

Markov Chain Monte Carlo with Imaginary Time Evolution for Portfolio Optimization

Shruti Jain Tobias Hartung Karl Jansen Hernan Leövey

Motivation Portfolio optimization (PO) seeks asset weights that best balance risk and return. Classical methods — mean-variance optimization, Monte Carlo simulation, and linear/quadratic programming — work well at small scale, but the number of candidate portfolios grows exponentially with asset count, making exhaustive search quickly intractable. This combinatorial blow-up motivates quantum approaches, where superposition and entanglement let many portfolio configurations be represented and explored at once.

However, variational quantum algorithms (VQAs) like VQE and QAOA suffer from known trainability problems: their optimization landscapes contain local minima and barren plateaus that worsen with circuit depth and qubit count — a serious issue at realistic portfolio sizes [1–12]. Separately, MCMC-based optimizers have been shown to give VQAs provable convergence to the global minimum on toy combinatorial problems [13]. Building on this, we use a VQA whose parameterized quantum circuit (PQC) is derived from Imaginary Time Evolution (ITE) and trained with a gradient-guided MCMC optimizer, which steers parameter updates along the descent direction while preserving MCMC’s global-convergence guarantees. Concretely, we solve the following problem:

Problem Description We consider a real-world PO problem with N strategies (investment rules for constructing a portfolio), where strategy i is allocated weight γ_i . The goal is to determine the optimal set of weighted strategies subject to practical constraints. The PO problem is thus formulated as maximizing the cost function, $C(\gamma_i) = \text{Trend}(\gamma_i) - \text{Deviation}(\gamma_i) - \text{Instability}(\gamma_i)$, with respect to γ_i , where the trend, deviation, and instability terms depend on γ_i linearly, via an absolute value, and quadratically, respectively. Since this mixed dependence prevents $C(\gamma_i)$ from being encoded directly into a Hamiltonian, we instead adopt a random bit-string generation approach: an N -bit bit-string — one bit per strategy, encoded as a qubit — is mapped onto the N weights γ_i . In this work, we restrict to a reduced dataset of 8 strategies with binary weights $\gamma_i \in \{0, 1\}^N$, indicating whether strategy i is activated or not.

Generating these bit-strings requires a PQC capable of exploring the solution space efficiently while remaining trainable at this problem size; we achieve this by approximating ITE using Pauli rotation gates.

Approximating ITE using Pauli rotation gates ITE evolves an arbitrary initial state toward the ground state [14, 15] of a Hamiltonian H by decaying the excited-energy states relative to the ground-state component. The energy gradient along the quantum imaginary-time evolution path is known to not vanish, with the energy decreasing monotonically [16]. Given a sufficiently expressive circuit [17], this ensures that the global solution is attainable [16]. However, since the ITE operator $e^{-H\tau}$ is non-unitary in nature, it cannot be realized directly as a quantum gate and must instead be approximated through variational or measurement-assisted schemes [18].

Writing the Hamiltonian H as a sum of Pauli strings (S_i), $H = \sum_i a_i S_i$ with a_i real coefficients, the first-order Suzuki-Trotter decomposition [19] of the ITE operator is

$$e^{-\Delta t H} = (e^{-\delta H})^{\Delta t/\delta} \approx \prod_{\tau=1}^{\Delta t/\delta} \prod_i e^{-\delta a_i S_i}, \quad (1)$$

where δ is the small imaginary-time step. Applying a Taylor expansion to the single-term factor reduces this to

$$e^{-\delta a_i S_i} = \cosh(\delta a_i) \mathbb{I} - \sinh(\delta a_i) S_i. \quad (2)$$

Renormalizing equation 2 amounts to a rotation within the plane spanned by $|\psi\rangle$ and $S_i |\psi\rangle$ —exactly the action generated by a unitary Pauli rotation gate. We therefore replace each single-term factor in equation 1 with a parameterized unitary rotation gate [16, 17, 20, 21]. In this work, we extend our approach to a second-order Trotter decomposition for approximating the ITE operator. This yields three PQC configurations: (i) a first-order approximation consisting of rotation Y-gates applied to each qubit, (ii) a second-order approximation consisting of rotation Y-gates together with two-qubit rotation gates in a nearest-neighbor Hamiltonian approximation, and (iii) a two-layer second-order approximation PQC.

While this construction yields a trainable, expressive PQC, realizing the full benefit of the ITE path requires a classical optimizer capable of faithfully tracking its gradient-descent-like trajectory; we address this next with a gradient-guided MCMC optimizer.

Gradient guided Markov Chain Monte Carlo (GradMCMC) Optimizer Since the real-world optimization problems encountered in VQAs are often characterized by high-dimensional search spaces with highly non-trivial cost function landscapes, we consider an MCMC-based optimizer, as such optimizers have shown remarkable success in high-energy physics applications with similar characteristics [22]. The MCMC optimizer is built on the Metropolis-Hastings algorithm [23, 24], with probability density proportional to $\exp(-\beta \Delta C(\gamma_i))$, where β is the inverse-temperature parameter.

We propose a gradient-guided MCMC optimizer (GradMCMC) that incorporates gradient descent information into the parameter updates: the proposed parameter component is a normally distributed perturbation with mean given by a gradient descent step and standard deviation σ_0 . Biasing proposals along the energy gradient direction steers them toward the variational quantum imaginary-time evolution (VarQITE) solution path, with the resulting proposal then corrected via a Metropolis-Hastings accept-reject step to optimize the target cost function.

The GradMCMC optimizer can be understood as an interlaced, component-wise Robbins-Monro algorithm [25] with random step variables $a_k \sim \mathcal{N}\left(-\frac{\nabla(C(\gamma))}{k}, \left(\frac{\sigma_0}{k}\right)^2\right)$. We therefore expect convergence to the optimum as the number of iterations diverges, provided $\sum_{k=0}^{\infty} a_k = \infty$ and $\sum_{k=0}^{\infty} a_k^2 < \infty$: the harmonic scaling ensures global reach, as the series of steps diverges, while convergence of the squares controls the total variance.

Results and Discussion *Error propagation* — Assuming convergence after I iterations, the approximate parameter θ returned by the GradMCMC optimizer has a per-component error of order σ_0/I . To choose σ_0 and I such that the cost function attains a desired accuracy, we propagate the error in θ through to the error in the cost function value, in three steps. Let $E_{1,p} := \|\theta - \Theta\|_p$ denote the error in the GradMCMC parameters in ℓ_p -norm. This yields an error $E_2 := \|U(\theta) - U(\Theta)\|$ in the quantum circuit, which in turn leads to an error $E_3 := \|U(\theta)|\psi\rangle - U(\Theta)|\psi\rangle\|$ in the prepared final state, where $|\psi\rangle$ is the initial state of the VQA. Finally, the state error E_3 leads to an observable error $E_4 = |\langle U(\theta)^* \Omega U(\theta) \rangle - \langle U(\Theta)^* \Omega U(\Theta) \rangle|$.

Bounding the distance between the correct and approximate output states gives $E_3 \leq E_2$. Writing the quantum circuit as $U(\Theta) = \prod_{j=1}^N e^{-i\Theta_j P_j/2}$ and applying the triangle inequality for the operator norm, $\|UV - uv\| \leq \|U - u\| + \|V - v\|$, gives $E_2 \leq c \cdot E_1$, with c a constant. The resulting difference between the correct and approximate measurements gives an error in measuring an observable Ω of

$E_4 \leq E_3 \|\Omega\| + E_3^2 \|\Omega\|$. Thus, given the parameter error, we can bound the errors associated with the quantum circuit, the resulting output states, and the final measurement.

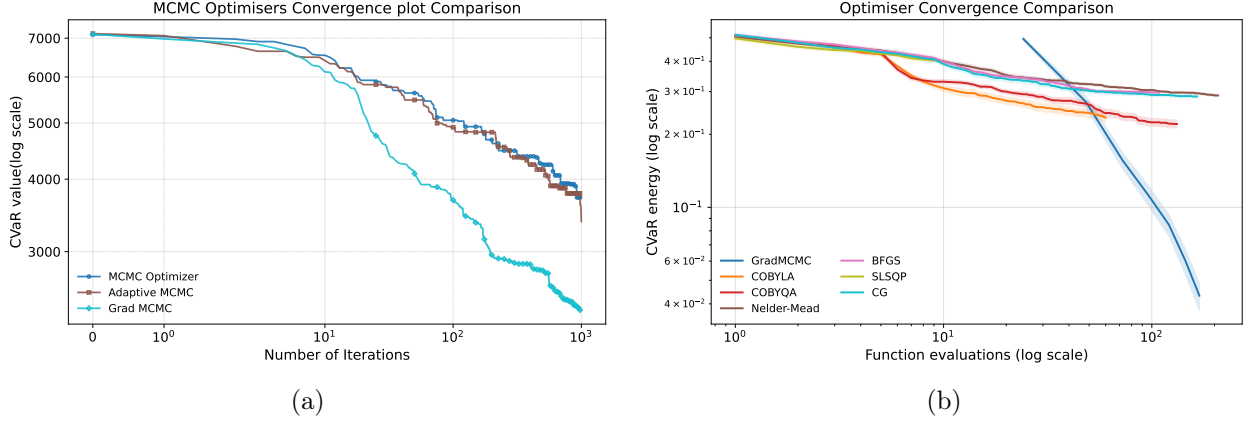


Figure 1: Comparing convergence of the GradMCMC for second-order approximate-ITE PQC against (a) MCMC optimizers and (b) Standard optimizers with a random Hamiltonian instance at each trial.

Comparison of convergence plots— (A) We compare convergence for standard MCMC, adaptive MCMC (where the standard deviation for parameter updates as linear decay schedule per step), and GradMCMC in Fig. 1a for three PQC configurations. GradMCMC consistently achieves the fastest convergence across all three PQCs, indicating effective exploitation of gradient information: incorporating this information accelerates convergence, as the optimizer searches in the direction of steepest descent.

(B) We compare the convergence behavior of optimizers like COBYLA, COBYQA, Nelder-Mead, BFGS, SLSQP, Conjugate Gradient (CG) against GradMCMC Fig. 1b under two conditions: (i) each trial uses a different, randomly generated instance, and (ii) all trials solve the same fixed random instance. In both settings, GradMCMC (blue) converges steadily throughout the budget of function evaluations.

Hyperparameter scan— We study the significance of the hyperparameters σ and β in the GradMCMC optimizer by plotting a heatmap of log-probability values—the average final energy for a given initial σ and β —for (a) the β term and (b) the $\beta(j+1)$ term in the Metropolis-Hastings accept-reject step. The best-performing region (minimum average final energy for given initial σ and β) occurs at larger β and σ values, while extreme combinations, particularly very low σ with any β , yield the worst log-probabilities. We perform this scan for both β conditions across all three PQC configurations.

Outlook The results suggest that our gradient-guided MCMC optimizer offers a promising, hardware-efficient direction for tackling the rugged optimization cost landscapes typical of near-term VQAs, a direction of interest to the NISQ and hybrid-algorithm design communities, and future work will extend this evaluation to larger financial datasets, relevant to combinatorial optimization and finance communities, and implementation on actual quantum hardware. While the results are also based on random Hamiltonian instances, they point towards the potential of GradMCMC for a broader class of high-dimensional VQA cost functions, which we plan to validate on larger-scale and hardware-realistic settings. Our findings indicate that GradMCMC can serve as a general-purpose optimizer for VQAs beyond PO, motivating further study on other combinatorial and energy-inspired cost landscapes.

References

- [1] Jarrod R. McClean et al. “Barren plateaus in quantum neural network training landscapes”. In: *Nature Communications* 9 (2018), p. 4812. DOI: 10.1038/s41467-018-07090-4.
- [2] Lennart Bittel and Martin Kliesch. “Training Variational Quantum Algorithms Is NP-Hard”. In: *Physical Review Letters* 127.12 (2021), p. 120502. DOI: 10.1103/PhysRevLett.127.120502.
- [3] Xuchen You and Xiaodi Wu. “Exponentially Many Local Minima in Quantum Neural Networks”. In: *Proceedings of the 38th International Conference on Machine Learning*. Vol. 139. Proceedings of Machine Learning Research. 2021, pp. 12144–12155. arXiv: 2110.02479.
- [4] Eric R. Anschuetz and Bobak T. Kiani. “Quantum variational algorithms are swamped with traps”. In: *Nature Communications* 13 (2022), p. 7760. DOI: 10.1038/s41467-022-35364-5. arXiv: 2205.05786.
- [5] M. Cerezo et al. “Cost function dependent barren plateaus in shallow parametrized quantum circuits”. In: *Nature Communications* 12 (2021), p. 1791. DOI: 10.1038/s41467-021-21728-w.
- [6] Samson Wang et al. “Noise-induced barren plateaus in variational quantum algorithms”. In: *Nature Communications* 12 (2021), p. 6961. DOI: 10.1038/s41467-021-27045-6.
- [7] Carlos Ortiz Marrero, Mária Kieferová, and Nathan Wiebe. “Entanglement-Induced Barren Plateaus”. In: *PRX Quantum* 2.4 (2021), p. 040316. DOI: 10.1103/PRXQuantum.2.040316.
- [8] Zoë Holmes et al. “Connecting Ansatz Expressibility to Gradient Magnitudes and Barren Plateaus”. In: *PRX Quantum* 3.1 (2022), p. 010313. DOI: 10.1103/PRXQuantum.3.010313.
- [9] Martin Larocca et al. “Barren plateaus in variational quantum computing”. In: *Nature Reviews Physics* (2025). DOI: 10.1038/s42254-025-00813-9.
- [10] “Best practices for portfolio optimization by quantum computing, experimented on real quantum devices”. In: *Scientific Reports* 13 (2023), p. 19017. DOI: 10.1038/s41598-023-45392-w.
- [11] “Scaling the Variational Quantum Eigensolver for Dynamic Portfolio Optimization”. In: *arXiv preprint* (2024). arXiv: 2412.19150.
- [12] “VQA for Dynamic Portfolio Optimization: Sampling Strategies, Optimizer Scheduling, and Hardware-Aware Ansatz Design”. In: *arXiv preprint* (2026). arXiv: 2606.10098.
- [13] Taylor L Patti et al. “Markov chain Monte Carlo enhanced variational quantum algorithms”. In: *Quantum Science and Technology* 8.1 (Dec. 2022), p. 015019. DOI: 10.1088/2058-9565/aca821.
- [14] David M. Ceperley. “Path integrals in the theory of condensed helium”. In: *Reviews of Modern Physics* 67.2 (1995), pp. 279–355. DOI: 10.1103/RevModPhys.67.279.
- [15] William MC Foulkes et al. “Quantum Monte Carlo simulations of solids”. In: *Reviews of Modern Physics* 73.1 (2001), p. 33.
- [16] Tobias Hartung and Karl Jansen. “Convergence and efficiency proof of quantum imaginary time evolution for bounded order systems”. In: *arXiv preprint arXiv:2506.03014* (2025).
- [17] Lena Funcke et al. “Dimensional expressivity analysis of parametric quantum circuits”. In: *Quantum* 5 (2021), p. 422.
- [18] Sam McArdle et al. “Variational ansatz-based quantum simulation of imaginary time evolution”. In: *npj Quantum Information* 5.1 (2019), p. 75.

- [19] Hale F Trotter. “On the product of semi-groups of operators”. In: *Proceedings of the American Mathematical Society* 10.4 (1959), pp. 545–551.
- [20] Xiaoyang Wang et al. “Imaginary hamiltonian variational ansatz for combinatorial optimization problems”. In: *Physical Review A* 111.3 (2025), p. 032612.
- [21] Xiaoyang Wang et al. “Symmetry enhanced variational quantum imaginary time evolution”. In: *arXiv preprint arXiv:2307.13598* (2023).
- [22] Karl Jansen et al. “Quasi-Monte Carlo methods for lattice systems: a first look”. In: *Computer Physics Communications* 185.3 (2014), pp. 948–959.
- [23] W Keith Hastings. “Monte Carlo sampling methods using Markov chains and their applications”. In: (1970).
- [24] Nicholas Metropolis et al. “Equation of state calculations by fast computing machines”. In: *The journal of chemical physics* 21.6 (1953), pp. 1087–1092.
- [25] Herbert Robbins and Sutton Monro. “A stochastic approximation method”. In: *The annals of mathematical statistics* (1951), pp. 400–407.