Centre for Quantum Technologies
National University of Singapore
Singapore 117543

19th May 2021

The Editorial Team
SciPost Physics

Dear Editor(s),

Hereby we would like to submit our manuscript "NISQ Algorithm for Hamiltonian Simulation via Truncated Taylor Series" co-authored by Jonathan Wei Zhong Lau, Tobias Haug, Leong Chuan Kwek and Kishor Bharti for publication in SciPost Physics.

Quantum simulation with classical computers is fundamentally limited by the exponentially growing Hilbert space of the underlying quantum systems. Even with better classical computers, past a few 10s of particles, it will be impossible to model them without resorting to approximations to cut down the dimensionality of the problem. However, many important problems in physics, which remain poorly understood, will benefit from the ability to conduct such simulations, especially in the fields of condensed matter and low-temperature physics. Feynman suggested to use quantum computing devices to simulate such quantum phenomenon, and much work has been done in recent years to realize this. Recently with the development of digital quantum computers, and especially *noisy intermediate-scale quantum* (NISQ) computers/devices, the power of quantum computing has been demonstrated with simple quantum supremacy experiments, such as those performed by Google. However, practical uses of such devices have yet to be seen. It is hoped that those devices can be applied to such quantum simulation problems. By doing so, it will also be a demonstration that NISQ devices do have practical use-cases.

To harness the NISQ devices' potential for quantum simulation, many near-term quantum simulation algorithms have been proposed. Most of the algorithms currently developed make use of a classical-quantum feedback loop. For example, Variational Quantum Simulation (VQS), as described in the work "Variational quantum simulation of general processes" (Physical Review Letters 125 (1), 010501), and Variational Fast Forwarding (VFF), as described in "Variational fast forwarding for quantum simulation beyond the coherence time" (npj Quantum Information 6 (1), 1-10), are two such examples. Such algorithms make use of of a parametric quantum circuit, which is a quantum circuit with parameterized 1-qubit and 2-qubit rotation gates, as their ansatz, and aim to optimize the parameters in a certain fashion. However, these algorithm suffers from some limitations. They all employ a classical optimizer to update the parameters of a parametric quantum circuit in an adaptive feedback fashion. This classical-quantum feedback loop makes such variational approaches very slow as the quantum computer has to wait for the classical computer's output at every step, especially if the quantum computer is accessed on a cloud in a queue system. Moreover, it has been shown that variational approaches based on parameterizing a quantum circuit suffer from a fundamental trainability problem known as the *barren plateau problem*, which limits the effectiveness of these algorithms. Another method to perform such simulations, by utilizing the Trotter-Suzuki decomposition of the unitary time evolution operator into small discrete steps, is limited by the fact that past a few steps, the circuit length grows too long to be implemented on current devices.

In our work, we present a new algorithm that circumvents all of the aforementioned weaknesses of the currently published variational approaches, such as the existence of the classical-quantum feedback loop and the barren plateau problem. On current quantum computers, our algorithm requires significantly less quantum resources than the alternatives. For example, to simulate a similar 2 qubit system, the best competing alternatives require around 4000 circuits to be evaluated on the quantum computer, while our algorithm only requires around 30 circuits, while retaining similar performance. We also show how our algorithm permits systematic improvements to the ansatz, while other algorithms do not have similar systematic methods to improve on an ansatz.

We believe that our work will be of high interest to the readership of SciPost Physics and ongoing investigations concerning quantum advantage for quantum simulation in the NISQ era.

Sincerely,

Jonathan Wei Zhong Lau, Tobias Haug, Leong Chuan Kwek and Kishor Bharti

NISQ Algorithm for Hamiltonian Simulation via Truncated Taylor Series

Jonathan Wei Zhong Lau,[1, *] Tobias Haug,[2] Leong Chuan Kwek,[1, 3, 4] and Kishor Bharti[1, †]

[1]*Centre for Quantum Technologies, National University of Singapore 117543, Singapore*
[2]*QOLS, Blackett Laboratory, Imperial College London SW7 2AZ, UK*
[3]*National Institute of Education, Nanyang Technological University, 1 Nanyang Walk, Singapore 637616*
[4]*MajuLab, CNRS-UNS-NUS-NTU International Joint Research Unit, UMI 3654, Singapore*

Simulating the dynamics of many-body quantum systems is believed to be one of the first fields that quantum computers can show a quantum advantage over classical computers. Noisy intermediate-scale quantum (NISQ) algorithms aim at effectively using the currently available quantum hardware. For quantum simulation, various types of NISQ algorithms have been proposed with individual advantages as well as challenges. In this work, we propose a new algorithm, truncated Taylor quantum simulator (TQS), that shares the advantages of existing algorithms and alleviates some of the shortcomings. Our algorithm does not have any classical-quantum feedback loop and bypasses the barren plateau problem by construction. The classical part in our hybrid quantum-classical algorithm corresponds to a quadratically constrained quadratic program (QCQP) with a single quadratic equality constraint, which admits a semidefinite relaxation. The QCQP based classical optimization was recently introduced as the classical step in quantum assisted eigensolver (QAE), a NISQ algorithm for the Hamiltonian ground state problem. Thus, our work provides a conceptual unification between the NISQ algorithms for the Hamiltonian ground state problem and the Hamiltonian simulation. We recover differential equation-based NISQ algorithms for Hamiltonian simulation such as quantum assisted simulator (QAS) and variational quantum simulator (VQS) as particular cases of our algorithm. We test our algorithm on some toy examples on current cloud quantum computers. We also provide a systematic approach to improve the accuracy of our algorithm.

Digital quantum computers have made immense progress in recent years, advancing to solving problems considered to take an unreasonable time to compute for classical computers [1, 2]. Further, quantum computers are reaching the stage where quantum chemistry problems such as finding the ground state of certain molecules can be achieved within chemical accuracy [3]. In short, we are now in the Noisy Intermediate-Scale Quantum (NISQ) era [4, 5], which is characterized by quantum computers with up to a few hundred noisy qubits and lacking full scale quantum error correction. Thus, noise will limit the usefulness of the computations carried out by these computers [4], preventing algorithms that offer quantum advantage for practical problems, such as Shor's algorithm for prime factorization [6], from being implemented.

However, just because such algorithms cannot be implemented on NISQ devices does not mean that quantum advantage for practical problems cannot be found with NISQ devices. There is currently great interest in the quantum computing and quantum information community to develop algorithms that can be run on NISQ devices but yet deal with problems that are practical [5, 7, 8]. Some of the most promising avenues deal with the problems in many-body physics and quantum chemistry. One major problem in this field is to develop algorithms capable of estimating the ground state and energy of many-body Hamiltonians. To such ends, algo-

rithms like variational quantum eigensolver (VQE) [9, 10] and quantum assisted eigensolver (QAE) [11, 12] have been proposed.

The other major problem is to be able to simulate the dynamics of these many-body Hamiltonians. This task can be extremely challenging for classical computers, and Feynman proposed that this would be one of the areas where quantum computers could exhibit clear advantages over classical computers [13]. Powerful methods to simulate quantum dynamics on fault-tolerant quantum computers have been proposed, such as the concept of using truncated Taylor series by Berry et al [14].

On NISQ devices, a standard approach in simulating quantum dynamics is to utilize the Trotter-Suzuki decomposition of the unitary time evolution operator into small discrete steps. Each step is made up of efficiently implementable quantum gates, which can be run on the quantum computer [15–21]. However, the depth of the quantum circuit increases linearly with evolution time and the desired target accuracy. On NISQ devices, fidelity rapidly decreases after a few Trotter steps [22], implying long time scales will be unfeasible to simulate with this method. Alternatives to Trotterization have been proposed, such as VQS [23–25], subspace variational quantum simulator (SVQS) [26], variational fast forwarding (VFF) [27, 28], fixed state variational fast forwarding (fsVFF) [29], quantum assisted simulator [30, 31] and generalized quantum assisted simulator (GQAS) [32] to name a few.

Recently, Otten, Cortes and Gray have proposed the idea of restarting the dynamics after every timestep by approximating the wavefunction with a variational

_____

[*] e0032323@u.nus.edu
[†] kishor.bharti1@gmail.com

ansatz [33]. Building on that, Barison, Vicentini and Carleo have proposed a new algorithm [34] for simulating quantum dynamics. Their algorithm, named projected variational quantum dynamics (pVQD) combines the Trotterization and VQS approaches [23, 24]. They replace the differential equation with an optimization problem, although not well characterized, and require much simpler circuits compared to VQS. However, pVQD requires a quantum-classical feedback loop and might suffer from the barren plateau problem [35] as well the optimization problem may be computationally hard [36]. Further, the feedback loop mandates that one has to wait for each computation to finish before the next computation is run, which can be a major bottleneck on cloud-based quantum computers that are accessed via a queue.

Here, we propose the truncated Taylor quantum simulator (TQS) as new algorithm to simulate quantum dynamics. Our algorithm is building on the ideas of pVQD [33, 34] combined with the ansatz generation of QAS [30], which we further enhance by applying the concept of truncated Taylor series by Berry et al [14]. The contributions of the paper and our algorithm are as following:

1. We recast the simulation of the quantum dynamics as a quadratically constrained quadratic program (QCQP). This optimization problem, unlike the optimization problem in pVQD, is well characterized and invites rigorous analysis. The QCQP in our algorithm admits a semidefinite relaxation [11]. Moreover, based on ideas from [11], one can provide a sufficient condition for a local minimum to be a global minimum, which a solver can further use as a stopping criterion. Since the classical optimization program in QAE is also a QCQP, it helps us achieve a conceptual unification of TQS with QAE.

2. The differential equations which form the classical part of QAS and VQS can be recovered in our framework. Since VQS is already a particular case of QAS [30], our approach yields both VQS and QAS as special cases of TQS.

3. We remove the need for the classical-quantum feedback loop in pVQD. The absence of the feedback loop yields our algorithm to be exceptionally faster than the feedback loop based NISQ algorithms for simulating quantum dynamics such as [23, 26–29]. The choice of problem-aware ansatz and the structure of the TQS algorithm helps bypass the barren plateau problem.

TQS Approach— Let us first assume that the Hamiltonian H is expressed as a linear combination of r tensored Pauli matrices

$$H = \sum_{i=1}^{r} \beta_i P_i, \qquad (1)$$

with coefficients $\beta_i \in \mathbb{C}$. The unitary evolution under the action of the Hamiltonian H for time Δt is given by

$$U(\Delta t) = \exp(-\iota H \Delta t) = \exp\left(-\iota \Delta t \sum_{j=1}^{r} \beta_j P_j\right) \quad (2)$$

$$= I - \iota \Delta t \left(\sum_{j=1}^{r} \beta_j P_j\right) - \frac{\Delta t^2}{2} \left(\sum_{j=1}^{r} \beta_j P_j\right)^2 + \mathcal{O}(\Delta t^3). \tag{3}$$

We do not need to implement the action of the unitary evolution in such a way. However, for purposes of describing the algorithm for the rest of the paper, we will use this power series expansion first, and talk more about alternatives later. We will now truncate the series, similar to [14]. If we choose small values of Δt with respect to the eigen energies of H, we can approximate the unitary evolution with $V(\Delta t)$

$$U(\Delta t) \approx I - \iota \Delta t \left(\sum_{j=1}^{r} \beta_j P_j\right) \equiv V(\Delta t). \qquad (4)$$

Let us next choose the ansatz at time t as linear combination of elements from cumulative K-moment states, $\mathbb{CS}_K$ (refer to the QAS paper [30] for the formal definition). These states are defined in the same way as in [30] and will be constructed with the help of the given Hamiltonian. Given a set of r tensored Pauli unitary matrices obtained from the unitary terms of the Hamiltonian $\mathcal{P} \equiv \{P_i\}_{i=1}^{r}$ and a positive integer K and some efficiently preparable quantum state $|\psi\rangle$, the K-moment states are the set of quantum states of the form

$$\{|\chi_i\rangle\}_i = \{P_{iK} \dots P_{i2} P_{i1} |\psi\rangle\}_{iK,\dots,i2,i1}, \qquad (5)$$

for $P_{il} \in \mathcal{P}$. This set is denoted by $\mathbb{S}_K$. The cumulative K-moment states $\mathbb{CS}_K$ are also defined in [30] as $\mathbb{CS}_K \equiv \cup_{j=0}^{K} \mathbb{S}_j$.

Now the ansatz is expressed as

$$|\psi(\alpha(t))\rangle_K = \sum_{|\chi_i\rangle \in \mathbb{CS}_K} \alpha_i(t)|\chi_i\rangle, \qquad (6)$$

with some $\alpha_i \in \mathbb{C}$. For small values of Δt, the ansatz at time $t + \Delta t$ is given by

$$|\psi(\alpha(t+\Delta t))\rangle_K = \frac{V(\Delta t)|\psi(\alpha(t))\rangle_K}{(\langle\psi(\alpha(t))|_K V^\dagger(\Delta t) V(\Delta t)|\psi(\alpha(t))\rangle_K)^{\frac{1}{2}}}. \qquad (7)$$

Using the ideas in [34], our goal now is to variationally approximate the time evolution of the system by adjusting our variational parameters. The crucial difference in our case is that our variational parameters α are coefficients which do not change the basis quantum states $|\chi_i\rangle$. Thus, they can be solely updated via a classical computer and do not require a quantum-classical feedback

loop. To evolve by time Δt, we update the α_i parameters to α_i' such that the following fidelity measure is maximized

$$F\left(\alpha'\right) = \frac{\left|\langle\psi\left(\alpha'\right)|_K V\left(\Delta t\right)|\psi\left(\alpha\right)\rangle_K\right|^2}{\langle\psi\left(\alpha\right)|_K V^\dagger\left(\Delta t\right) V\left(\Delta t\right)|\psi\left(\alpha\right)\rangle_K} \quad (8)$$

Using the notation $|\phi\rangle = V\left(\Delta t\right)|\psi\left(\alpha\right)\rangle_K$, the expression for fidelity becomes

$$F\left(\alpha'\right) = \frac{\langle\psi\left(\alpha'\right)|\phi\rangle_K\langle\phi|\psi\left(\alpha'\right)\rangle_K}{\langle\phi|\phi\rangle}. \quad (9)$$

Using the notation $W_\phi \equiv \frac{|\phi\rangle\langle\phi|}{\langle\phi|\phi\rangle}$, the above expression further simplifies to

$$F\left(\alpha'\right) = \langle\psi\left(\alpha'\right)|_K W_\phi|\psi\left(\alpha'\right)\rangle_K. \quad (10)$$

The goal is to maximize the fidelity subject to the constraint that $\langle\psi\left(\alpha'\right)|\psi\left(\alpha'\right)\rangle = 1$. Thus, the optimization program at timestep t is given by

$$\max_{\alpha'} \ \langle\psi\left(\alpha'\right)|_K W_\phi|\psi\left(\alpha'\right)\rangle_K$$
$$\text{s.t.} \ \langle\psi\left(\alpha'\right)|\psi\left(\alpha'\right)\rangle_K = 1. \quad (11)$$

Using the elements from $\mathbb{CS}_K$ and the Hamiltonian H, we define the overlap matrices $\mathcal{E}$ and $\mathcal{D}$ as the following

$$\mathcal{E}_{m,n} = \langle\chi_m|\chi_n\rangle, \quad (12)$$
$$\mathcal{D}_{m,n} = \sum_j \beta_j\langle\chi_m|P_j|\chi_n\rangle. \quad (13)$$

Because of the way the $|\chi_n\rangle$ states are constructed, these values can be easily computed on a quantum computer, as they simplify to the expectation values of Pauli strings acting on the original quantum state $|\psi\rangle$. The constraint in the optimization program 11 can written in terms of α' as

$$\alpha'^\dagger \mathcal{E}\alpha' = 1. \quad (14)$$

We proceed to write the objective in the optimization program 11 in terms of the overlap matrices $\mathcal{E}$ and $\mathcal{D}$. In first order, we can simplify the expression

$$\langle\phi|\phi\rangle = \langle\psi(\alpha)|_K \left(I + (\Delta t)^2 H^2\right)|\psi(\alpha)\rangle_K$$
$$= \alpha^\dagger\mathcal{E}\alpha + O((\Delta t)^2) \approx \alpha^\dagger\mathcal{E}\alpha. \quad (15)$$

Further, using the notation $G \equiv (\mathcal{E} - \iota\Delta t\mathcal{D})$ we find

$$\langle\psi\left(\alpha'\right)|\phi\rangle_K\langle\phi|\psi\left(\alpha'\right)\rangle_K = \alpha'^\dagger G\alpha\alpha^\dagger G^\dagger\alpha'. \quad (16)$$

Using Eq.14,15,16 and the notation $W_\alpha \equiv \frac{G\alpha\alpha^\dagger G^\dagger}{\alpha^\dagger\mathcal{E}\alpha}$, the optimization program in 11 can be re-expressed in terms of overlap matrices as

$$\max_{\alpha'} \ \alpha'^\dagger W_\alpha\alpha' \quad (17)$$
$$\text{s.t} \ \alpha'^\dagger\mathcal{E}\alpha' = 1. \quad (18)$$

The aforementioned optimization program is a quadratically constrained quadratic program with a single equality constraint. As described in [11], this QCQP admits a direct convex SDP relaxation. Moreover, the results from [11] provide a sufficient condition for a local minimum to be a global minimum, which a solver can further use as a stopping criterion. Alternatively, the problem can be solved with the classic Rayleigh-Ritz procedure by finding the eigenvector associated with the smallest eigenvalue λ of the generalized eigenvalue problem $-W_\alpha\alpha' = \lambda E\alpha'$.

It can be shown that in the limit of small Δt, TQS reduces to QAS (see Appendix C). This could potentially give us a way to obtain systematic higher-order corrections to the QAS matrix differential equation. Interestingly, this is a conceptual unification of the ground state problem (QAE) with the dynamics problem (QAS) in the quantum assisted framework. In QAE, finding the ground state and ground state energy of a Hamiltonian was formulated to become a QCQP. In TQS, the problem of simulating the dynamics is also given as a QCQP. This is conceptually satisfying as the problem of finding the dynamics is expressed as $e^{-itH}|\psi\rangle$, which is mathematically similar to using imaginary time evolution to finding the ground state via $e^{-\tau H}|\psi\rangle$. The aforementioned connection is also one of the primary justifications for ansatz selection in [12]. We note that as alternative it is possible implement the unitary evolution operator $U(\Delta t)$ directly instead of the Taylor series expansion of Eq.7, however this would require the usage of Hadamard tests (see Appendix D).

We want to emphasize again that the quantum computer is only required to measure the overlap matrices $\mathcal{E}$ and $\mathcal{D}$ at the start of the algorithm. No quantum-classical feedback loop for optimization is required. The only optimization steps required are performed solely on the classical computer with knowledge of the overlap matrices. The algorithm is as follows:

1. Choose an efficiently implementable initial state $|\psi\rangle$, then choose some K>0 and form the K-moment states $|\chi_i\rangle$ to construct the ansatz. This step can be done on paper.

2. With knowledge of the Hamiltonian H, calculate the overlap matrices $\mathcal{E}$ and $\mathcal{D}$ on the quantum computer. The job of the quantum computer is now done.

3. Choose a small Δt with respect to the eigenvalues of H and evolve the state forward in time using a classical computer, by solving the optimization program 17 subject to the constraint 18. If results are not up to the desired fidelity, increase K and repeat the algorithm.

The timestep Δt could be increased by including higher order terms in the power series expansion of $U(\Delta t)$ in our calculations (Described in Appendix E).
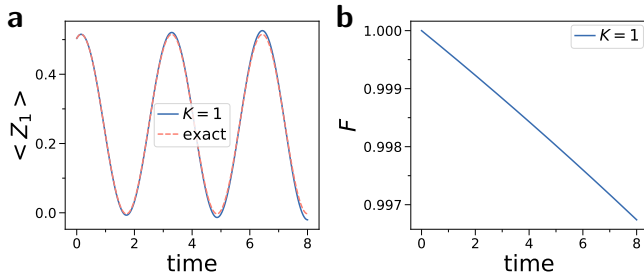
FIG. 1. Time evolution of TQS on a 2 qubit state, with Hamiltonian H_2, simulated on the IBM quantum processor *ibmq_rome*. **a)** Expectation value of $\langle Z_1 \rangle$ **b)** Fidelity of the state.

FIG. 2. Time evolution of TQS on a 4 qubit state with Hamiltonian H_4 simulated on the IBM quantum processor *ibmq_rome*. **a)** Expectation value of $\langle Z_1 \rangle$ **b)** Fidelity with exact solution.

Results— We first use TQS to simulate a 2 qubit Heisenberg model

$$H_2 = \frac{1}{2}X_1 X_2 + \frac{1}{2}Y_1 Y_2 + \frac{1}{2}Z_1 Z_2. \tag{19}$$

We apply it to evolve an initial randomized 2 qubit state $|\psi_2\rangle$. This initial state is generated by 5 layers of U_3 rotations and CNOT gates on the 2 qubits (see Appendix A). We ran the TQS algorithm on the 5-qubit quantum computer *ibmq_rome*, available through IBM Quantum Experience. We used error mitigation by calibrating the measurement errors and applying a filter obtained from that calibration on our data with the toolbox provided in Qiskit [37]. The results are shown in Fig.1. The evolution of the state under TQS reproduces the exact behavior very well for an ansatz with $K = 1$ moment states, even in the presence of the noise of a real quantum computer.

Next, we apply TQS to simulate a 4 qubit XX chain model on a quantum computer. Although this Hamiltonian is analytically solvable, we simulate this as a proof of principle.

$$H_4 = \frac{1}{2}X_1 X_2 + \frac{1}{2}X_2 X_3 + \frac{1}{2}X_3 X_4. \tag{20}$$

In Fig.2, we simulate this Hamiltonian on *ibmq_rome* with an initial randomized 4 qubit state, generated by 5 layers of U_3 rotations and CNOT gates (see Appendix A). We run it for the $K = 1$ to $K = 3$ moment states. The evolution of the state under TQS again reproduces the exact behavior very well for the $K = 3$ case.

Next, we investigate in Fig.3 the transverse Ising model with 8 qubits by simulating TQS on a classical computer.

$$H_8 = \sum_{i=0}^{7} \frac{1}{2}Z_i Z_{i+1} + \sum_{j=0}^{8} X_j. \tag{21}$$

With an initial random state, we find that the evolution of the state reproduces the exact dynamics for the case of $K = 3$ moment expansion.

Lastly, we compare TQS to pVQD for a 2 qubit transverse Ising model on a simulation. We consider the 2
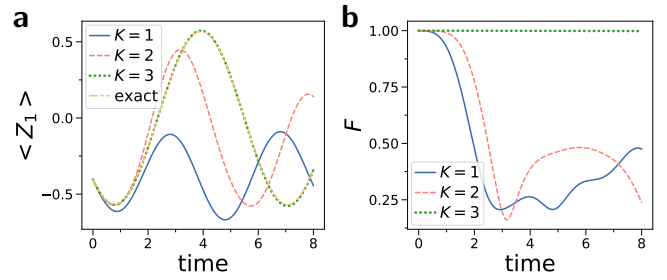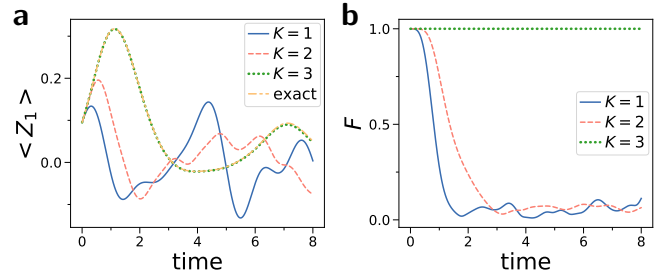

FIG. 3. Time evolution of TQS on a 8 qubit state, with Hamiltonian H_8, simulated on a classical computer, with a random initial state. **a)** Expectation value of $\langle Z_1 \rangle$. **b)** Fidelity of the state.

qubit transverse Ising Hamiltonian:

$$H_{TFI,2} = \sum_{i=0}^{2} \frac{1}{2}Z_i Z_{i+1} + \sum_{j=0}^{2} X_j. \tag{22}$$

We compared them with noisy simulators, with the noise models taken from the IBM Quantum Experience provider, which is meant to mimic the noise on their actual quantum computers. The results are shown in Fig.4. As can be seen, while both TQS and pVQD do have errors when trying to simulate this Hamiltonian in the presence of noise, the results for the expectation values of the state for TQS are closer to the classical results most of the time. This is especially so for the expectation value of $\langle Z_1 \rangle$. However, while the results might be argued to be somewhat similar, the resource needs of both algorithms on the quantum computer are quite different. The TQS algorithm in our case required ≈ 30 circuits to be run, while the pVQD simulator required well over 4000 circuits to be run, which is already a little prohibitory for us to run on the IBM Quantum Experience. It should be mentioned that if we wanted to increase the simulation time for this example, since the algebra has already closed, we could do that with no extra circuits with TQS, while the number of circuits in pVQD scales linearly with the number of steps required.

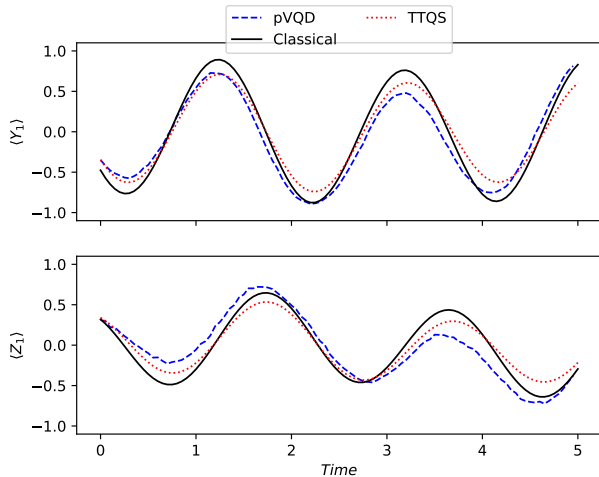*Discussion and Conclusion*— The currently proposed

FIG. 4. Time evolution of TQS and pVQD on a 2 qubit state, with Hamiltonian $H_{TFI,2}$, simulated with a noisy simulator. The noise model was taken from the IBM Quantum Experience provider, mimicking the noise of the real quantum processor *ibmq_bogota*. pVQD was run for 100 optimization steps, and made use of a parametric quantum circuit with 8 parameters, made out of sucessive layers of single qubit X rotations and 2-qubit ZZ rotations. The expectation values of $\langle Y_1 \rangle$ and $\langle Z_1 \rangle$ are plotted. The results are somewhat similar, although TQS does have results closer to the classical results most of the time, especially so for the expectation value of $\langle Z_1 \rangle$, as it is better able to capture the peaks and troughs of the expectation values. However, while the results might be argued to be somewhat similar, the resource needs of both algorithms on the quantum computer are quite different. The TQS algorithm in our case required ≈ 30 circuits to be run, while the pVQD simulator required well over 4000 circuits to be run.

NISQ algorithms face problems in scaling up to system sizes where classical computers cannot simulate the same systems, or in other words, to the point where we would see quantum advantage. For example, VQS/SVQS/pVQD require the use of a quantum-classical feedback loop, usually require complicated circuits, share similar problems as VQE like the barren plateau problem, and lack a systematic way to generate a parameterized ansatz. VFF/fsVFF also suffers from lacking a systematic way to generate the ansatz, usually requires complicated circuits and has to run a quantum-classical feedback loop it at the start. Further, the no fast-forwarding theorem suggests that not all Hamiltonians will be able to be accurately diagonalized with a reasonable amount of gates and circuit length, and the optimization step of the cost function in VFF might be too difficult to carry out efficiently. However, the barren plateau problem and ansatz state generation could be improved upon by applying various techniques [38–42].

One problem that VQS and QAS share is that they require solving a differential equation which includes the pseudo-inverse of a matrix, whose elements are mea-

sured on a quantum computer. This matrix can be ill-conditioned. This procedure, via singular value decomposition, can be numerically unstable and sensitive to noise, especially as the system increases in size [43]. However, the sensitivity of these matrices has not been rigorously analyzed and more work has to be done to understand the scaling of the sensitivity.

In this work, we develop TQS for simulating quantum dynamics on digital quantum computers. TQS recasts the dynamical problem as a QCQP optimization program, which is well characterized unlike the optimization program in pVQD, allowing us to avoid the aforementioned problem in VQS and QAS.

At the same time, TQS retains the advantages of QAS, namely providing us a systematic method to select the ansatz, avoiding complicated Hadamard tests and controlled unitaries, avoiding the barren plateau problem, and only requiring usage of the quantum computer at the start, all of which are problems that are present in pVQD.

However, there are still many problems to tackle in our approach. One problem is an inherited problem from QAS. As the Hamiltonian size and complexity increase, large K values may be needed to generate enough states for a sufficiently expressible ansatz to produce accurate results. Though QAS uses a problem aware ansatz, more information from the problem such as the combination coefficients β_i and symmetries of the Hamiltonian should be employed to further tame the complexity.

As the system size increases, it may be required to reduce Δt to preserve accuracy in the post-processing part of the algorithm. This will increase the computational cost of the classical computer. The number of optimization steps to be carried out increases linearly with the number of discretizations steps of the evolution time. Determining whether this poses a bottleneck for TQS when applied to large systems requires further studies.

Furthermore, in the presence of noise, the calculated fidelity of our states can go above one. A possible origin are small eigenvalues in the $\mathcal{E}$ overlap matrix, which can give the procedure of optimizing or solving the generalized eigenvalue problem numerical instability. As we scale up the system and consider more ansatz states, this issue can become more prevalent.

In the future, the NISQ community should investigate these challenges, so that we can successfully run NISQ algorithms for larger qubit numbers.

[1] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, *et al.*, Nature **574**, 505 (2019).

[2] H.-S. Zhong, H. Wang, Y.-H. Deng, M.-C. Chen, L.-C. Peng, Y.-H. Luo, J. Qin, D. Wu, X. Ding, Y. Hu, *et al.*, Science **370**, 1460 (2020).

[3] G. A. Quantum *et al.*, Science **369**, 1084 (2020).

[4] J. Preskill, Quantum **2**, 79 (2018).

[5] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, *et al.*, arXiv preprint arXiv:2101.08448 (2021).

[6] P. W. Shor, in *Proceedings 35th annual symposium on foundations of computer science* (Ieee, 1994) pp. 124–134.

[7] I. H. Deutsch, arXiv preprint arXiv:2010.10283 (2020).

[8] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, *et al.*, arXiv preprint arXiv:2012.09265 (2020).

[9] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. Obrien, Nature communications **5**, 4213 (2014).

[10] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, New Journal of Physics **18**, 023023 (2016).

[11] K. Bharti, arXiv preprint arXiv:2009.11001 (2020).

[12] K. Bharti and T. Haug, arXiv preprint arXiv:2010.05638 (2020).

[13] R. P. Feynman, Int. J. Theor. Phys **21** (1982).

[14] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Physical Review Letters **114**, 090502 (2015).

[15] S. Lloyd, Science , 1073 (1996).

[16] B. P. Lanyon, C. Hempel, D. Nigg, M. Müller, R. Gerritsma, F. Zähringer, P. Schindler, J. T. Barreiro, M. Rambach, G. Kirchmair, *et al.*, Science **334**, 57 (2011).

[17] X. Peng, J. Du, and D. Suter, Physical review A **71**, 012307 (2005).

[18] R. Barends, A. Shabani, L. Lamata, J. Kelly, A. Mezzacapo, U. Las Heras, R. Babbush, A. G. Fowler, B. Campbell, Y. Chen, *et al.*, Nature **534**, 222 (2016).

[19] R. Barends, L. Lamata, J. Kelly, L. García-Álvarez, A. Fowler, A. Megrant, E. Jeffrey, T. White, D. Sank, J. Mutus, *et al.*, Nature communications **6**, 1 (2015).

[20] E. A. Martinez, C. A. Muschik, P. Schindler, D. Nigg, A. Erhard, M. Heyl, P. Hauke, M. Dalmonte, T. Monz, P. Zoller, *et al.*, Nature **534**, 516 (2016).

[21] L. M. Sieberer, T. Olsacher, A. Elben, M. Heyl, P. Hauke, F. Haake, and P. Zoller, npj Quantum Information **5**, 1 (2019).

[22] D. Poulin, M. B. Hastings, D. Wecker, N. Wiebe, A. C. Doherty, and M. Troyer, arXiv:1406.4920 (2014).

[23] Y. Li and S. C. Benjamin, Physical Review X **7**, 021050 (2017).

[24] X. Yuan, S. Endo, Q. Zhao, Y. Li, and S. C. Benjamin, Quantum **3**, 191 (2019).

[25] M. Benedetti, M. Fiorentini, and M. Lubasch, arXiv preprint arXiv:2009.12361 (2020).

[26] K. Heya, K. M. Nakanishi, K. Mitarai, and K. Fujii, arXiv preprint arXiv:1904.08566 (2019).

[27] C. Cirstoiu, Z. Holmes, J. Iosue, L. Cincio, P. J. Coles, and A. Sornborger, npj Quantum Information **6**, 1 (2020).

[28] B. Commeau, M. Cerezo, Z. Holmes, L. Cincio, P. J. Coles, and A. Sornborger, arXiv preprint arXiv:2009.02559 (2020).

[29] J. Gibbs, K. Gili, Z. Holmes, B. Commeau, A. Arrasmith, L. Cincio, P. J. Coles, and A. Sornborger, arXiv preprint arXiv:2102.04313 (2021).

[30] K. Bharti and T. Haug, arXiv preprint arXiv:2011.06911 (2020).

[31] J. W. Z. Lau, K. Bharti, T. Haug, and L. C. Kwek, arXiv preprint arXiv:2101.07677 (2021).

[32] T. Haug and K. Bharti, arXiv preprint arXiv:2011.14737 (2020).

[33] M. Otten, C. L. Cortes, and S. K. Gray, arXiv preprint arXiv:1910.06284 (2019).

[34] S. Barison, F. Vicentini, and G. Carleo, arXiv:2101.04579 (2021).

[35] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, Nature communications **9**, 4812 (2018).

[36] L. Bittel and M. Kliesch, arXiv:2101.07267 (2021).

[37] H. Abraham *et al.*, "Qiskit: An open-source framework for quantum computing," (2019).

[38] T. Haug, K. Bharti, and M. Kim, arXiv preprint arXiv:2102.01659 (2021).

[39] K. Nakaji and N. Yamamoto, arXiv:2005.12537 (2020).

[40] T. Volkoff and P. J. Coles, Quantum Science and Technology **6**, 025008 (2021).

[41] Z. Holmes, K. Sharma, M. Cerezo, and P. J. Coles, arXiv preprint arXiv:2101.02138 (2021).

[42] H.-Y. Huang, K. Bharti, and P. Rebentrost, arXiv preprint arXiv:1909.07344 (2019).

[43] J. W. Demmel, Numerische Mathematik **51**, 251 (1987).

[44] A. McLachlan, Molecular Physics **8**, 39 (1964).

	$K = 1$	$K = 2$	$K = 3$	$K = 4$
2 Qubit Case	1	4		
4 Qubit Case	1	4	7	8
8 Qubit Case	1	17	137	

TABLE I. Comparison of the number of basis states used to construct the hybrid ansatz for each K for each Hamiltonian. For example, the $K = 2$ expansion for the 4 qubit case, using the Hamiltonian H_4, requires 4 quantum states to construct the hybrid ansatz.

Appendix A: Details on running circuits on the IBM quantum computer

For the runs on the real quantum computer, we generated an initial state with randomized parameters to evolve with the following circuit. It comprised 5 layers of successive U_3 rotation with randomized parameters on each qubit, followed by a CNOT/entangling gate. We (see Fig.5 and 6). We sampled from each circuit 8192 shots.
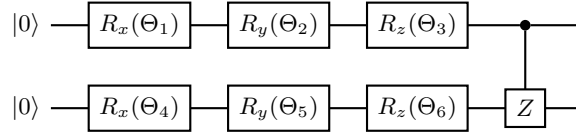

FIG. 5. Circuit for two qubits that generate one set of U_3 rotation with randomized parameters, followed by a CNOT gate between the 2 qubits. 5 successive layers of this circuit were used to generate the initial starting state for the 2 qubit case on the IBM quantum computer for our runs of TQS. The Θs were randomly generated.
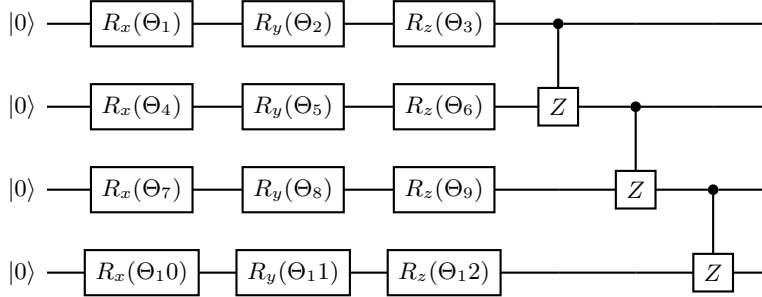

FIG. 6. Circuit for four qubits that generate one set of U_3 rotation with randomized parameters, followed by a series of CNOT gates between the adjacent qubits. 5 successive layers of this circuit were used to generate the initial starting state for the 4 qubit case on the IBM quantum computer for our runs of TQS. The Θs were randomly generated.

Appendix B: Number of basis states considered for each K

The number of basis states that was used to construct the hybrid ansatz, for each K moment expansion, for each Hamiltonian, is given in Table I.

Appendix C: QAS and VQS as special cases of TQS

In this appendix, we show that in the limit of choosing a very small Δt, one obtains QAS from TQS. Since VQS is a special case of QAS [30], we get VQS also as special case of TQS. We start out with the series expansion of $|\psi(\vec{\alpha} + \delta\vec{\alpha})\rangle$

$$|\psi(\vec{\alpha} + \delta\vec{\alpha})\rangle = |\psi(\vec{\alpha})\rangle + \sum_j \frac{\partial}{\partial\alpha_j} |\psi(\vec{\alpha})\rangle \, \delta\alpha_j. \tag{C1}$$

Now in TQS we want to maximize the overlap of $U(\Delta t)\,|\psi(\vec{\alpha})\rangle$ and $|\psi(\vec{\alpha}+\delta\vec{\alpha})\rangle$, which is essentially the fidelity measure in equation 8

$$|\langle\psi(\vec{\alpha})|\,U^{\dagger}(\Delta t)\,|\psi(\vec{\alpha}+\delta\vec{\alpha})\rangle|^2 = \left[\langle\psi(\vec{\alpha})|\,U^{\dagger}(\Delta t)\,|\psi(\vec{\alpha})\rangle + \sum_j |\psi(\vec{\alpha})\rangle\,U^{\dagger}(\Delta t)\frac{\partial\,|\psi(\vec{\alpha})\rangle}{\partial\alpha_j}\delta\alpha_j\right] \times [\text{C. C.}]$$

$$\stackrel{|\psi(\vec{\alpha})\rangle=\sum_j\alpha_j|\chi_j\rangle}{=} \left[\langle\psi(\vec{\alpha})|\,U^{\dagger}(\Delta t)\,|\psi(\vec{\alpha})\rangle + \sum_j |\psi(\vec{\alpha})\rangle\,U^{\dagger}(\Delta t)\,|\chi_j\rangle\,\delta\alpha_j\right] \times [\text{C. C.}]$$

$$= |\langle\psi(\vec{\alpha})|\,U^{\dagger}(\Delta t)\,|\psi(\vec{\alpha})\rangle|^2 + \sum_j \langle\psi(\vec{\alpha})|\,U^{\dagger}(\Delta t)\,|\chi_j\rangle\,\langle\psi(\vec{\alpha})|\,U(\Delta t)\,|\psi(\vec{\alpha})\rangle\,\delta\alpha_j$$

$$+ \sum_j \langle\chi_j|\,U(\Delta t)\,|\psi(\vec{\alpha})\rangle\,\langle\psi(\vec{\alpha})|\,U^{\dagger}(\Delta t)\,|\psi(\vec{\alpha})\rangle\,\delta\alpha_j^* + \sum_{j,k} \langle\psi(\vec{\alpha})|U^{\dagger}(\Delta t)|\chi_j\rangle\,\langle\chi_k|U(\Delta t)|\psi(\vec{\alpha})\rangle\,\delta\alpha_j\delta\alpha_k^*. \tag{C2}$$

Now in the same manner as QAS, using the Mclachlan's variational principle [24, 30, 31, 44], we demand that the variation of this fidelity is equal to 0 with respect to α_j:

$$\implies \langle\psi(\vec{\alpha})|\,U^{\dagger}(\Delta t)\,|\chi_j\rangle\,\langle\psi(\vec{\alpha})|\,U(\Delta t)\,|\psi(\vec{\alpha})\rangle + \sum_k \langle\psi(\vec{\alpha})|U^{\dagger}(\Delta t)|\chi_j\rangle\,\langle\chi_k|U(\Delta t)|\psi(\vec{\alpha})\rangle\,\delta\alpha_k^* = 0$$

$$\implies \langle\psi(\vec{\alpha})|\,U(\Delta t)\,|\psi(\vec{\alpha})\rangle + \sum_k \langle\chi_k|U(\Delta t)|\psi(\vec{\alpha})\rangle\,\delta\alpha_k^* = 0. \tag{C3}$$

Now we substitute in $U(\delta t) = I - i\Delta t H$:

$$\implies \langle\psi(\vec{\alpha})|\psi(\vec{\alpha})\rangle - i\Delta t\,\langle\psi(\vec{\alpha})|\,H\,|\psi(\vec{\alpha})\rangle + \sum_k \langle\chi_k|\psi(\vec{\alpha})\rangle\,\delta\alpha_k^* - i\Delta t \sum_k \langle\chi_k|H|\psi(\vec{\alpha})\rangle\,\delta\alpha_k^* = 0. \tag{C4}$$

Now we take the derivative of this equation with respect to Δt. Note that $\frac{d}{d\Delta t}\delta\alpha_k^* = \delta\dot{\alpha}_k^*$. We then discard any terms remaining that are linear in Δt or in $\delta\alpha$ (implying we have chosen such a small Δt that $\delta\alpha$ is also very small).

$$\implies -i\,\langle\psi(\vec{\alpha})|\,H\,|\psi(\vec{\alpha})\rangle + \sum_k \delta\dot{\alpha}_k^*\,\langle\chi_k|\psi(\vec{\alpha})\rangle\,\delta\alpha_k^* = 0. \tag{C5}$$

Using the above definition of the $\mathcal{E}$ and $\mathcal{D}$ matrices in equation 12 and 13, this simplifies to:

$$\implies -i\vec{\alpha}^{\dagger}\mathcal{D}\vec{\alpha} + \dot{\vec{\alpha}}^{\dagger}\mathcal{E}\vec{\alpha} = 0$$
$$\implies \mathcal{E}\dot{\vec{\alpha}} = -i\mathcal{D}\vec{\alpha}. \tag{C6}$$

This is exactly the same differential equation that we aim to solve in QAS. If we do not ignore the higher order terms, we could obtain systematic higher order corrections to the QAS matrix differential equation using such a method.

Appendix D: Unitary implementation

As alternative, we could implement the unitary evolution operator $U(\Delta t)$ directly instead of the Taylor series expansion of Eq.7

$$|\psi(\alpha(t+\Delta t))\rangle_K = U(\Delta t)\,|\psi(\alpha(t))\rangle_K. \tag{D1}$$

and defining the matrix $\mathcal{R}_{m,n} = \langle\chi_m|U(\Delta t)|\chi_n\rangle$ to solve the program

$$\max_{\alpha'} \alpha'^{\dagger}\mathcal{R}\alpha\alpha^{\dagger}\mathcal{R}^{\dagger}\alpha' \tag{D2}$$

$$\text{s.t } \alpha'^{\dagger}\mathcal{E}\alpha' = 1. \tag{D3}$$

$U(\Delta t)$ could be implemented with a Trotter decomposition or with an oracle. However, this complicates the circuits needed to calculate the $\mathcal{R}$ matrix, requiring the usage of Hadamard tests.

Appendix E: Higher order approximations

We investigate higher order expansion for the evolution operator in this section. First, we define the overlap matrix $\mathcal{J}$

$$\mathcal{J}_{m,n} = \sum_{i,j} \beta_i \beta_j \langle \chi_m | P_i P_j | \chi_n \rangle. \tag{E1}$$

Considering the next highest power expansion of $U(\Delta t)$:

$$U(\Delta t) \approx I - \iota \Delta t \left(\sum_{j=1}^{r} \beta_j P_j \right) - \frac{\Delta t^2}{2} \left(\sum_{j=1}^{r} \beta_j P_j \right)^2 \equiv V_2(\Delta t), \tag{E2}$$

and defining $|\phi\rangle = V_2(\Delta t) |\psi(\alpha)\rangle_K$, the constraint in the optimization program 11 turns out to be still the same as equation 14:

$$\langle \psi(\alpha'^{\dagger}) | \psi(\alpha'^{\dagger}) \rangle = \alpha'^{\dagger} \mathcal{E} \alpha'. \tag{E3}$$

It turns out that $\langle \phi | \phi \rangle$ is actually exactly equal to $\alpha^{\dagger} E \alpha$, which is the result we used earlier in equation 15, as all the 2nd order terms nicely cancel out.

Now, using the notation $G_2 \equiv \left(\mathcal{E} - \iota \Delta t \mathcal{D} - \frac{\Delta t^2}{2} \mathcal{J} \right)$,

$$\langle \psi(\alpha') | \phi \rangle_K \langle \phi | \psi(\alpha') \rangle_K = \alpha'^{\dagger} G_2 \alpha \alpha^{\dagger} G_2^{\dagger} \alpha'. \tag{E4}$$

Now the optimization program in 11 can be re-expression in this higher order approximation as

$$\max_{\alpha'} \ \alpha'^{\dagger} \left(\frac{G_2 \alpha \alpha^{\dagger} G_2^{\dagger}}{\alpha^{\dagger} \mathcal{E} \alpha} \right) \alpha' \tag{E5}$$

$$\text{s.t } \alpha'^{\dagger} \mathcal{E} \alpha' = 1.$$

And using the notation $W_{2,\alpha} \equiv \frac{G_2 \alpha \alpha^{\dagger} G_2^{\dagger}}{\alpha' \mathcal{E} \alpha}$, we further condense the above optimization program as

$$\max_{\alpha'} \ \alpha'^{\dagger} W_{2,\alpha} \alpha' \tag{E6}$$

$$\text{s.t } \alpha'^{\dagger} \mathcal{E} \alpha' = 1. \tag{E7}$$

Once again, the only work that the quantum computer need to do is to calculate overlap matrices in the start, in this case having to calculate $\mathcal{E}$, $\mathcal{D}$ and $\mathcal{J}$. In fact, when going from lower order approximations to higher order approximations, you can reuse the saved matrices and only calculate the new ones needed. In this case, in the original TQS, which uses a first order approximation for $U(\Delta t)$, we already have the $\mathcal{E}$ and $\mathcal{D}$ matrices, so if we deem the results not up to our desired accuracy, we can easily go to the second order approximation showed here, and only require calculation of one additional matrix $\mathcal{J}$.