Experimenting with machine learning algorithms on quantum computers

Nagy Dániel[1]

[1]Institute for Physics, Eötvös Loránd University, H-1117, Pázmány Péter sétány 1/A. Budapest, Hungary

Created on December 1, 2019

Last update: May 3, 2021

Abstract

A modern számítástechnika jelentős eredményei közé tartozik a gépi tanulás és mesterséges intelligancia alapvető algoritmusainak kifejlesztése és ezek hasznosságának tesztelése különböző feladatokon. Ugyanakkor az elmúlt években a kvantumszámítás is jelentős fejlődésen ment keresztül, olyannyira, hogy 2019-ben a Google kísérleti csapatának sikerült demonstrálnia a kvantumfölényt. A munka során megvizsgáljuk a két terület átfedéséből származó lehetőségeket: klasszikus adatok kvantumos feldolgozását illetve a klasszikus gépi tanulás segítségével történő kvantumos hibajavítást.

Gépi tanulási algoritmusok vizsgálata kvantumszámítógépeken

Nagy Dániel[1]

[1]Eötvös Loránd Tudományegyetem, Fizika Intézet, H-1117, Pázmány Péter sétány 1/A. Budapest, Magyarország

Elkezdve: December 1, 2019

Utolsó frissítés: 2021. május 3.

Kivonat

A modern számítástechnika jelentős eredményei közé tartozik a gépi tanulás és mesterséges intelligancia alapvető algoritmusainak kifejlesztése és ezek hasznosságának tesztelése különböző feladatokon. Ugyanakkor az elmúlt években a kvantumszámítás is jelentős fejlődésen ment keresztül, olyannyira, hogy 2019-ben a Google kísérleti csapatának sikerült demonstrálnia a kvantumfölényt. A munka során megvizsgáljuk a két terület átfedéséből származó lehetőségeket: klasszikus adatok kvantumos feldolgozását illetve a klasszikus gépi tanulás segítségével történő kvantumos hibajavítást.

Contents

List of Figures

1 Introduction

Quantum computing is probably the most promising emerging technology with many possible applications across all domains of science and business. The idea of quantum computing was first proposed by Richard P. Feynman as a method to simulate quantum mechanics. Since the size of the Hilbert-space grows exponentially with the complexity of the quantum system, and thus the calculations become intractable on any classical computer, Feynmans idea was to simulate quantum physics on devices that behave themselfs according to the rules of quantum physics.

Soon after Feynmans proposal, scientists became to establish the theoretical background of quantum information and quantum computing. Some of the quantum algorithms are proven to have an advantage over any known classical algorithm. One of the first such quantum algorithm is the Deutsch-Jozsa algorithm [3], which decides if a binary function f is balanced or constant. Probably the most notable quantum algorithm was proposed by Peter W. Shor in 1994, which is a polynomial-time quantum algorithm for factoring large integers [19]. This is the first quantum algorithm with a real-life application, because most of the public-key cryptosystems could be broken with an efficient algorithm for integer factoring. Another important quantum algorithm is the Grover's algorithm proposed by Lov K. Grover in 1996 for searching an unordered database [7]. While the best known classical algorithm for searching unordered databases runs in $O(N)$ time, Grover's algorithm solves this problem with $O(\sqrt{N})$ oracle queries. Beyond quantum algorithms, many quantum-inspired algorithms were discovered [20, 4, 1].

2 Quantum Computing with continuous variables

[Dani: cite: Xanadu paper, Photonic Supremacy paper, PsiQuantum Fusion-Based QC]
Continuous variable (CV) systems have infinite degrees of freedom, and therefore the corresponding Hilbert-space is of infinite dimensions. Such systems can be modeled by M harmonic oscillators, which are usually called *modes*. With H_k denoting the Hilbert-space of the kth mode, the Hilbert-space of the full CV system is their direct sum:

$$\mathcal{H} = \bigoplus_{k=1}^{M} \mathcal{H}_k. \tag{1}$$

The Hamiltonian of mode j can be described by introducing bosonic creation and annihilation operators $a_j^\dagger$ and a_j, obeying the following canonical commutation relations:

$$\left[a_j, a_k^\dagger\right] = \delta_{jk}, \left[a_j^\dagger, a_k^\dagger\right] = [a_j, a_k] = 0. \tag{2}$$

Each of the single-mode Hilbert-space $\mathcal{H}_k$ is an infinite-dimensional Fock-space spanned by the eigenvectors of the number operator $n_k = a_k^\dagger a_k$ labeled $\{|n\rangle\}_k$. It is often useful to introduce the canonical position operators X_j and momentum operators P_j which are related to the creation and annihilation

operators by the following transformations:

$$X_j = \frac{a_j + a_j^\dagger}{\sqrt{2}} \qquad P_j = \frac{a_j - a_j^\dagger}{i\sqrt{2}} \tag{3}$$

$$a_j = \frac{X_j + iP_j}{\sqrt{2}} \qquad a_j^\dagger = \frac{X_j - iP_j}{\sqrt{2}} \tag{4}$$

Operators X_j and P_j are called quadrature operators. Following the notation often used in literature, we group these operators into a vector

$$\boldsymbol{\xi}^\top = (X_1, P_1, X_2, P_2, ..., X_j, P_j, ..., X_M, P_M) \tag{5}$$

We can calculate the commutators $[\boldsymbol{\xi}_k, \boldsymbol{\xi}_l]$ by introducing a symplectic form $\boldsymbol{\Omega}$ as

$$\boldsymbol{\Omega} = \bigoplus_{j=1}^{M} \boldsymbol{\omega},\ \boldsymbol{\omega} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}. \tag{6}$$

Using this notation, we can write

$$[\boldsymbol{\xi}_k, \boldsymbol{\xi}_l] = i\boldsymbol{\Omega}_{kl} \tag{7}$$

3 Machine learning

3.1 Classification and Regression

3.2 Reinforcement learning

3.2.1 Proximal Policy optimization

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)} = \log \pi_\theta(a_t|s_t) - \log \pi_{\theta_{\text{old}}}(a_t|s_t) \tag{8}$$

$$\delta_t = r_t + \gamma V^\pi(s_{t+1}) - V^\pi(s_t) \tag{9}$$

$$\hat{A}_t = \sum_{l=0}^{T-t-1} (\gamma\lambda)^l \delta_{t+l} = (\gamma\lambda)^0 \delta_t + (\gamma\lambda)\delta_{t+1} + ... + (\gamma\lambda)^{T-t-1}\delta_{T-1} \tag{10}$$

$$L^{CLIP}(\theta) = \mathop{\mathbb{E}}_t \left[\min\left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t \right) \right] = \mathop{\mathbb{E}}_t \left[\min\left(r_t(\theta)\hat{A}_t, g(\epsilon, \hat{A}_t) \right) \right] \tag{11}$$

$$g(\varepsilon, \hat{A}_t) = \begin{cases} (1+\epsilon)\hat{A}_t, & \hat{A}_t \geq 0 \\ (1-\epsilon)\hat{A}_t, & \text{otherwise} \end{cases} \tag{12}$$

$$V^{\pi}_{\text{targ}}(s_t) = \sum_{l=0}^{T-t} \gamma^l r_{t+l} \tag{13}$$

$$L^{VF} = \mathop{\mathbb{E}}_{t}\left[\left(V^{\pi}(s_t) - V^{\pi}_{\text{targ}}(s_t)\right)^2\right] \tag{14}$$

$$H[\pi] = \mathop{\mathbb{E}}_{t}\left[-\sum_{a\in\mathcal{A}} \pi_\theta(a|s_t)\log\pi_\theta(a|s_t)\right] \tag{15}$$

$$L = L^{CLIP} + c_1 L^{VF} + c_2 H[\pi] \tag{16}$$

How to measure the entropy term for a quantum state ρ? How to measure the KL-divergence of two quantum states ρ_1, ρ_2?

Algorithm 1 PPO-Clip

```
procedure PPOClip(ε, E, N, T, K)
    for all i ∈ {1, ..., E} do
        for all n ∈ {1, ..., N} do
            Run the old policy π_old in the environment for T timesteps.
            for all t ∈ {1, ..., T} do
                Calculate the advantage estimate Â_t
            end for
        end for
        for all k ∈ {1, ..., K} do
            Sample a batch of size M ≤ NT and optimize the surrogate loss L.
        end for
    end for
end procedure
```

4 Quantum Machine Learning

QSVM paper = [16] HHL paper = [8] Feature Hilbert Spaces = [17] Cv-Qnns -LLoyd = [13] [Dani: related papers: quantum svm, Kernel methods, HHL algorithm for matrix inversion, quantum-inspired classical algos

motivation: summary from Lloyd paper, regression, classification, hybrid traning, encoding, inference

General structure: encoding, rotations and beamsplitter = interferometer, squeezing, interferometer, displacement, all = affine transformation + at the end non-linear, e.g., cubic phase operation (numeric instability switch to Kerr gates –> write at the performance test)

Loss functions: regression, classification
Regularization: keep trace close to zero, non-Gaussian
]

4.1 Parametric quantum circuits

4.2 Calculating the gradients of the parameters

In this section, we introduce the theory of calculating quantum gradients in the continuous variable setting. We follow the notation of Shuld et. al. [18]. A variational quantum circuit is defined by a unitary $U(\boldsymbol{\theta})$, $\boldsymbol{\theta} \in \mathbb{R}^m$, an initial state $|\psi_0\rangle$ and a measurment operator B. We can understand a variational citcuit as $f : \mathbb{R}^m \to \mathbb{R}$, **[Dani: The Schuld paper writes $f : \mathbb{R}^m \to \mathbb{R}^n$, maybe a typo???]** which maps the circuit parameters to the expectation value of operator B:

$$f(\boldsymbol{\theta}) = \langle\psi_0| U^\dagger(\boldsymbol{\theta}) B U(\boldsymbol{\theta}) |\psi_0\rangle \tag{17}$$

In gradient-based optimizations, we need the gradient $\nabla_{\boldsymbol{\theta}} f$, and this eventually means calculating the partial derivatives $\partial_\mu f$, $\mu \in \boldsymbol{\theta}$. In the continuous variable setting, observables are usually polynomials of the quadrature operators X_j and P_j.

4.3 Variational Quantum Eigensolver

Variational Quantum Eigensolvers [15] are hybrid quantum-classical algorithms designed to find the lowest energy eigenvalue of some Hamiltonian H. VQEs are one of the most studied quantum algorithms because of their expected use cases in quantum chemistry and materials science [21, 2, 12, 9] The basic idea behind the variational quantum eigensolver is as follows. First we fix the structure of a parametric quantum circuit called the ansatz, defined by a unitary $U(\boldsymbol{\theta})$, where $\boldsymbol{\theta} \in \mathbb{R}^m$ are real parameters. Then we start by some initial state $|\psi_0\rangle$, which is usually $|0\rangle^{\otimes M}$ in the qubit setting. For the continuous variable setting, the initial state can be a displaced, squeezed state, or a photon number eigenstate. In our work we used $|\psi_0\rangle = |1\rangle^{\otimes M}$ for our initial state i.e. we started with a single photon in each mode. After preparing the circuit, we evaluate the expectation value of the Hamiltonian which is eventually the cost, or loss:

$$\mathcal{L}(\boldsymbol{\theta}) = \langle H \rangle = \langle\psi_0| U^\dagger(\boldsymbol{\theta}) H U(\boldsymbol{\theta}) |\psi_0\rangle \tag{18}$$

The goal is to minimize the loss $\mathcal{L}$ by iteratively updating the parameters $\boldsymbol{\theta}$ with some update rule. This update rule can be any gradient-free [22] or gradient-based method **[Dani: idezet]**. To use gradient descent optimization in the VQE setting, we need to evaluate the gradient $\nabla_{\boldsymbol{\theta}} \mathcal{L}$, which can be challenging, and is described in details in chapter **??**. Algorithm 2 describes the variational quantum eigensolver with a general gradient-based learning method. Notice that we wrote $\alpha_j^{(t)}$ for the learning rate, since in most modern optimizers like Adam [14], each parameter has its own learning rate and it

may depend on the number of iterations already done.

Algorithm 2 VQE

procedure VQE(H, K, T, α)
  for all $t \in \{1, ..., T\}$ **do**
    Prepare the ansatz $U(\boldsymbol{\theta})$
    Estimate the loss $\mathcal{L} = \langle\psi_0| U^\dagger(\boldsymbol{\theta}) H U(\boldsymbol{\theta}) |\psi_0\rangle$ by averaging K shots
    for all $\theta_j \in \boldsymbol{\theta}$ **do**
      Calculate the quantum gradient $\partial_{\theta_j}\mathcal{L}$
      Update the parameter $\theta_j \leftarrow \theta_j - \alpha_j^{(t)} \partial_{\theta_j}\mathcal{L}$
    end for
  end for
end procedure

5 Results

5.1 Regression

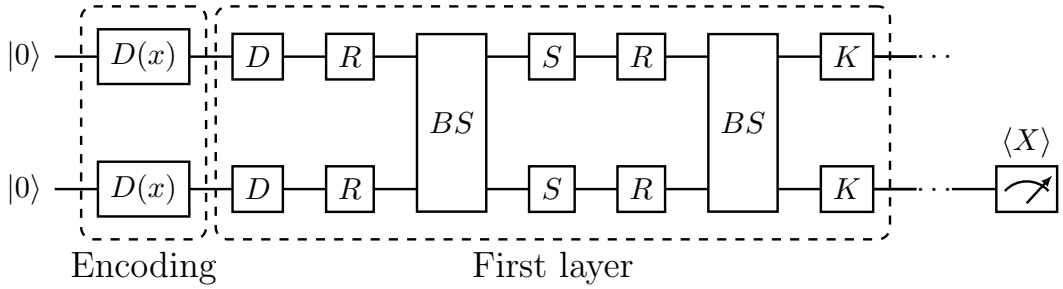


Figure 1.

5.2 Classification

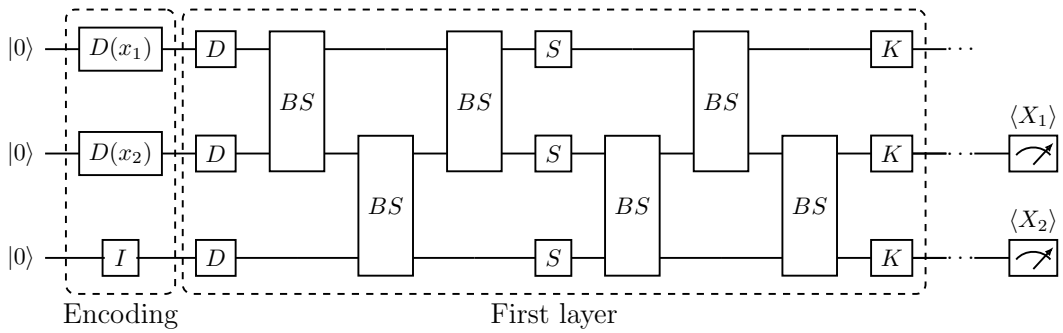


Figure 2.

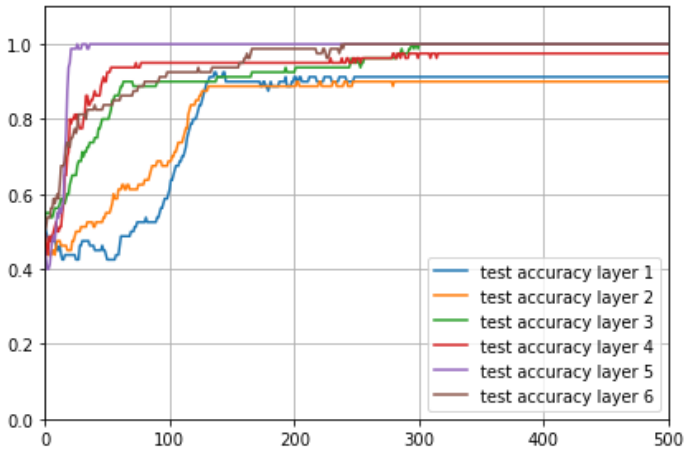


Figure 3.

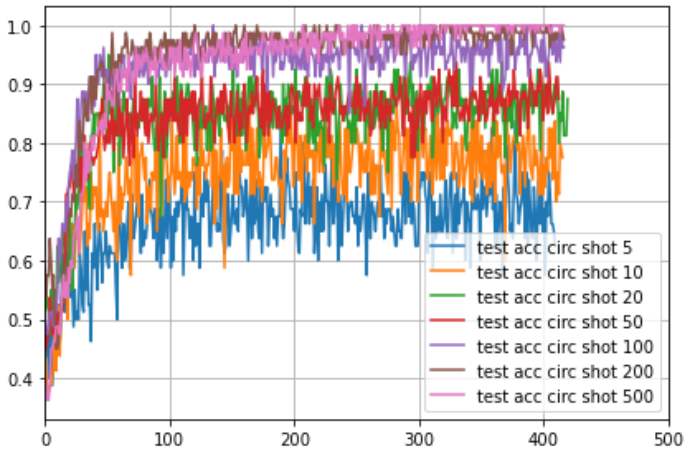


Figure 4.

5.3 CV-VQE for the Bose-Hubbard model

[Dani: Analytic solution by diagonalization] The Bose-Hubbard model introduced first by Gersch and Knollman in 1963 for the description of interacting spinless bosons on a lattice, such as ultracold bosonic atoms on an optical lattice [5, 10]. This model has also gained popularity because it is found to be an adequate model for the description of the Superfluidity-Mott insulator (SF-MI) transition, which has been experimentally demonstrated [11, 6]. We chose this model, because an exact diagonalization of the Bose-Hubbard Hamiltonian with respect to a finite dimensional Fock-basis is possible and thus the model is numerically solvable. This enables the verification of the correctness of the results given by our variational quantum solver for small problem instances.

The Bose Hubbard model which we used in our simulations is defined by the following Hamiltonian expressed with the bosonic creation and annihillation operators:

$$H = -t \sum_{i=1}^{M-1} (a_i^\dagger a_{i+1} + a_{i+1}^\dagger a_i) + \frac{U}{2} \sum_{i=1}^{M} n_i^2, \tag{19}$$

where M is the number of bosonic modes in the system. If we restrict the total number of bosons in the system to be N, the dimension of the restricted Fock-space is

$$D = \frac{(N+M-1)!}{N!(M-1)!}, \tag{20}$$

which explosively grows with the system size. Therefore finding the groundstate of the Bose-Hubbard Hamiltonian for larger systems can be intractable even on classical supercomputers, making quantum computers, especially continuous-variable quantum computers a potential candidate for solving these problems in the future. Compared to the original definition of the Bose-Hubbard Hamiltonian, we omitted the term $-\mu \sum_i^M n_i$, because it is not relevant in our particular experiments.

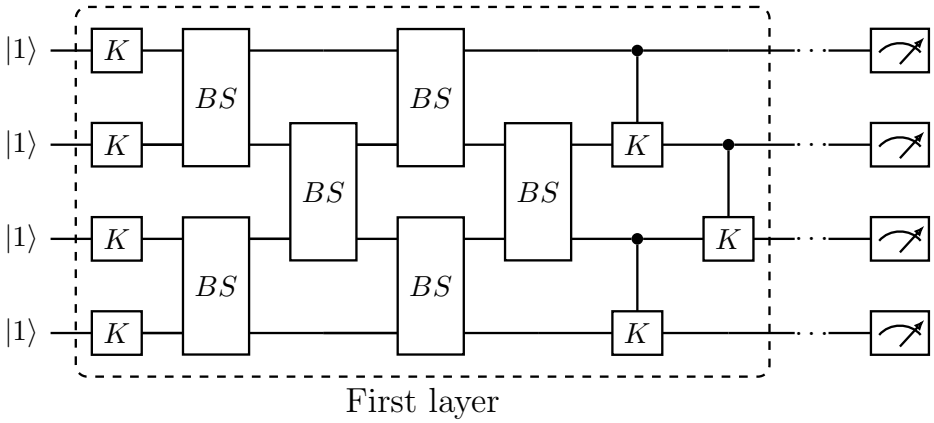


Figure 5. Illustration of a variational ansatz circuit with its first layer. Initially, each of the four qumodes contain a single photon. The layer is built up from Kerr-gates, CrossKerr gates and Beamsplitters, which are passive optical gates, so the total number of photons is preserved.

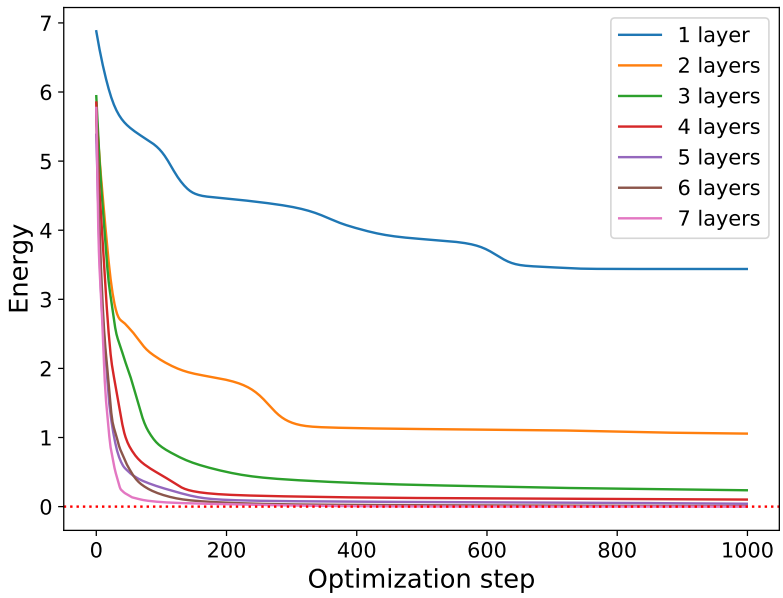
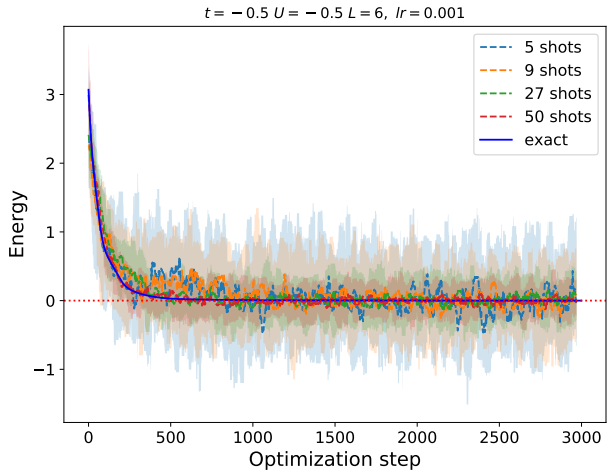
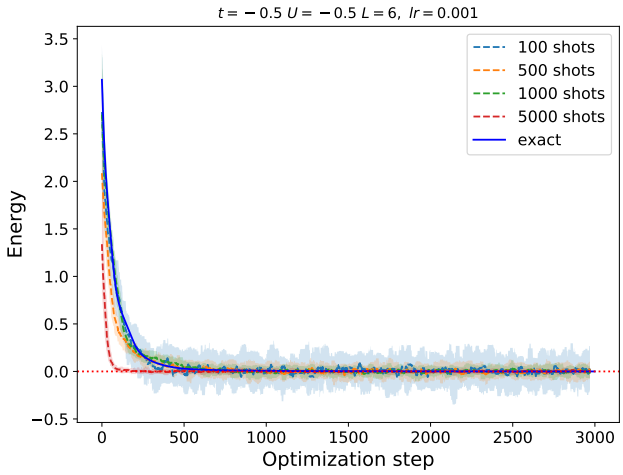


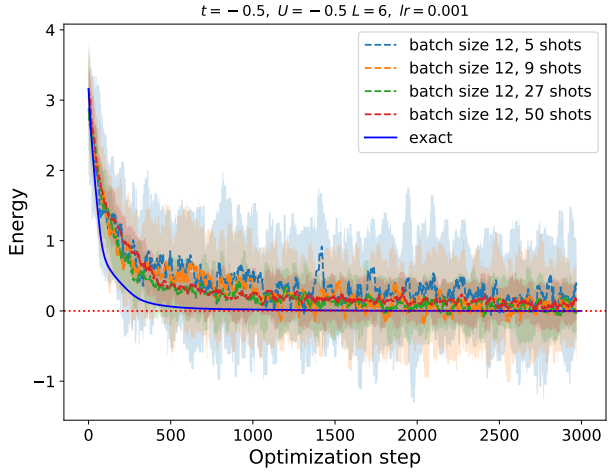
Figure 6.



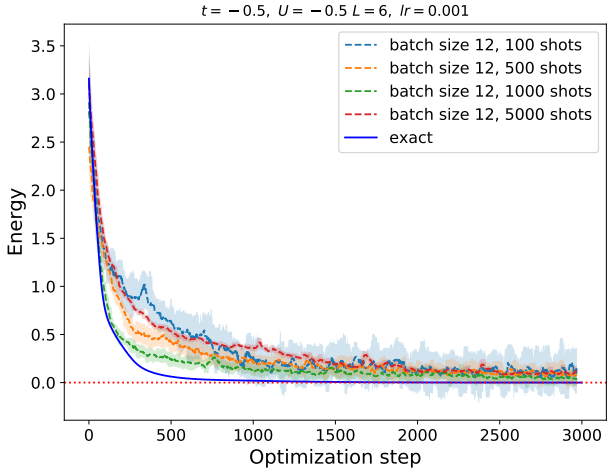
(a) first



(b) second



(c) third



(d) fourth

Figure 7. caption

References

[1] J. M. Arrazola, A. Delgado, B. R. Bardhan, and S. Lloyd. Quantum-inspired algorithms in practice. *arXiv e-prints*, art. arXiv:1905.10415, May 2019.

[2] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, S. Boixo, M. Broughton, B. B. Buckley, D. A. Buell, B. Burkett, N. Bushnell, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, S. Demura, A. Dunsworth, E. Farhi, A. Fowler, B. Foxen, C. Gidney, M. Giustina, R. Graff, S. Habegger, M. P. Harrigan, A. Ho, S. Hong, T. Huang, W. J. Huggins, L. Ioffe, S. V. Isakov, E. Jeffrey, Z. Jiang, C. Jones, D. Kafri, K. Kechedzhi, J. Kelly, S. Kim, P. V. Klimov, A. Korotkov, F. Kostritsa, D. Landhuis, P. Laptev, M. Lindmark, E. Lucero, O. Martin, J. M. Martinis, J. R. McClean, M. McEwen, A. Megrant, X. Mi, M. Mohseni, W. Mruczkiewicz, J. Mutus, O. Naaman, M. Neeley, C. Neill, H. Neven, M. Y. Niu, T. E. O'Brien, E. Ostby, A. Petukhov,

H. Putterman, C. Quintana, P. Roushan, N. C. Rubin, D. Sank, K. J. Satzinger, V. Smelyanskiy, D. Strain, K. J. Sung, M. Szalay, T. Y. Takeshita, A. Vainsencher, T. White, N. Wiebe, Z. J. Yao, P. Yeh, and A. Zalcman. Hartree-fock on a superconducting qubit quantum computer. *Science*, 369 (6507):1084–1089, 2020. ISSN 0036-8075. doi: 10.1126/science.abb9811. URL https://science.sciencemag.org/content/369/6507/1084.

[3] D. Deutsch and R. Jozsa. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 439(1907):553–558, Dec. 1992. doi: 10.1098/rspa.1992.0167. URL https://doi.org/10.1098/rspa.1992.0167.

[4] C. Ding, T.-Y. Bao, and H.-L. Huang. Quantum-Inspired Support Vector Machine. *arXiv e-prints*, art. arXiv:1906.08902, June 2019.

[5] H. A. Gersch and G. C. Knollman. Quantum cell model for bosons. *Phys. Rev.*, 129:959–967, Jan 1963. doi: 10.1103/PhysRev.129.959. URL https://link.aps.org/doi/10.1103/PhysRev.129.959.

[6] M. Greiner, O. Mandel, T. Esslinger, T. W. Hänsch, and I. Bloch. Quantum phase transition from a superfluid to a mott insulator in a gas of ultracold atoms. *Nature*, 415(6867):39–44, Jan. 2002. doi: 10.1038/415039a. URL https://doi.org/10.1038/415039a.

[7] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing - STOC 1996*. ACM Press, 1996. doi: 10.1145/237814.237866. URL https://doi.org/10.1145/237814.237866.

[8] A. W. Harrow, A. Hassidim, and S. Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103:150502, Oct 2009. doi: 10.1103/PhysRevLett.103.150502. URL https://link.aps.org/doi/10.1103/PhysRevLett.103.150502.

[9] C. Hempel, C. Maier, J. Romero, J. McClean, T. Monz, H. Shen, P. Jurcevic, B. P. Lanyon, P. Love, R. Babbush, A. Aspuru-Guzik, R. Blatt, and C. F. Roos. Quantum chemistry calculations on a trapped-ion quantum simulator. *Phys. Rev. X*, 8:031022, Jul 2018. doi: 10.1103/PhysRevX.8.031022. URL https://link.aps.org/doi/10.1103/PhysRevX.8.031022.

[10] D. Jaksch and P. Zoller. The cold atom Hubbard toolbox. *Annals of Physics*, 315(1):52–79, Jan. 2005. doi: 10.1016/j.aop.2004.09.010.

[11] D. Jaksch, C. Bruder, J. I. Cirac, C. W. Gardiner, and P. Zoller. Cold bosonic atoms in optical lattices. *Phys. Rev. Lett.*, 81:3108–3111, Oct 1998. doi: 10.1103/PhysRevLett.81.3108. URL https://link.aps.org/doi/10.1103/PhysRevLett.81.3108.

[12] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242–246, Sept. 2017. doi: 10.1038/nature23879. URL https://doi.org/10.1038/nature23879.

[13] N. Killoran, T. R. Bromley, J. M. Arrazola, M. Schuld, N. Quesada, and S. Lloyd. Continuous-variable quantum neural networks. *Phys. Rev. Research*, 1:033063, Oct 2019. doi: 10.1103/PhysRevResearch.1.033063. URL https://link.aps.org/doi/10.1103/PhysRevResearch.1.033063.

[14] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

[15] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1), July 2014. doi: 10.1038/ncomms5213. URL https://doi.org/10.1038/ncomms5213.

[16] P. Rebentrost, M. Mohseni, and S. Lloyd. Quantum support vector machine for big data classification. *Phys. Rev. Lett.*, 113:130503, Sep 2014. doi: 10.1103/PhysRevLett.113.130503. URL https://link.aps.org/doi/10.1103/PhysRevLett.113.130503.

[17] M. Schuld and N. Killoran. Quantum machine learning in feature hilbert spaces. *Phys. Rev. Lett.*, 122:040504, Feb 2019. doi: 10.1103/PhysRevLett.122.040504. URL https://link.aps.org/doi/10.1103/PhysRevLett.122.040504.

[18] M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, and N. Killoran. Evaluating analytic gradients on quantum hardware. *Phys. Rev. A*, 99:032331, Mar 2019. doi: 10.1103/PhysRevA.99.032331. URL https://link.aps.org/doi/10.1103/PhysRevA.99.032331.

[19] P. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*. IEEE Comput. Soc. Press. doi: 10.1109/sfcs.1994.365700. URL https://doi.org/10.1109/sfcs.1994.365700.

[20] E. Tang. A quantum-inspired classical algorithm for recommendation systems. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. ACM, June 2019. doi: 10.1145/3313276.3316310. URL https://doi.org/10.1145/3313276.3316310.

[21] S. Wei, H. Li, and G. Long. A full quantum eigensolver for quantum chemistry simulations. *Research*, 2020:1–11, Mar. 2020. doi: 10.34133/2020/1486935. URL https://doi.org/10.34133/2020/1486935.

[22] D. Zhu, N. M. Linke, M. Benedetti, K. A. Landsman, N. H. Nguyen, C. H. Alderete, A. Perdomo-Ortiz, N. Korda, A. Garfoot, C. Brecque, L. Egan, O. Perdomo, and C. Monroe. Training of quantum circuits on a hybrid quantum computer. *Science Advances*, 5(10):eaaw9918, Oct. 2019. doi: 10.1126/sciadv.aaw9918. URL https://doi.org/10.1126/sciadv.aaw9918.

A Mathematical preliminaries

A.1 Hilbert spaces

Definition 1. ***Hilbert-space***
Given a field T (real or complex), a vector space $\mathcal{H}$ endowed with an inner product, is called a Hilbert-space, if it is a complete metric space with respect to the distance function induced by the inner product. The inner product is a map $\langle\cdot|\cdot\rangle : \mathcal{H} \times \mathcal{H} \to T$, for which $\forall x, y, z \in \mathcal{H}$:

- $\langle x|x\rangle \geq 0$
- $\langle x|x\rangle = 0 \iff x = \mathbf{0} \in \mathcal{H}$
- $\langle x|y\rangle = \langle y|x\rangle^*$, *where* * *denotes complex conjugation.*
- $\langle x|\alpha y + \beta z\rangle = \alpha \langle x|y\rangle + \beta \langle x|z\rangle$, *where* $\alpha, \beta \in T$

The norm induced by this inner product is a map $||\cdot|| : \mathcal{H} \to T$ defined as

$$||x|| = \sqrt{\langle x|x\rangle},$$

And the metric induced by this norm is defined as

$$d(x, y) = ||x - y|| = \sqrt{\langle x - y|x - y\rangle}.$$

The space $\mathcal{H}$ is said to be complete if every Cauchy-sequence is convergent with respect to the norm, and the limit is in $\mathcal{H}$. That is, each sequence $x_1, x_2, ...$, for which

$$\forall \varepsilon > 0 \ \exists N(\varepsilon) \text{ so, that } n > m > N(\varepsilon) \implies ||x_n - x_m|| < \varepsilon.$$

Definition 2. ***Linear functional***
Let $\mathcal{H}$ be a Hilbert-space over the field T. Then, the map $\varphi : \mathcal{H} \to T$ is called a linear functional, if

$$\varphi(\alpha x + \beta y) = \alpha\varphi(x) + \beta\varphi(y), \ \forall \alpha, \beta \in T, \ x, y \in \mathcal{H}.$$

Definition 3. ***Dual space***
Given a Hilbert-space $\mathcal{H}$, its dual space, $\mathcal{H}^*$ is the space of all continuous linear functionals from the space H into the base field. The norm of an element in $\mathcal{H}^*$ is

$$||\varphi||_{\mathcal{H}^*} \stackrel{def}{=} \sup_{||x||=1, \, x \in \mathcal{H}} |\varphi(x)|.$$

Theorem 1. ***Riesz representation theorem***

For every element $y \in \mathcal{H}$, there exists a unique element $\varphi_y \in \mathcal{H}^*$, defined by

$$\varphi_y(x) = \langle y|x\rangle \,,\ \forall x \in \mathcal{H}.$$

The mapping $y \mapsto \varphi_y$ is an antilinear mapping i.e. $\alpha y_1 + \beta y_2 \mapsto \alpha^* \varphi_{y_1} + \beta^* \varphi_{y_2}$, and the Riesz-representation theorem states that this mapping is an antilinear isomorphism. The inner product in $\mathcal{H}^*$ satisfies

$$\langle \varphi_x | \varphi_y \rangle = \langle x|y\rangle^* = \langle y|x\rangle \,.$$

Moreover, $||y||_{\mathcal{H}} = ||\varphi_y||_{\mathcal{H}^*}$.

Definition 4. ***Dirac-notation***
From now on, the elements in $\mathcal{H}$ will be denoted by $|x\rangle$ and their corresponding element in $\mathcal{H}^*$ as $\langle x|$.

A.2 Linear operators on Hilbert spaces

Definition 5. ***Linear operators***
A map $\hat{A} : \mathcal{H}_1 \to \mathcal{H}_2$ is a linear operator, if

$$\hat{A}(\alpha\,|x\rangle + \beta\,|y\rangle) = \alpha(\hat{A}\,|x\rangle) + \beta(\hat{A}\,|y\rangle).$$

Remark 1. *If not stated otherwise, we will assume that $\mathcal{H}_1 = \mathcal{H}_2 = \mathcal{H}$.*

Remark 2. *Operators will be denoted with a hat ($\hat{\cdot}$).*

Definition 6. ***Bounded linear operators***
A linear operator $\hat{A} : \mathcal{H} \to \mathcal{H}$ is bounded, if

$$\exists m \in \mathbb{R} : |\,\langle v|\hat{A}|v\rangle\,| \leq m\,\langle v|v\rangle \,,\ \forall\,|v\rangle \in \mathcal{H}$$

Remark 3. *The set of all bounded operators on $\mathcal{H}$ is denoted $\mathcal{B}(\mathcal{H})$.*

Definition 7. ***Commutators and anticommutators***
Since operators usually do not commute, its useful to define their commutator and anticommutator:

$$[\hat{A}, \hat{B}] = \hat{A}\hat{B} - \hat{B}\hat{A}$$
$$\{\hat{A}, \hat{B}\} = \hat{A}\hat{B} + \hat{B}\hat{A}$$

Definition 8. ***Operator norm***
The operator norm of an operator $\hat{A}$ is defined as

$$||\hat{A}|| \overset{def}{=} \inf\{c \geq 0 : ||\hat{A}\,|v\rangle\,|| \leq c||\,|v\rangle\,||,\ \forall\,|v\rangle \in \mathcal{H}\}$$

Definition 9. ***Trace-class operators***
An operator $\hat{A}$ is called trace-class if it admits a well defined and finite trace $\mathrm{Tr}\left[\hat{A}\right] = \sum_j \langle j|\hat{A}|j\rangle$

Definition 10. ***Positive operators***
An operator $\hat{A}$ is called positive if $\langle v|\hat{A}|v\rangle \geq 0, \forall\, |v\rangle \in \mathcal{H}$. If $\hat{A} = \sum_j \lambda_j |j\rangle \langle j|$ then $\hat{A}$ is positive if $\lambda_j \geq 0$.

Definition 11. ***Projections*** *An operator $\Pi : \mathcal{H} \to \mathcal{H}$ is a projection if $\Pi^2 = \Pi$.*

A.3 Hermitian Operators, Unitary Operators, Spectral theorem, Hadamard-lemma

Definition 12. ***Hermitian adjoint***
Consider a ***bounded*** *linear operator $\hat{A} : \mathcal{H} \to \mathcal{H}$. The hermitian adjoint of $\hat{A}$ is a bounded linear operator $\hat{A}^\dagger : \mathcal{H} \to \mathcal{H}$ which satisfies*

$$\langle y|\, \hat{A}\, |x\rangle = \left(\langle x|\, \hat{A}^\dagger\, |y\rangle\right)^*,\ \forall\, |x\rangle\,, |y\rangle \in \mathcal{H}. \tag{21}$$

Definition 13. ***Hermitian operators***
A bounded linear operator $\hat{H} : \mathcal{H} \to \mathcal{H}$ is Hermitian if

$$\hat{H} = \hat{H}^\dagger,\ \textit{i.e. } \hat{H}\, |x\rangle = \hat{H}^\dagger\, |x\rangle\,,\ \forall\, |x\rangle \in \mathcal{H}. \tag{22}$$

Definition 14. ***Unitary operator***
A bounded linear operator $\hat{U} : \mathcal{H} \to \mathcal{H}$ is unitary if

$$\hat{U}\hat{U}^\dagger = \hat{U}^\dagger\hat{U} = 1,\ \textit{in other words, } \hat{U}^\dagger = \hat{U}^{-1}. \tag{23}$$

Definition 15. ***Eigenvalues and eigenvectors***
Consider bounded linear operator $\hat{A}$. If exist a vectors $|k\rangle \in \mathcal{H}$ such that

$$\hat{A}\, |k\rangle = \lambda_k\, |k\rangle\,, \tag{24}$$

then $|k\rangle$ is called an eigenvector of $\hat{A}$ and λk is the corresponding eigenvalue.

An important property of Hermitian operators is that they can be diagonalized with real eigenvalues. This is formally stated by the spectral theorem:

Theorem 2. ***The Spectral theorem***
Let $\hat{A}$ be a bounded Hermitian operator on some Hilbert-space $\mathcal{H}$. Then there exists an orthonormal basis in $\mathcal{H}$ which consists of the eigenvectors of $\hat{A}$ and each eigenvalue of $\hat{A}$ is real.

This means that any bounded Hermitian operator $\hat{H}$ can be decomposed as

$$\hat{H} = \sum_k \lambda_k \hat{P}_k = \sum_k \lambda_k |k\rangle\langle k| \tag{25}$$

where λ_k and $|k\rangle$ are the eigenvalues and eigenvectors of $\hat{H}$.

Definition 16. ***Exponential of operators*** *If X is a linear operator, we can define the exponential of X:*

$$e^X = \sum_{n=0}^{\infty} \frac{X^n}{n!}$$

Important: The product of exponentials of operators generally isn't equal to the exponential of their sum:

$$e^X e^Y = e^{Z(X,Y)} \neq e^{X+Y},$$

where $Z(X,Y)$ is given by the Baker-Campbell-Hausdorff formula:

$$\begin{aligned} Z(X,Y) = X + Y + \frac{1}{2}[X,Y] + \frac{1}{12}[X,[X,Y]] - \frac{1}{12}[Y,[X,Y]] - \frac{1}{24}[Y,[X,[X,Y]]] \\ - \frac{1}{720}([[[[X,Y],Y],Y],Y] + [[[Y,X],X],X],X]) + ... \end{aligned}$$

It is however equal if $[X,Y] = 0$:

$$\text{if } [X,Y] = 0 \implies e^X e^Y = e^{X+Y}$$

There are 2 important special cases:

Theorem 3. ***The Hadamard-lemma***

$$e^X Y e^{-X} = Y + [X,Y] + \frac{1}{2!}[X,[X,Y]] + \frac{1}{3!}[X,[X,[X,Y]]] + ...$$

Theorem 4. *If X and Y commute with their commutator, i.e. $[X,[X,Y]] = [Y,[X,Y]] = 0$, then:*

$$e^X e^Y = e^{X+Y+\frac{1}{2}[X,Y]}$$

Theorem 5. *If $[X,Y] = sY$ with $s \in \mathbb{C}, s \neq 2i\pi n, n \in \mathbb{Z}$ then:*

$$e^X e^Y = \exp\left(X + \frac{s}{1-e^{-s}}Y\right)$$

A.4 Pure and mixed quantum states

Definition 17. ***Quantum states***
A quantum state of a quantum system is a mathematical entity that provides a probability distribution for the outcomes of each possible measurement on the system.

Definition 18. ***Pure quantum states***
Pure quantum states are quantum states that can be described by a vector $|\psi\rangle$ of norm 1.

If one multiplies a pure quantum state by a complex scalar $e^{i\alpha}$, then the new state is physically equivalent to the former, thus $|\psi\rangle$ and $e^{i\alpha}|\psi\rangle$ are the same pure state. The transformation $|\psi\rangle \to e^{i\alpha}|\psi\rangle$

does not change the outcomes of measurements on the state, however the phase α is important in quantum algorithms.

Example. *For example, the states* $\frac{1}{\sqrt{2}}(|0\rangle + e^{i\pi}|1\rangle)$ *and* $\frac{1}{\sqrt{2}}(|0\rangle + e^{i\frac{\pi}{2}}|1\rangle)$ *are not the same quantum state, but in both states there is 50-50 percent probability of measuring* $|0\rangle$ *and* $|1\rangle$*.*

Definition 19. ***Density Matrix***
A quantum state $\hat{\rho}$ *is a trace-1, self-adjoint, positive semidefinite operator. The set of quantum states is*

$$\mathcal{S}(\mathcal{H}) = \{\hat{\rho} : \hat{\rho} \geq 0, \hat{\rho} = \hat{\rho}^\dagger, \mathrm{Tr}\,[\hat{\rho}] = 1\}$$

A quantum state is pure if and only if $\hat{\rho}^2 = \hat{\rho}$*. Also, if* ρ *is a pure state, then it can be written as* $\hat{\rho} = |\psi\rangle\langle\psi|$*. The operator* ρ *is called the density operator or density matrix.*