\gamma^2 + 2V_d} \\
&= \sigma_0^2 \frac{\gamma^2(1+2V_d) + 2\sum_{v=1}^{V_d}\sum_{w=0}^{2V_d} \left(\cos\frac{2wv\pi}{1+2V_d}\cos v\alpha' + \sin\frac{2wv\pi}{1+2V_d}\sin v\alpha'\right)}{\gamma^2 + 2V_d} \\
&= \sigma_0^2 \frac{(1+2V_d)\gamma^2}{\gamma^2 + 2V_d}. \qquad (26)
\end{aligned}
$$

With Eqs. (24)–(26), the predicitive variance is computed as

$$
\begin{aligned}
s(\boldsymbol{x}', \boldsymbol{x}') &= k'' - \boldsymbol{k}'^\top \left(\boldsymbol{K} + \sigma^2 \boldsymbol{I}_{1+2V_d}\right)^{-1} \boldsymbol{k}' \\
&= \sigma_0^2 - \boldsymbol{k}'^\top \frac{1}{\sigma_0^2} a \left(\boldsymbol{I}_{1+2V_d} + b\mathbf{1}_{1+2V_d}\mathbf{1}_{1+2V_d}^\top\right) \boldsymbol{k}' \\
&= \sigma_0^2 - \frac{1}{\sigma_0^2} a \left(\|\boldsymbol{k}'\|^2 + b(\boldsymbol{k}'^\top \mathbf{1}_{1+2V_d})^2\right) \\
&= \sigma_0^2 - \frac{1}{\sigma_0^2} a \left(\sigma_0^4 \frac{(1+2V_d)\left(\gamma^4 + 2V_d\right)}{(\gamma^2+2V_d)^2} + b\sigma_0^4 \frac{(1+2V_d)^2\gamma^4}{(\gamma^2+2V_d)^2}\right) \\
&= \sigma_0^2 \frac{(\gamma^2+2V_d)^2 - a\left((1+2V_d)\left(\gamma^4+2V_d\right) + \frac{-(1+2V_d)^2\gamma^4(\gamma^2-1)}{(\gamma^2+2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2}\right)}{(\gamma^2+2V_d)^2} \\
&= \sigma_0^2 \frac{(\gamma^2+2V_d)^2 - \frac{\gamma^2+2V_d}{(\gamma^2+2V_d)\sigma^2/\sigma_0^2+1+2V_d}\left(\frac{(1+2V_d)(\gamma^4+2V_d)(\gamma^2+2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)^2\gamma^2(\gamma^2+2V_d)}{(\gamma^2+2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2}\right)}{(\gamma^2+2V_d)^2} \\
&= \sigma^2 \left(\frac{(\gamma^2+2V_d)^2\sigma^2/\sigma_0^2 + (1+2V_d)^2\gamma^2}{((\gamma^2+2V_d)\sigma^2/\sigma_0^2 + (1+2V_d))\left((\gamma^2+2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2\right)}\right), \qquad (27)
\end{aligned}
$$

which proves Theorem 3.1.

Since $(\gamma^2 + 2V_d)\sigma^2/\sigma_0^2 > 0$, we have

$$
s(\boldsymbol{x}', \boldsymbol{x}') < \sigma^2 \left(\frac{(\gamma^2+2V_d)^2\sigma^2/\sigma_0^2 + (1+2V_d)^2\gamma^2}{(1+2V_d)\left((\gamma^2+2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2\right)}\right) = \sigma^2,
$$

which proves Corollary 3.2. In addition, asymptotic expansion of Eq.(27) for the case when $\sigma^2/\sigma_0^2 \ll 1$ gives the following corollary.

Corollary C.1. *If* $\frac{\sigma^2}{\sigma_0^2} \ll 1$,

$$
s_{\boldsymbol{X}}(\boldsymbol{x}', \boldsymbol{x}') = \begin{cases} \sigma^2\left(1 + O(\frac{\sigma^2}{\sigma_0^2})\right) & \textit{for } \gamma^2 = \Theta(1), \\ \sigma^2\left(\frac{2V_d}{1+2V_d} + O(\frac{\sigma^2}{\sigma_0^2})\right) & \textit{for } \gamma^2 \to +0, \\ \sigma^2\left(\frac{1}{1+2V_d} + O(\frac{\sigma^2}{\sigma_0^2})\right) & \textit{for } \gamma^2 \to \infty. \end{cases}
$$

Corollary C.1 implies that, for any $\gamma^2 \in (0, \infty)$, the asymptotic uncertainty has the same form. Setting $\gamma^2 \to +0$ removes the constant term, and thus the degree of freedom is reduced by one. Similarly $\gamma^2 \to \infty$ removes the $2V_d$ terms except the constant, and thus the degree of freedom is reduced by $2V_d$.

C.1. Relation to Fourier Analysis

C.1.1. Equivalence between GP regression and Bayesian linear regression

In general, it is known that the GP regression model

$$
p(y_n|\boldsymbol{x}_n, f(\cdot)) = \mathcal{N}_D(y_n; f(\boldsymbol{x}_n), \sigma_n^2), \qquad (28)
$$

$$p(f) = \mathrm{GP}(f; 0(\cdot), k(\cdot,\cdot)), \tag{29}$$
$$\tag{30}$$

is equivalent to the Bayesian linear regression model

$$p(y_n|\boldsymbol{x}_n, \boldsymbol{b}) = \mathcal{N}_D(y_n; f(\boldsymbol{x}_n; \boldsymbol{b}), \sigma_n^2), \tag{31}$$
$$f(\boldsymbol{x}; \boldsymbol{b}) = \boldsymbol{b}^\top \boldsymbol{\phi}(\boldsymbol{x}), \tag{32}$$
$$p(\boldsymbol{b}) = \mathcal{N}_H(\boldsymbol{b}; \boldsymbol{0}, \boldsymbol{I}_H), \tag{33}$$

where $\boldsymbol{\phi}(\boldsymbol{x}) \in \mathbb{R}^H$ is the finite-dimensional input feature such that $k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\phi}(\boldsymbol{x})^\top \boldsymbol{\phi}(\boldsymbol{x}')$. This can be shown as follows. Let $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) \in \mathbb{R}^{D\times N}, \boldsymbol{y} = (y_1, \ldots, y_N)^\top \in \mathbb{R}^N$ be the training data, and $\boldsymbol{\Phi} = (\boldsymbol{\phi}(\boldsymbol{x}_1), \ldots, \boldsymbol{\phi}(\boldsymbol{x}_N)) \in \mathbb{R}^{H\times N}$ be the corresponding input features. The posterior of the Bayesian liner regression model (31)–(33) is

$$\begin{aligned}
p(\boldsymbol{b}|\boldsymbol{X}, \boldsymbol{y}) &\propto p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{b}) p(\boldsymbol{b}) \\
&\propto \exp\left(-\frac{(\boldsymbol{y} - \boldsymbol{\Phi}^\top \boldsymbol{b})^\top \boldsymbol{S}^{-1} (\boldsymbol{y} - \boldsymbol{\Phi}^\top \boldsymbol{b})}{2} - \frac{\boldsymbol{b}^\top \boldsymbol{I}_H^{-1} \boldsymbol{b}}{2}\right) \\
&\propto \exp\left(-\frac{(\boldsymbol{b} - \boldsymbol{\mu}_b)^\top \boldsymbol{\Sigma}_b^{-1} (\boldsymbol{b} - \boldsymbol{\mu}_b)}{2}\right),
\end{aligned}$$

where

$$\begin{aligned}
\boldsymbol{\mu}_b &= \boldsymbol{\Sigma}_b \boldsymbol{\Phi} \boldsymbol{S}^{-1} \boldsymbol{y}, \\
\boldsymbol{\Sigma}_b &= \left(\boldsymbol{\Phi} \boldsymbol{S}^{-1} \boldsymbol{\Phi}^\top + \boldsymbol{I}_H\right)^{-1} = \boldsymbol{I}_H - \boldsymbol{\Phi}\left(\boldsymbol{S} + \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^\top.
\end{aligned}$$

Therefore, the posterior is

$$p(\boldsymbol{b}|\boldsymbol{X}, \boldsymbol{y}) = \mathcal{N}_H(\boldsymbol{b}; \boldsymbol{\mu}_b, \boldsymbol{\Sigma}_b).$$

The noiseless predictions $\boldsymbol{f}' = (f(\boldsymbol{x}'_1), \ldots, f(\boldsymbol{x}'_M))^\top \in \mathbb{R}^M$ at the test points $\boldsymbol{X}' = (\boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_M) \in \mathbb{R}^{D\times M}$ with the corresponding features $\boldsymbol{\Phi}' = (\boldsymbol{\phi}(\boldsymbol{x}'_1), \ldots, \boldsymbol{\phi}(\boldsymbol{x}'_M)) \in \mathbb{R}^{H\times M}$ follow

$$\begin{aligned}
p(\boldsymbol{f}'|\boldsymbol{X}', \boldsymbol{X}, \boldsymbol{y}) &= \mathcal{N}_M(\boldsymbol{y}'; \boldsymbol{\mu}_{f'}, \boldsymbol{\Sigma}_{f'}), \\
\boldsymbol{\mu}_{f'} &= \boldsymbol{\Phi}'^\top \boldsymbol{\mu}_b \\
&= \boldsymbol{\Phi}'^\top \boldsymbol{\Sigma}_b \boldsymbol{\Phi} \boldsymbol{S}^{-1} \boldsymbol{y}, \\
&= \boldsymbol{\Phi}'^\top \left(\boldsymbol{I}_H - \boldsymbol{\Phi}\left(\boldsymbol{S} + \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^\top\right) \boldsymbol{\Phi} \boldsymbol{S}^{-1} \boldsymbol{y} \\
&= \left(\boldsymbol{\Phi}'^\top \boldsymbol{\Phi} - \boldsymbol{\Phi}'^\top \boldsymbol{\Phi}\left(\boldsymbol{S} + \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\right) \boldsymbol{S}^{-1} \boldsymbol{y} \\
&= \boldsymbol{\Phi}'^\top \boldsymbol{\Phi}\left(\boldsymbol{I}_N - \left(\boldsymbol{S} + \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\right) \boldsymbol{S}^{-1} \boldsymbol{y} \\
&= \boldsymbol{K}'^\top \left(\boldsymbol{I}_N - (\boldsymbol{S} + \boldsymbol{K})^{-1} \boldsymbol{K}\right) \boldsymbol{S}^{-1} \boldsymbol{y} \\
&= \boldsymbol{K}'^\top \left(\boldsymbol{S}^{-1} - (\boldsymbol{S} + \boldsymbol{K})^{-1} \boldsymbol{K} \boldsymbol{S}^{-1}\right) \boldsymbol{y} \\
&= \boldsymbol{K}'^\top (\boldsymbol{S} + \boldsymbol{K})^{-1} \boldsymbol{y} \qquad (34) \\
\boldsymbol{\Sigma}_{f'} &= \boldsymbol{\Phi}'^\top \boldsymbol{\Sigma}_b \boldsymbol{\Phi}' \\
&= \boldsymbol{\Phi}'^\top \left(\boldsymbol{I}_H - \boldsymbol{\Phi}\left(\boldsymbol{S} + \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^\top\right) \boldsymbol{\Phi}' \\
&= \boldsymbol{\Phi}'^\top \boldsymbol{\Phi}' - \boldsymbol{\Phi}'^\top \boldsymbol{\Phi}\left(\boldsymbol{S} + \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^\top \boldsymbol{\Phi}'
\end{aligned}$$

$$= \boldsymbol{K}'' - \boldsymbol{K}'^{\top} \left(\boldsymbol{S} + \boldsymbol{K}\right)^{-1} \boldsymbol{K}', \tag{35}$$

which match the posterior (3)–(5) of the GP regression model (28)–(29). Here, we obtained Eq. (34) by using the matrix inversion lemma

$$(\boldsymbol{A} + \boldsymbol{U}\boldsymbol{C}\boldsymbol{V})^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{U}(\boldsymbol{C}^{-1} + \boldsymbol{V}\boldsymbol{A}^{-1}\boldsymbol{U})^{-1}\boldsymbol{V}\boldsymbol{A}^{-1}$$

with $\boldsymbol{A} = \boldsymbol{S}, \boldsymbol{U} = \boldsymbol{S}, \boldsymbol{C} = \boldsymbol{S}^{-1}, \boldsymbol{V} = \boldsymbol{K}$, namely,

$$\begin{aligned}(\boldsymbol{S} + \boldsymbol{S}\boldsymbol{S}^{-1}\boldsymbol{K})^{-1} &= \boldsymbol{S}^{-1} - \boldsymbol{S}^{-1}\boldsymbol{S}((\boldsymbol{S}^{-1})^{-1} + \boldsymbol{K}\boldsymbol{S}^{-1}\boldsymbol{S})^{-1}\boldsymbol{K}\boldsymbol{S}^{-1} \\ &= \boldsymbol{S}^{-1} - (\boldsymbol{S} + \boldsymbol{K})^{-1}\boldsymbol{K}\boldsymbol{S}^{-1}.\end{aligned}$$

C.1.2. Relation to Fourier Transform

Similarly to the analysis in Appendix C, we compute the posterior mean of the GP regression at a test point $\boldsymbol{x}' = \widehat{\boldsymbol{x}} + \alpha' \boldsymbol{e}_d$,

$$\begin{aligned}
\mu(\boldsymbol{x}') &= \boldsymbol{k}'^{\top} \left(\boldsymbol{K} + \sigma^2 \boldsymbol{I}_{1+2V_d}\right)^{-1} \boldsymbol{y}, \\
&= \boldsymbol{k}'^{\top} \frac{1}{\sigma_0^2} a \left(\boldsymbol{I}_{1+2V_d} + b \boldsymbol{1}_{1+2V_d} \boldsymbol{1}_{1+2V_d}^{\top}\right) \boldsymbol{y} \\
&= \frac{1}{\sigma_0^2} a \left(\boldsymbol{k}'^{\top} \boldsymbol{y} + b \boldsymbol{k}'^{\top} \boldsymbol{1}_{1+2V_d} \boldsymbol{1}_{1+2V_d}^{\top} \boldsymbol{y}\right) \\
&= \frac{1}{\sigma_0^2} a \left(\sigma_0^2 \left(\sum_{w=0}^{2V_d} y_w \frac{\gamma^2 + 2\sum_{v=1}^{V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right)}{\gamma^2 + 2V_d}\right) + b\sigma_0^2 \frac{(1+2V_d)\gamma^2}{\gamma^2 + 2V_d} \sum_{w=0}^{2V_d} y_w\right) \\
&= \frac{a}{\gamma^2 + 2V_d} \sum_{w=0}^{2V_d} y_w \left(\gamma^2 + 2\sum_{v=1}^{V_d} \cos\left(v\left(\frac{2w\pi}{1+2V_d} - \alpha'\right)\right) + b(1+2V_d)\gamma^2\right) \\
&= \frac{a}{\gamma^2 + 2V_d} \sum_{w=0}^{2V_d} y_w \left(\gamma^2 + 2\sum_{v=1}^{V_d} \left\{\cos v\left(\frac{2w\pi}{1+2V_d}\right) \cos v\alpha' + \sin v\left(\frac{2w\pi}{1+2V_d}\right) \sin v\alpha'\right\}\right. \\
&\qquad \left. - \frac{(\gamma^2 - 1)(1+2V_d)\gamma^2}{(\gamma^2 + 2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2}\right) \\
&= \frac{a}{\gamma^2 + 2V_d} \sum_{w=0}^{2V_d} y_w \left(2\sum_{v=1}^{V_d} \left\{\cos v\left(\frac{2w\pi}{1+2V_d}\right) \cos v\alpha' + \sin v\left(\frac{2w\pi}{1+2V_d}\right) \sin v\alpha'\right\}\right. \\
&\qquad \left. + \frac{\gamma^2\left((\gamma^2 + 2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2\right) - (\gamma^2 - 1)(1+2V_d)\gamma^2}{(\gamma^2 + 2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2}\right) \\
&= \frac{1}{(\gamma^2 + 2V_d)\sigma^2/\sigma_0^2 + 1 + 2V_d} \sum_{w=0}^{2V_d} y_w \left(2\sum_{v=1}^{V_d} \left\{\cos v\left(\frac{2w\pi}{1+2V_d}\right) \cos v\alpha' + \sin v\left(\frac{2w\pi}{1+2V_d}\right) \sin v\alpha'\right\}\right. \\
&\qquad \left. + \frac{\gamma^2(\gamma^2 + 2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2}{(\gamma^2 + 2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2}\right) \\
&= \eta_0 + \sum_{v=1}^{V_d} \left(\eta_{\cos,v}\sqrt{2}\cos v\alpha' + \eta_{\sin,v}\sqrt{2}\sin v\alpha'\right),
\end{aligned}$$

where

$$\begin{aligned}
\eta_0 &= \frac{\gamma^2}{(\gamma^2 + 2V_d)\sigma^2/\sigma_0^2 + (1+2V_d)\gamma^2} \sum_{w=0}^{2V_d} y_w, \\
\eta_{\cos,v} &= \frac{1}{(\gamma^2 + 2V_d)\sigma^2/\sigma_0^2 + 1 + 2V_d} \sum_{w=0}^{2V_d} y_w \sqrt{2}\cos v\left(\frac{2w\pi}{1+2V_d}\right),
\end{aligned}$$

Table 1. Algorithm-specific choice of parameters for EMICoRe and SubsCoRe for all experiments (unless specified otherwise).

	Algorithm-specific parameters	
`--acq-params`	**EMICoRe params**	as in Nicoli et al. (2023a)
`func`	`func=ei`	Base acq. func. type
`optim`	`optim=emicore`	Optimizer type
`pairsize` (J_{SG})	`20`	# of candidate points
`gridsize` (J_{OG})	`100`	# of evaluation points
`corethresh` (κ)	`1.0`	CoRe threshold κ
`corethresh_width` (T_{Ave})	`10`	# averaging steps to update κ
`coremin_scale` (C_0)	`0.0`	Coefficient C_0 for updating κ
`corethresh_scale` (C_1)	`1.0`	Coefficient C_1 for updating κ
`samplesize` (N_{MC})	`100`	# of MC samples
`smo-steps` (T_{NFT})	`0`	# of initial NFT steps
`smo-axis`	`True`	Sequential direction choice
`--acq-params`	**SubsCoRe params**	this paper[‖]
`optim`	`readout`	Optimizer type
`readout-strategy`	`center/core`	Alg type SubsCoRe/SubsCoRe-Bound
`corethresh-strategy`	`linreg`	Linear regression for κ
`corethresh` (κ)	`512`	Initial N_{shots} for CoRe
`corethresh_width` (T_{Ave})	`40`	# averaging steps to update κ
`coremin_scale` (C_0)	`1024`	Coefficient C_0 for updating κ
`corethresh_scale` (C_1)	`1.0`	Coefficient C_1 for updating κ
`coremetric`	`readout`	Metric to set CoRe

$$\eta_{\sin,v} = \frac{1}{(\gamma^2 + 2V_d)\sigma^2/\sigma_0^2 + 1 + 2V_d} \sum_{w=0}^{2V_d} y_w \sqrt{2} \sin v \left(\frac{2w\pi}{1+2V_d} \right).$$

When $\gamma = 1$,

$$\begin{aligned}
\eta_0 &= \frac{1}{(1+2V_d)\left(\sigma^2/\sigma_0^2 + 1\right)} \sum_{w=0}^{2V_d} y_w, \\
\eta_{\cos,v} &= \frac{1}{(1+2V_d)\left(\sigma^2/\sigma_0^2 + 1\right)} \sum_{w=0}^{2V_d} y_w \sqrt{2} \cos v \left(\frac{2w\pi}{1+2V_d} \right), \\
\eta_{\sin,v} &= \frac{1}{(1+2V_d)\left(\sigma^2/\sigma_0^2 + 1\right)} \sum_{w=0}^{2V_d} y_w \sqrt{2} \sin v \left(\frac{2w\pi}{1+2V_d} \right),
\end{aligned}$$

which is the regularized discrete Fourier transform, i.e., converges to the standard Fourier transform as $\sigma^2/\sigma_0^2 \to 0$.

D. Algorithm Details

D.1. Parameter Setting

Every algorithm evaluated in our benchmarking analysis has several hyperparameters to be set. For transparency and to allow the reproduction of our experiments, we detail the choice of parameters for EMICoRe and SubsCoRe in Table 1. The SGLBO results were obtained using the original code from Tamiya & Yamasaki (2022) and we used the default setting from the original paper. For the NFT run, we used the default parameters specified in Table 2. For (classical) computational

[‖]All hyperparameters not specified in the table are set to the default values in Nicoli et al. (2023a).

efficiency, all EMICoRe and SubsCoRe runs used the inducer option retaining only the last < 100 measured points once more than 120 points were stored in the GP. Specifically, after determining the points to be discarded, we train a temporary GP on those points and replace them with the pivot point right after the latest discarded points, along with its prediction by the temporary GP. This allows the GP to be of constant size throughout the optimization.

D.2. Pseudocode

In this section we provide a detailed description of our proposed algorithm. Specifically, Algorithm 1 describes the full SubsCoRe algorithm while Algorithm 2 focuses on the sub-routine used therein.

Algorithm 1 (SubsCoRe Algorithm) Optimization scheme using the SubsCoRe subroutine, as described in Algorithm 2, for finding the optimal parameters $\hat{\mathbf{x}}$ for the quantum circuit. The optimization stops when the total number of measurement shots reaches the maximum number of observation shots allowed, i.e., $N_{\text{tot}-\text{shots}}$. To avoid cluttering notation, the algorithm is restricted to the case where $V_d = 1$. Generalization to an arbitrary V_d is straightforward.

Input :
- $\hat{\boldsymbol{x}}^0$: initial starting point (best point)
- $\bar{N}_0$: initial number of shots for starting point
- κ_0 : Initial CoRe threshold at step $t = 0$.

Parameters :
- $V_d = 1$
- D : number of parameters to optimize, i.e., $\hat{\boldsymbol{x}} \in \mathbb{R}^D$.
- $\hat{\alpha}$: shift from best point at the previous step (default to $\hat{\alpha} = \frac{2\pi}{3}$)
- $N_{\text{tot-shots}}$: Total # of shots, i.e., maximum allowed quantum computing budget.

Output :
- $\hat{\boldsymbol{x}}_T$: optimal choice of parameters for the quantum circuit.

$n \leftarrow 0$ /* initialize consumed shot budget */
$t \leftarrow 0$ /* initialize optimization step */
$d \leftarrow 0$ /* initialize optimization subspace/ direction */
$y_0 \leftarrow$ quantum_circuit(parameters=$\hat{\boldsymbol{x}}^0$, shots=$\bar{N}_0$) /* measure initial best point */
$\boldsymbol{X}^0, \boldsymbol{y}^0, \boldsymbol{\sigma}^0 \leftarrow (\hat{\boldsymbol{x}}_0), (y_0), (\frac{\eta^2}{\bar{N}_0})$ /* initialize gaussian process */
while $n < N_{\textit{tot-shots}}$ **do**
  $\breve{\mathbf{X}} \leftarrow \left(\hat{\boldsymbol{x}}^t - \hat{\alpha} \cdot \boldsymbol{e}_d, \hat{\boldsymbol{x}}^t, \hat{\boldsymbol{x}}^t + \hat{\alpha} \cdot \boldsymbol{e}_d\right)$ /* choose points to measure along d */
  $\bar{N}^{t+1} \leftarrow$ choose_shots($\boldsymbol{X}^t, \boldsymbol{y}^t, \boldsymbol{\sigma}^t, \breve{\mathbf{X}}, \kappa_t, d$) /* choose number of shots (Alg. 2) */
  for $i \in \{1, ..., |\breve{\mathbf{X}}|\}$ **do**
    $\breve{y}_i \leftarrow$ quantum_circuit(parameters=$\breve{\mathbf{X}}_i$, shots=$\bar{N}_0$) /* measure chosen points */
    $\breve{\sigma}_i \leftarrow \frac{\eta^2}{\bar{N}_i^{t+1}}$
  end
  $\breve{\boldsymbol{y}}, \breve{\boldsymbol{\sigma}} \leftarrow (\breve{y}_1, ..., \breve{y}_{|\breve{\mathbf{X}}|}), (\breve{\sigma}_1, ..., \breve{\sigma}_{|\breve{\mathbf{X}}|})$
  $\boldsymbol{X}^{t+1}, \boldsymbol{y}^{t+1}, \boldsymbol{\sigma}^{t+1} \leftarrow (\boldsymbol{X}^t, \breve{\boldsymbol{X}}), (\boldsymbol{y}^t, \breve{\boldsymbol{y}}), (\boldsymbol{\sigma}^t, \breve{\boldsymbol{\sigma}})$ /* add new points to gaussian process */
  $\widehat{\boldsymbol{x}}^{t+1} \leftarrow \text{argmin}_{\boldsymbol{x}'} \mu_{|\boldsymbol{X}^{t+1}, \boldsymbol{\sigma}^{t+1}, \boldsymbol{y}^{t+1}}(\boldsymbol{x}')$ /* find point with minimum GP mean on d */
  $\hat{y}^{t+1} \leftarrow \mu_{|\boldsymbol{X}^{t+1}, \boldsymbol{\sigma}^{t+1}, \boldsymbol{y}^{t+1}}(\widehat{\boldsymbol{x}}^{t+1})$ /* remember current minimum GP mean as best value */
  if $t > T_{\text{Ave}}$ **then**
    $\kappa^{t+1} \leftarrow \max(C_0, -C_1 \cdot \text{Slope}(\{\widehat{y}^{t'}\}_{t'=(t+1)-T_{\text{Ave}}}^{t+1}))$. /* update the CoRe threshold */
    /* N.B. Slope($\cdot$) estimates the progress of optimization by linear regression to the T_{Ave} recent best values. */
  end
  $t \leftarrow t + 1$ /* update the step */
  $n \leftarrow n + \sum_i \bar{N}_i^{t+1}$ /* update the consumed shot budget */
  $d \leftarrow (d + 1) \mod D$ /* select the new subspace/direction to optimize */
end
return $\hat{\boldsymbol{x}}_T$

Algorithm 2 (SubsCoRe Subroutine to identify the number of shots) Find the number of shots for the points chosen to be measured in SubsCoRe. Can be used for either SubsCoRe-Bound, or SubsCoRe-Center.

Input :
- $\boldsymbol{X}^t, \boldsymbol{y}^t, \boldsymbol{\sigma}^t$: Gaussian Process at step t
- $\check{\boldsymbol{X}}$: Points for which to choose the number of shots (in order: left shift, center, right shift; see line 7 of Alg. 1)
- κ : CoRe threshold at step t.
- d : direction along which $\check{\boldsymbol{X}}$ are distributed

Parameters :
- $V_d = 1$
- η^2 : measurement variance using a single shot.

Output :
- $\bar{N}_{t+1}$: shots for performing new measurements at step $t+1$.

begin
- $\mathcal{S}_d(\check{\boldsymbol{X}}_1) \leftarrow \{\check{\boldsymbol{X}}_1 + \alpha \cdot \boldsymbol{e}_d\,;\ \alpha \in [0, 2\pi)\}$; /* define points along d */
- $\check{\sigma}_\pm \leftarrow \kappa$ /* initialize shifted point variance to worst case */
- **for** $\tilde{\sigma} \in [\eta, \kappa]$ **do**
  - /* create temporary GP copies, add chosen points with $\tilde{\sigma}$ observation noise */
  - $\boldsymbol{X}^{t'}, \boldsymbol{y}^{t'}, \boldsymbol{\sigma}^{t'} \leftarrow (\boldsymbol{X}^t, \check{\boldsymbol{X}}), (\boldsymbol{y}^t, 0, 0, 0), (\boldsymbol{\sigma}^t, \tilde{\sigma}, \tilde{\sigma}, \tilde{\sigma})$
  - /* get smallest observation noise for which temporary GP's prediction of all points along d are within the core */
  - **if** $(\forall \boldsymbol{x} \in \mathcal{S}_d(\check{\boldsymbol{X}}_1))\ :\ s_{|\boldsymbol{X}^{t'}, \boldsymbol{\sigma}^{t'}}(\boldsymbol{x}, \boldsymbol{x}) \leq \kappa^2) \wedge (\check{\sigma}_\pm > \tilde{\sigma})$ **then**
    - $\check{\sigma}_\pm \leftarrow \tilde{\sigma}$
  - **end**
- **end**
- $\bar{N}_\pm \leftarrow \frac{\eta^2}{\check{\sigma}_\pm}$ /* compute shots from variance through single shot variance η^2 */
- **if** *using SubsCoRe-Bound* **then**
  - $\bar{N}_{t+1} \leftarrow (\bar{N}_\pm, \bar{N}_\pm, \bar{N}_\pm)$ /* choose the same number of shots for each point */
  - **return** $\bar{N}_{t+1}$
- **end**
- /* otherwise, we use SubsCoRe-Center, where we reevaluate the center */
- $\check{\sigma}_0 \leftarrow \kappa$ /* initialize center point variance to worst case */
- **for** $\tilde{\sigma} \in [\eta, \kappa]$ **do**
  - /* create temporary GP copies, add shifted points with $\check{\sigma}_\pm$ and center with $\tilde{\sigma}$ observation noise */
  - $\boldsymbol{X}^{t'}, \boldsymbol{y}^{t'}, \boldsymbol{\sigma}^{t'} \leftarrow (\boldsymbol{X}^t, \check{\boldsymbol{X}}), (\boldsymbol{y}^t, 0, 0, 0), (\boldsymbol{\sigma}^t, \check{\sigma}_\pm, \tilde{\sigma}, \check{\sigma}_\pm)$
  - /* get smallest observation noise for which temporary GP's prediction of all points along d are within the core */
  - **if** $(\forall \boldsymbol{x} \in \mathcal{S}_d(\check{\boldsymbol{X}}_1))\ :\ s_{|\boldsymbol{X}^{t'}, \boldsymbol{\sigma}^{t'}}(\boldsymbol{x}, \boldsymbol{x}) \leq \kappa^2) \wedge (\check{\sigma}_0 > \tilde{\sigma})$ **then**
    - $\check{\sigma}_0 \leftarrow \tilde{\sigma}$
  - **end**
- **end**
- $\bar{N}_0 \leftarrow \frac{\eta^2}{\check{\sigma}_0}$ /* compute shots from variance through single shot variance η^2 */
- $\bar{N}_{t+1} \leftarrow (\bar{N}_\pm, \bar{N}_0, \bar{N}_\pm)$ /* choose shots for center different from shifted points */
- **return** $\bar{N}_{t+1}$

end

E. Experimental Details

E.1. Experimental Setting

As mentioned in the main text, we follow the experimental setup in Nicoli et al. (2023a). Specifically, starting from the quantum Heisenberg Hamiltonian, we focus on the special case of the Ising Hamiltonian at the critical point by choosing the

Table 2. Default choice of circuit parameters and hyperparameter optimization (for EMICoRe and SubsCoRe) in all experiments (unless specified otherwise).

	Deafult params	
`--n-qbits`	`5`	# of qubits
`--n-layers`	`3`	# of circuit layers
`--circuit`	`esu2`	Circuit name
`--pbc`	`False`	Open Boundary Conditions
`--n-iter`	`3*10**6`	# max number of readouts
`--iter-mode`	`readout`	# max number of readouts
`--kernel`	`vqe`	Name of the kernel
`--hyperopt`	**Hyperparams optim**	
`optim`	`grid`	Grid-search optimization of γ
`max_gamma`	`20`	Max value for γ
`interval`	`100*1+20*9+10*100`	Scheduling for grid-search
`steps`	`90`	# steps in grid
`loss`	`loo`	Loss type

suitable couplings, namely

$$\text{Ising Hamiltonian at criticality: } J = (-1.0,\ 0.0,\ 0.0);\ h = (0.0,\ 0.0,\ -1.0).$$

It is important to note that for the system at hand, this choice of parameters already represents a challenging setup for a fixed number of qubits, as discussed in Sec. I.2 in Nicoli et al. (2023a).

For all methods, we stopped the optimization when a maximum number of cumulative shots (total measurement budget on the quantum computer) is reached; unless specified otherwise we set this cutoff to $N_{\text{shots}}^{\text{max}} = 2.5 \cdot 10^6$. We based our implementation on the EMICoRe code available on GitHub (Nicoli et al., 2023b) at *https://github.com/emicore/emicore*. Since EMICoRe and SubsCoRe use a GP combined with the VQE kernel (Nicoli et al., 2023a), we need to set the kernel parameters γ and σ_0. Similarly to Nicoli et al. (2023a), we set σ_0 based on a rough approximation of the ground-state energy, and optimize γ by the leave-one-out cross validation with grid-search (90 points) in the range $[\gamma_{\text{min}}, \gamma_{\text{max}}] = [\sqrt{2}, 20]$. γ_{max} and the frequency of the γ update can be specified by `max_gamma` and `interval`, respectively, in our code (see Table 2). For NFT, we used the fixed equidistant shifts $\alpha = \pm\frac{2\pi}{3}$.

Each experiment shown in the paper has been repeated 100 times (independent seeded trials) with different starting points determined by the seeds. We aggregated the statistics from these independent seeded trials and presented them in our plots. For a fair comparison, we used the same starting point in each trial (i.e., for each corresponding seed) for all algorithms. All experiments were conducted on Intel Xeon Silver 4316 @ 2.30GHz CPUs.

E.2. Number of Shots per SMO Step

The adaptive number of total shots (for all measurements combined) per SMO step for SubsCoRe-Center (red) is shown in Figure 6.

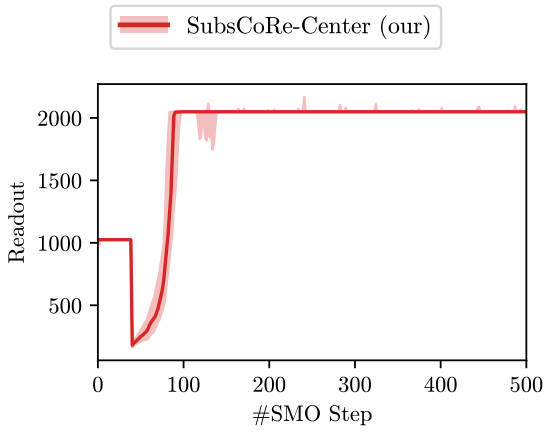

Figure 6. Number of measurement shots (Readout) taken by SubsCoRe-Center for each SMO step. The x-axis indicates the SMO step while the y-axis represents the total number of measurement shots taken at every step (i.e., distributed over the three observed points per step).