

PAPER • OPEN ACCESS

A universal quantum circuit design for periodical functions

To cite this article: Junxu Li and Sabre Kais 2021 *New J. Phys.* **23** 103022

View the [article online](#) for updates and enhancements.

You may also like

- [Quantum algorithm and circuit design solving the Poisson equation](#)
Yudong Cao, Anargyros Papageorgiou, Iasonas Petras et al.
- [Quantum circuit design for accurate simulation of qudit channels](#)
Dong-Sheng Wang and Barry C Sanders
- [Neural predictor based quantum architecture search](#)
Shi-Xin Zhang, Chang-Yu Hsieh, Shengyu Zhang et al.



PAPER

A universal quantum circuit design for periodical functions

OPEN ACCESS

RECEIVED
18 June 2021REVISED
10 September 2021ACCEPTED FOR PUBLICATION
4 October 2021PUBLISHED
19 October 2021

Original content from
this work may be used
under the terms of the
[Creative Commons
Attribution 4.0 licence](#).

Any further distribution
of this work must
maintain attribution to
the author(s) and the
title of the work, journal
citation and DOI.

Junxu Li  and Sabre Kais* 

Department of Chemistry, Department of Physics and Astronomy, and Purdue Quantum Science and Engineering Institute, Purdue University, West Lafayette, IN 47907, United States of America

* Author to whom any correspondence should be addressed.

E-mail: kais@purdue.edu**Keywords:** quantum algorithm, quantum circuit, quantum simulationSupplementary material for this article is available [online](#)

Abstract

We propose a universal quantum circuit design that can estimate any arbitrary one-dimensional periodic functions based on the corresponding Fourier expansion. The quantum circuit contains N -qubits to store the information on the different N -Fourier components and $M + 2$ auxiliary qubits with $M = \lceil \log_2 N \rceil$ for control operations. The desired output will be measured in the last qubit q_N with a time complexity of the computation of $O(N^2 \lceil \log_2 N \rceil^2)$, which leads to polynomial speedup under certain circumstances. We illustrate the approach by constructing the quantum circuit for the square wave function with accurate results obtained by direct simulations using the IBM-QASM simulator. The approach is general and can be applied to any arbitrary periodic function.

Introduction

The field of quantum information and quantum computing advances in both software and hardware in the past few years. The achievement of 72-qubit quantum chip, Sycamore, with programmable superconducting processor [1] heralded a remarkable triumph toward quantum supremacy experiment [2]. On the other hand, the photonic quantum computer, Jiuzhang [3], demonstrated quantum computational advantages with Boson sampling using photons. The blooming of hardware development by IBM, Google, IonQ and many others provokes tremendous enthusiasm developing quantum algorithms utilizing near term quantum devices and pursuit of application in various fields of science and engineering. Recently there arises a growing body of research focusing on quantum optimization [4, 5], solving linear system of equations [6–8], electronic structure calculations [9–15], quantum encryption [16, 17], variational quantum eigensolver [18, 19] for various problems [20–23] and open quantum dynamics [24–28]. Recently, quantum machine learning further explored and implemented quantum software that could show advantages compared with the corresponding classical ones [29–36].

However, difficulties arise inevitably when attempting to include nonlinear functions into quantum circuits. For example, the very existence of nonpolynomial activation functions guarantees that multilayer feedforward networks can approximate any functions [37]. Even though, the nonlinear activation functions do not immediately correspond to the mathematical framework of quantum theory, which describes system evolution with linear operations and probabilistic observation. Conventionally, it is found extremely difficult generating these nonlinearities with a simple quantum circuit. The alternative approach is to make a compromise, such as applying simple cosine functions as activation [38], or imitating the nonlinear functions with repeated measurements [39–41], or with assistance of the quantum Fourier transform [42] (QFT [43, 44]). How to simulate an arbitrary function, especially nonlinear functions from a quantum circuit is an important issue to be addressed.

In this paper, we proposed a universal design of quantum circuit, which is able to generate arbitrary finite continuous periodic 1D functions, even nonlinear ones such as the square wave function, with the given Fourier expansion. The output information is all stored in the last qubit, which could be measured for

the function estimation, or used as an intermediate state for following computation, as the case of estimating the nonlinear activation function between layers in quantum machine learning methods. We presented the details of the quantum circuit design in the first section followed by numerical simulation of the circuit imitating the square wave function on IBM-QASM. The final section contains complexity analysis and further applications.

1. Design of the quantum circuit

Consider a periodic 1D function $F_N(x)$ that can be expanded as Fourier series with N nontrivial components,

$$F_N(x) = \sum_{n=1}^N a_n \cos\left(\frac{2\pi}{T}nx + b_n\right), \quad (1)$$

where T is the period, and for simplicity we set $T = \pi$. To construct the quantum circuit that estimates the output function $F_N(x)$, we need N -qubits to store the input information, all of which are initially prepared at the state $|\psi(x)\rangle = \cos x|0\rangle + \sin x|1\rangle$. Additionally, there are $M + 2$ auxiliary qubits, with $M = \lceil \log_2 N \rceil$ qubits assigned q'_1, \dots, q'_M and the other two are q''_1, q''_2 . All the auxiliary qubits are initially set as $|0\rangle$ states. Thus, the input state can be written as

$$|\Psi_{\text{in}}(x)\rangle = |0\rangle_{q'}^{\otimes M} \otimes |0\rangle_{q''}^{\otimes 2} \otimes |\psi(x)\rangle_q^{\otimes N}, \quad (2)$$

where subscripts indicate the group of the three different registers. Figure 1(a) illustrates the structure of the quantum circuit design for the output function $F_N(x)$, while the detailed evolution is demonstrated in figure 1(b). There are two main modules in the quantum circuit: the first one contains U_{pre} acting on the auxiliary qubits q' , converting them from $|0\rangle$ to state $|\psi'_f\rangle$, and Hadamard gates acting on q'' , converting them to states $|+\rangle = 1/\sqrt{2}[|0\rangle + |1\rangle]$. The intermediate state $|\psi'_f\rangle$ can be described as

$$|\psi'_f(\gamma)\rangle = \sum_{n=0}^{2^M-1} \sqrt{\gamma_n} |n\rangle, \quad (3)$$

where $\sum_{n=0}^{2^M-1} \gamma_n = 1$ and $\gamma_n \geq 0$. Details about the design of U_{pre} and γ_n can be found in the supplementary materials (<https://stacks.iop.org/NJP/23/103022/mmedia>). The succeeding module is formed by N controlled unitary operations, where q' are the control qubits for the target qubits q'' and q . Denote the unitary operations as U_n , with a general structure shown in figure 1(c). Initially, the Hadamard gates are applied on q'' , and a rotation Y gate is applied on the first qubit q_1 . All these three qubits then acts as control qubits, while q_2 is the target in following operation. Next, qubit q_2, q_3, \dots, q_n are connected as a chain with simple control rotation Y gates. Finally a swap gate between q_n and q_N is included, ensuring that all the necessary information are stored in the last qubit q_N .

For simplicity, in the operation U_n , we define $w_{0,1}^k$ and $v_{0,1}^k$ as

$$\begin{aligned} R_y(\theta_k)^\dagger |0\rangle &= \cos(w_0^k)|0\rangle + \sin(w_0^k)|1\rangle \\ R_y(\theta_k)^\dagger |1\rangle &= \cos(w_1^k)|0\rangle + \sin(w_1^k)|1\rangle \\ R_y(\theta'_k)^\dagger |0\rangle &= \cos(v_0^k)|0\rangle + \sin(v_0^k)|1\rangle \\ R_y(\theta'_k)^\dagger |1\rangle &= \cos(v_1^k)|0\rangle + \sin(v_1^k)|1\rangle \end{aligned} \quad (4)$$

and α, β when $k \geq 2$ as

$$\alpha = \frac{1}{2} \prod_{k=2}^n |\sin(v_1^k - w_1^k)| \quad (5)$$

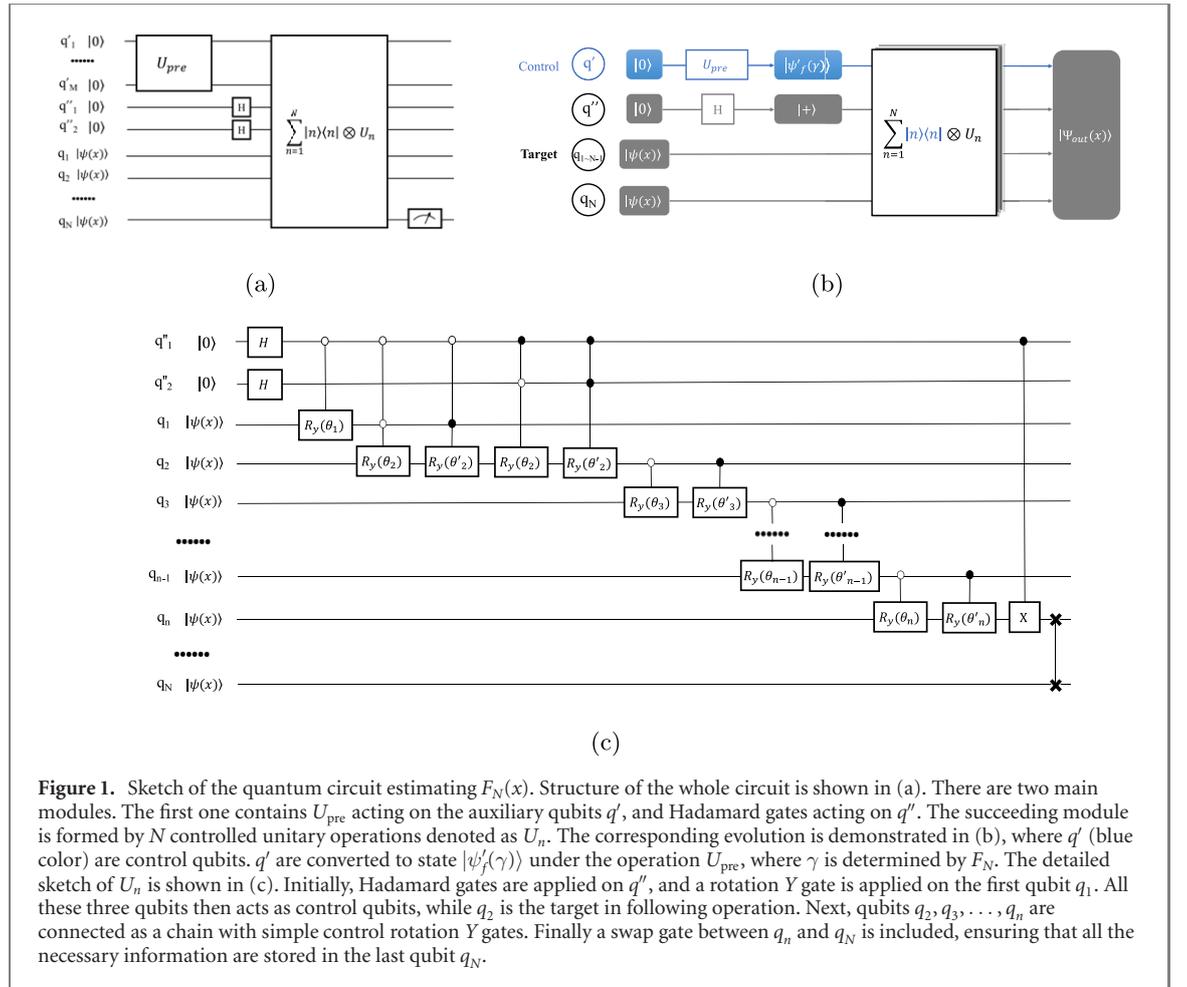
$$\beta_k = \arctan 2(\sin 2v_1^{k+1} - \sin 2w_1^{k+1}, \cos 2v_1^{k+1} - \cos 2w_1^{k+1}) \quad (6)$$

while we have $\alpha = 1/2$ and $\beta = 2w_1^1$ when $k = 1$.

To estimate $F_N(x)$, we need to ensure that

$$C \sum_{n=1}^N a_n \cos(2nx + b_n) + \frac{1}{2} = \sum_{m=1}^N \gamma_m \alpha^m \prod_{k=1}^m \cos(2x - \beta_k^m) + \frac{1}{2}, \quad (7)$$

where C is a nonzero constant ensuring that $|CF_N(x)| \leq \frac{1}{2}$, and the superscript m of α^m, β_k^m indicating that they belong to the operation U_m . The right-hand side of equation (7) is the probability to get the outcome



result $|1\rangle$ when measuring q_N itself after running the whole quantum circuit (more details can be found in the supplementary materials).

Thus, subsequent to the whole operation, the output state will be

$$\begin{aligned}
 |\Psi_{out}(x)\rangle &= \sqrt{-CF_N(x) + \frac{1}{2}} |\phi_0\rangle_{q',q'',q_1,\dots,q_{N-1}} \otimes |0\rangle_{q_N} \\
 &+ \sqrt{CF_N(x) + \frac{1}{2}} |\phi_1\rangle_{q',q'',q_1,\dots,q_{N-1}} \otimes |1\rangle_{q_N},
 \end{aligned} \tag{8}$$

where $|\phi_{0,1}\rangle$ describing output state of all qubits and auxiliary qubits except the last one q_N . Hence, information stored in q_N is essential and sufficient. After measuring q_N , the probability of getting $|1\rangle$ is an estimation of $F_N(x)$. Whereas, it is also appropriate to apply succeeding operations on q_N , regarding it as an intermediate state for further computation. The same quantum circuit structure works when the input is in a superposition states, namely $|\Psi_{in}^s(\mathbf{x})\rangle$ describing a vector \mathbf{x} that contains $L + 1$ components,

$$|\Psi_{in}^s(\mathbf{x})\rangle = \sum_{l=0}^L c_l |\Psi_{in}(x_l)\rangle_{q',q'',q} \otimes |\Phi_l\rangle_Q, \tag{9}$$

where q', q'', q are entangled with some other qubits, namely in Q . The superscript s represents superposition, subscripts q', q'', q and Q indicate the group of different qubits in different registers, and $\sum_{l=0}^L |c_l|^2 = 1$. $|\Phi_l\rangle$ is a complete orthogonal set of the subspace expanded by Q , ensuring that $\langle \Phi_l | \Phi_l \rangle = \delta_{ll}$. After the whole operation, the output state is given by

$$|\Psi_{out}^s(\mathbf{x})\rangle = \sum_{l=0}^L c_l |\Psi_{out}(x_l)\rangle_{q',q'',q} \otimes |\Phi_l\rangle_Q.$$

If we only focus on the subspace expanded by q_N and Q ,

$$|\Psi_{\text{out}}^s(\mathbf{x})\rangle_{q_N, Q} = \sum_{l=1}^L c_l \left[\sqrt{-CF_N(x_l) + \frac{1}{2}} |0\rangle_{q_N} + \sqrt{CF_N(x_l) + \frac{1}{2}} |1\rangle_{q_N} \right] \otimes |\Phi_l\rangle_Q. \quad (10)$$

Then if q_N is measured, probability to get result $|1\rangle$ will be $\frac{1}{2} + C \sum_{l=0}^L |c_l|^2 F_N(x_l) \frac{1}{2}$. This property leads to potential applications in quantum algorithm development, for example, the design of nonlinear activation between layers in quantum machine learning.

2. Implementation: simulation for the square wave function

In this section we will demonstrate the quantum circuit design with a trivial example: simulation of the square wave function. Consider the following Fourier expansion for the square wave function $f(x) = \text{sign}(\sin 2x)$, the sum of the first seven terms is given by,

$$F(x) = \sum_{n=1 \in \text{odd}}^7 \frac{1}{n} \cos \left[n \cdot 2x - \frac{\pi}{2} \right], \quad (11)$$

where $F(x)$ is an odd function with $a_{2,4,6} = 0$. $q_{1,\dots,7}$ are required carrying out the input information, all of which will be initially prepared in the state $|\psi(x)\rangle$. Additionally, we need five auxiliary qubits, denoting them as $q'_{1,2}$, $q''_{1,2,3}$, and a'_n , b'_n satisfy

$$\begin{aligned} C \sum_{n=1}^j a'_n \cos(2nx + b'_n) &= C \sum_{n=1}^N a_n \cos(2nx + b_n) \\ &- \sum_{m=j+1}^N \gamma_m \alpha^m \prod_{k=1}^m \cos(2x - \beta_k^m) \quad 1 \leq j < N. \end{aligned} \quad (12)$$

Then for $n > 1$ we set

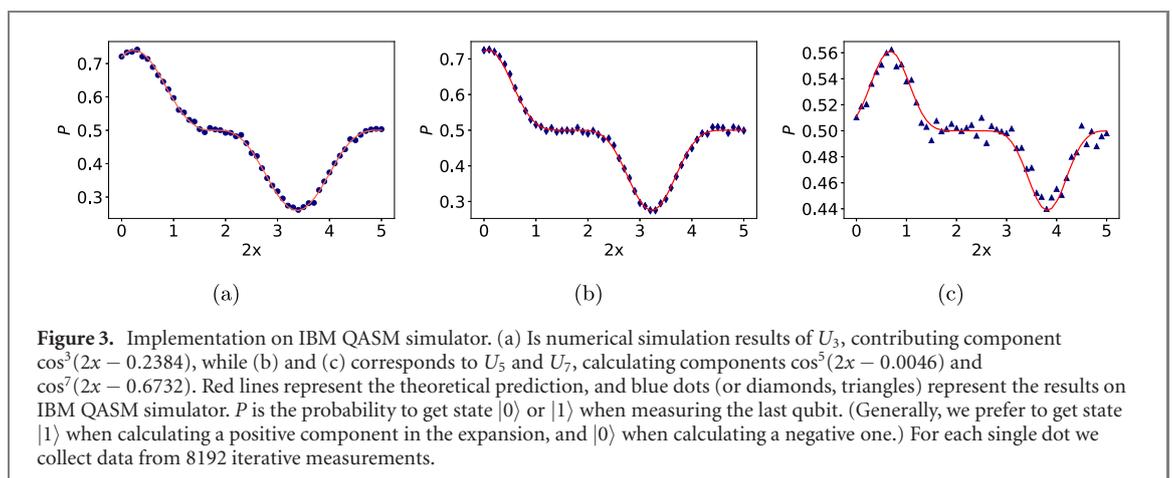
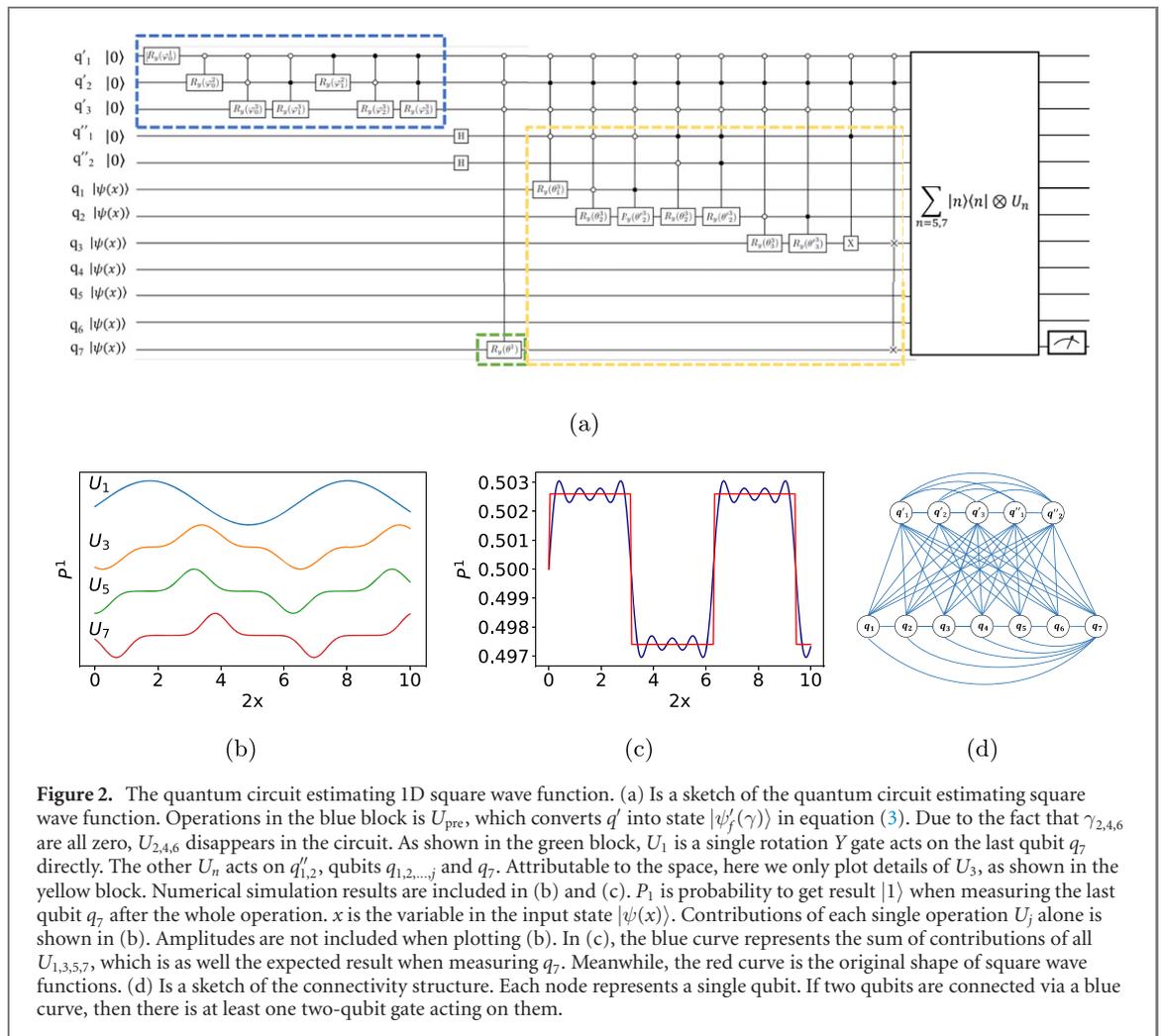
$$\begin{cases} \gamma_n = \frac{2^{n-1} C a'_n}{\alpha^n} \\ \beta_k^n = -\frac{b'_n}{n}, & b'_n \neq 0 \\ \beta_k^n = \frac{\pi}{3} \delta_{1,0} - \frac{\pi}{3(n-1)}, & b'_n = 0 \end{cases} \quad (13)$$

Equation (13) ensures that $\gamma_{2,4,6}$ are all zero, so that we do not need to construct $U_{2,4,6}$ in the circuit. We need to stress the fact that the above setting is not optimal, especially when one prefer a greater value of $|C|$ instead of a shallow circuit. Figure 2(a) is a scheme of the whole operation. Operations in the blue block is U_{pre} , which converts q' into state $|\psi'_f(\gamma)\rangle$ in equation (3). Due to the fact that $\gamma_{2,4,6}$ are all zero, then there is no need for construct $U_{2,4,6}$ in the quantum circuit. As shown in the green block, U_1 is a single rotation Y gate acts on the last qubit q_7 directly. The other U_n acts on $q''_{1,2}$, $q_{1,2,\dots,j}$ and q_7 . For illustration, we only plot details of U_3 , as shown in the yellow block. Based on equations (7) and (11) can be rewritten as

$$\begin{aligned} F(x) &= -0.8211 \cdot \cos(2x - 4.8812) + 18.9339 \cdot \cos^3(2x - 0.2384) \\ &- 15.6030 \cdot \cos^5(2x - 0.0046) - 9.1429 \cdot \cos^7(2x - 0.6732) \end{aligned}$$

Contributions of each single operation U_j alone is shown in figure 2(b). The sum of all their contributions are shown in figure 2(c), which is the output result. P_1 is probability to get the outcome result $|1\rangle$ when measuring the last qubit q_7 after the whole operation. Figure 2(d) is a sketch of the connectivity structure. Each node represents a single qubit. If two qubits are connected via a blue curve, then there is at least one two-qubit gate acting on them. All auxiliary qubits q' and q'' are connected with each other. Meanwhile, all qubits q are connected to all of the other auxiliary qubits. For any $1 \leq n < N$, there is also a connection between q_n and its neighbor q_{n+1} . The last qubit q_N is connected to all other qubits.

Further, we also implemented $U_{3,5,7}$ independently based on IBM QASM simulator, as shown in figure 3, where for each single dot we collect data from 8192 iterative measurements. Results from IBM QASM simulator fit well with the theoretical prediction corresponding to U_3 and U_5 , as shown in figures 3(b) and (c). More details of the simulations can be found in the supplementary materials.



3. Time complexity

We will compare the time complexity for three various situations: in the first situation we consider classical inputs, while for others we consider some unknown quantum states as inputs. One can either estimate x from the input and calculate $F_N(x)$ classically, or apply the method presented in this article to estimate the outputs.

Situation I. Estimate $F_N(x)$ with a classical input x :

To estimate a periodic function $F_N(x)$ within error ϵ , $O(\frac{1}{\epsilon})$ times of measurement are required [45]. Initially NR_y rotation gates are required for mapping x into quantum state $|\psi(x)\rangle^{\otimes N}$. Moreover, we need

$M + 2$ auxiliary qubits, where $M = \lceil \log_2 N \rceil$ for control qubits. U_{pre} contains $O(\exp(M))$ multi control gates, and in each of them there are at most $M - 1$ control qubits. The time scaling of n -control gates is $O(n^2)$ [46], thus the time complexity of U_{pre} is $O(N \lceil \log_2 N \rceil^2)$.

Consider the basic unit U_n . When $n > 2$, there are four control–control rotation gates, $4n - 8$ control-rotation gates and one optional two-qubit swap gate. Thus, it takes time $O(n \lceil \log_2 n \rceil^2)$ to implement the operation $|n\rangle\langle n| \otimes U_n$. After taking the M auxiliary qubits into account, M control qubits are added to all gates in U_n . Therefore, it takes time $O(n)$ to achieve a single U_n . Notice that there are N similar U_n , where $n = 1, 2, \dots, N$, time complexity to finish all $\sum_{n=1}^N |n\rangle\langle n| \otimes U_n$ is $O(N^2 \lceil \log_2 N \rceil^2)$.

Totally, to derive the estimation $CF_N(x)$ within error ϵ , the time complexity is of order $O(N^2 \lceil \log_2 N \rceil^2 / \epsilon)$, which is still polynomial. On the other hand, time complexity to estimate $F_N(x)$ for a single x based on Taylor expansion is also polynomial to N . Hence under this situation there is no speedup comparing with the classical calculation.

Situation II. Estimate $F_N(x)$ with input quantum state $|\psi(x)\rangle^{\otimes N}$, where the value of x is still unknown:

The only difference from the previous situation is that the mapping process now can be skipped. Still, time consuming is of order $O(N^2 \lceil \log_2 N \rceil^2 / \epsilon)$. By contrast, to calculate $F_N(x)$ from $|\psi(x)\rangle^{\otimes N}$ classically, the first step is to derive x from the input states. It requires $O(\frac{1}{\epsilon})$ measurements to get $\cos x$ within error ϵ , and then the time complexity to estimate $F_N(x)$ is polynomial to N . Still, under this situation there is no speedup comparing with the classical calculation.

Situation III. Estimate $\sum_{l=0}^L |c_l|^2 F_N(x_l)$ with input quantum state $|\Psi_{\text{in}}^s(\mathbf{x})\rangle$, as described in equation (9):

Here we denote N' as the number of qubits that form state $|\Phi\rangle$, and $L = 2^{N'} - 1$. Even though more variables are introduced into the input, we do not need to change anything in the quantum circuit. To get an estimation of $C \sum_{l=0}^L |c_l|^2 F_N(x_l)$ with the quantum method, time consuming is still of order $O(N^2 \lceil \log_2 N \rceil^2 / \epsilon)$, which does not depend on the scale of N' (number of qubits in Q). However, in the classical method, as c_j are unknown initially, one must estimate them before calculating $\sum_{l=0}^L |c_l|^2 F_N(x_l)$. Time complexity of quantum tomography is exponential in N' , or at least polynomial in N' with shadow quantum tomography. The time complexity of quantum method is only determined by N , while the classical method is at least polynomial in N' [47]. Thus, when $N' \gg N$, the quantum method will lead to polynomial speedup comparing with the classical one. Consider the task to estimate $|c_0|^2 F(x_0) + |c_1|^2 F(x_1)$, where $|c_0|^2 + |c_1|^2 = 1$. For simplicity, here we set $F(x) = \cos^3(x - 0.2384)$, which can be implemented by U_3 itself, excluding the q' registers. (Structure of U_3 can be found in figure 2 in the main article and figure S2 in the SM.) In **situation III**, the input states $|\Psi_{\text{in}}^s(\mathbf{x})\rangle$ can be described by equation (9).

In the task estimating $|c_0|^2 F(x_0) + |c_1|^2 F(x_1)$, we only need one qubit as Q preparing the initial states. A sketch of the quantum circuit calculating $|c_0|^2 F(x_0) + |c_1|^2 F(x_1)$ is shown in figure 4. There are totally six qubits, $q''_{1,2}$ and $q_{1,2,3}$ implementing U_3 , and qubit Q corresponding to equation (9). Initially, all qubits are set as $|0\rangle$. Operations in the dashed blue square converts them into state

$$|\Psi_{\text{in}}^s(x_0, x_1)\rangle = c_0 |\Psi_{\text{in}}(x_0)\rangle_{q',q'',q} \otimes |0\rangle_Q + c_1 |\Psi_{\text{in}}(x_1)\rangle_{q',q'',q} \otimes |1\rangle_Q, \tag{14}$$

where $|\Psi_{\text{in}}(x)\rangle_{q',q'',q} = |0\rangle_{q'}^{\otimes 2} \otimes |\psi(x)\rangle_{q''}^{\otimes 3}$, and $|\psi(x)\rangle = \cos x |0\rangle + \sin x |1\rangle$. Coefficients $c_{0,1}$ are determined by the gate $R_y(\theta_{\text{sup}})$ acting on Q , leading to

$$c_0 = \cos\left(\frac{\theta_{\text{sup}}}{2}\right)$$

$$c_1 = \sin\left(\frac{\theta_{\text{sup}}}{2}\right).$$

By tuning the parameter θ_{sup} , we can study the performance of the quantum circuit that calculates $|c_0|^2 F(x_0) + |c_1|^2 F(x_1)$ with various c_0, c_1 .

Then, operation U_3 is applied on q' and q , implementing the calculation of $F(x)$. After the whole operations, the output state is given by

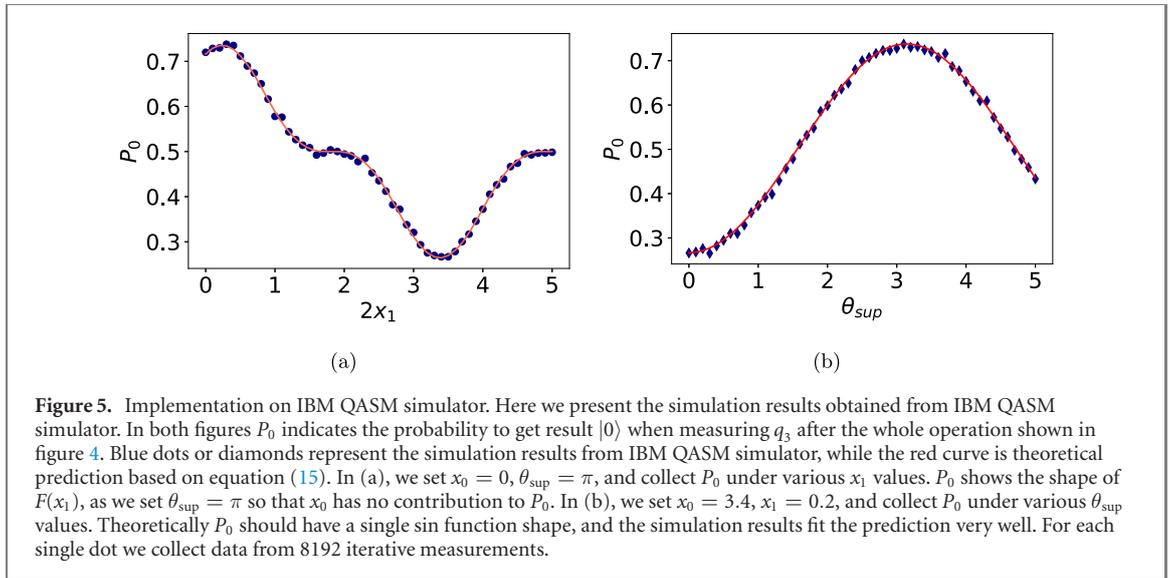
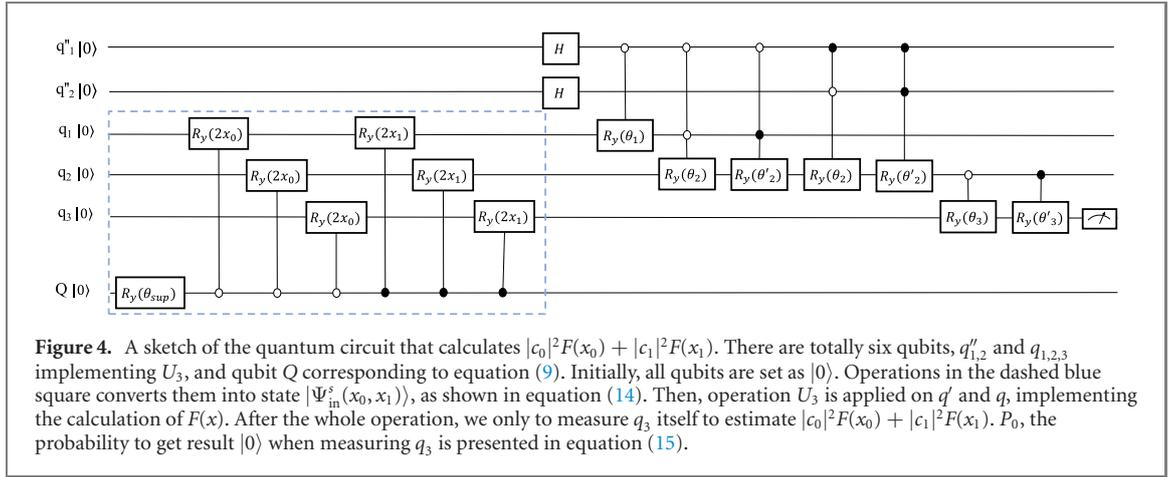
$$|\Psi_{\text{out}}^s(x_0, x_1)\rangle = c_0 |\Psi_{\text{out}}(x_0)\rangle_{q',q'',q} \otimes |0\rangle_Q + c_1 |\Psi_{\text{out}}(x_1)\rangle_{q',q'',q} \otimes |1\rangle_Q,$$

where $|\Psi_{\text{out}}(x)\rangle$ is described by equation (8).

In this example, P_0 , which represents the probability to get result $|0\rangle$ when measuring q_3 , can be calculated as

$$P_0 = \cos^2\left(\frac{\theta_{\text{sup}}}{2}\right) [0.2362 \cos^3(2x_0 - 0.2384) + 0.5]$$

$$+ \sin^2\left(\frac{\theta_{\text{sup}}}{2}\right) [0.2362 \cos^3(2x_1 - 0.2384) + 0.5]. \tag{15}$$



Thus, we only need to measure q_3 itself to estimate $|c_0|^2 F(x_0) + |c_1|^2 F(x_1)$.

Further, we implement the operation shown in figure 4 on IBM QASM simulator. Simulation results obtained from IBM QASM simulator are shown in figure 5(b), where P_0 indicates the probability to get result $|0\rangle$ when measuring q_3 after the whole operation. Blue dots or diamonds represent the simulation results from IBM QASM simulator, while the red curve is theoretical prediction based on equation (15). In figure 5(a), we set $x_0 = 0$, $\theta_{sup} = \pi$, and collect P_0 under various x_1 values. P_0 shows the shape of $F(x_1)$, as we set $\theta_{sup} = \pi$ so that x_0 has no contribution to P_0 . In figure 5(b), we set $x_0 = 3.4$, $x_1 = 0.2$, and collect P_0 under various θ_{sup} values. Theoretically P_0 should have a single sin function shape, and the simulation results fit the prediction very well. For each single dot we collect data from 8192 iterative measurements.

In the simulation only one qubit, q_3 , is measured, and $|c_0|^2 F(x_0) + |c_1|^2 F(x_1)$ can be estimated from P_0 , the probability to get result $|0\rangle$. Thus, with the quantum algorithm as we proposed, we do not need to estimate the exact values of c_l . In other words, the time consuming of quantum algorithm does not rely on N' . On the contrary, evaluating the coefficients c_l requires polynomial time of N' with the shadow quantum tomography [47]. When $N' \gg N$, the total time consuming mainly rely on the terms determined by N' . Therefore, in **situation III** the quantum method will lead to polynomial speedup comparing with the classical one.

4. Conclusion

Here, we propose a universal quantum circuit design for any arbitrary one-dimensional periodic function. The inputs are sufficient qubits prepared at the same state, while the last one will represent the output outcome. One can either estimate the exact value from repeating measurements on the last qubit, or regard it as an intermediate state prepared for succeeding operations. Superposition in the input leads to similarly

superposition in the output, which leads to polynomial speedup under some certain circumstances when dealing with unknown quantum inputs. As a simple example we illustrate the quantum circuit design for the square wave function. Both exact simulations and implementation on IBM-QASM gave very accurate result and illustrate the power of this proposed general design. This general approach might be used to construct an appropriate quantum circuit for the electronic wave function in periodic solids and materials, moreover in quantum machine learning particularly simulating the non-linear function used in the network.

Acknowledgments

The authors would like to thank Zinxuan Hu and Blake Wilson for the helpful suggestions and discussions. We acknowledge the financial support in part by the National Science Foundation under Award No. 1955907 and funding by the US Department of Energy (Office of Basic Energy Sciences) under Award No. de-sc0019215.

Data availability statement

The data generated and/or analysed during the current study are not publicly available for legal/ethical reasons but are available from the corresponding author on reasonable request.

ORCID iDs

Junxu Li  <https://orcid.org/0000-0002-2189-0626>

Sabre Kais  <https://orcid.org/0000-0003-0574-5346>

References

- [1] Arute F *et al* 2019 Quantum supremacy using a programmable superconducting processor *Nature* **574** 505–10
- [2] Preskill J 2018 Quantum computing in the NISQ era and beyond *Quantum* **2** 79
- [3] Zhong H-S *et al* 2020 Quantum computational advantage using photons *Science* **370** 1460–3
- [4] Moll N *et al* 2018 Quantum optimization using variational algorithms on near-term quantum devices *Quantum Sci. Technol.* **3** 030503
- [5] Zhou L, Wang S-T, Choi S, Pichler H and Lukin M D 2020 Quantum approximate optimization algorithm: performance, mechanism, and implementation on near-term devices *Phys. Rev. X* **10** 021067
- [6] Pan J, Cao Y, Yao X, Li Z, Ju C, Chen H, Peng X, Kais S and Du J 2014 Experimental realization of quantum algorithm for solving linear systems of equations *Phys. Rev. A* **89** 022313
- [7] Huang H-Y, Bharti K and Rebentrost P 2019 Near-term quantum algorithms for linear systems of equations (arXiv:1909.07344)
- [8] Wossnig L, Zhao Z and Prakash A 2018 Quantum linear system algorithm for dense matrices *Phys. Rev. Lett.* **120** 050502
- [9] Huang Z, Wang H and Kais S 2006 Entanglement and electron correlation in quantum chemistry calculations *J. Mod. Opt.* **53** 2543–58
- [10] Kandala A, Mezzacapo A, Temme K, Takita M, Brink M, Chow J M and Gambetta J M 2017 Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets *Nature* **549** 242–6
- [11] Xia R, Bian T and Kais S 2017 Electronic structure calculations and the Ising Hamiltonian *J. Phys. Chem. B* **122** 3384–95
- [12] Parrish R M, Hohenstein E G, McMahan P L and Martinez T J 2019 Quantum computation of electronic transitions using a variational quantum eigensolver *Phys. Rev. Lett.* **122** 230401
- [13] Bian T and Kais S 2021 Quantum computing for atomic and molecular resonances *J. Chem. Phys.* **154** 194107
- [14] Xia R and Kais S 2020 Hybrid quantum-classical neural network for calculating ground state energies of molecules *Entropy* **22** 828
- [15] Xia R and Kais S 2020 Qubit coupled cluster singles and doubles variational quantum eigensolver ansatz for electronic structure calculations *Quantum Sci. Technol.* **6** 015001
- [16] Hu Z and Kais S 2020 A quantum encryption scheme featuring confusion, diffusion, and mode of operation (arXiv:2010.03062)
- [17] Li J, Hu Z and Kais S 2021 A practical quantum encryption protocol with varying encryption configurations (arXiv:2101.09314)
- [18] Peruzzo A, McClean J, Shadbolt P, Yung M-H, Zhou X-Q, Love P J, Aspuru-Guzik A and O'Brien J L 2014 A variational eigenvalue solver on a photonic quantum processor *Nat. Commun.* **5** 1–7
- [19] McClean J R, Romero J, Ryan B and Aspuru-Guzik A 2016 The theory of variational hybrid quantum-classical algorithms *New J. Phys.* **18** 023023
- [20] Sagastizabal R *et al* 2019 Experimental error mitigation via symmetry verification in a variational quantum eigensolver *Phys. Rev. A* **100** 010302
- [21] Ryabinkin I G, Genin S N and Izmaylov A F 2018 Constrained variational quantum eigensolver: quantum computer search engine in the Fock space *J. Chem. Theory Comput.* **15** 249–55
- [22] Nakanishi K M, Mitarai K and Fujii K 2019 Subspace-search variational quantum eigensolver for excited states *Phys. Rev. Res.* **1** 033062
- [23] Hadfield S A and Aho A V 2018 Quantum algorithms for scientific computing and approximate optimization ProQuest *Dissertations and Theses* Columbia University p 264

- [24] Barreiro J T *et al* 2011 An open-system quantum simulator with trapped ions *Nature* **470** 486–91
- [25] Hu Z, Xia R and Kais S 2020 A quantum algorithm for evolving open quantum dynamics on quantum computing devices *Sci. Rep.* **10** 1–9
- [26] Smart S E and Mazziotti D A 2021 Quantum solver of contracted eigenvalue equations for scalable molecular simulations on quantum computing devices *Phys. Rev. Lett.* **126** 070504
- [27] Head-Marsden K, Krastanov S, Mazziotti D A and Narang P 2021 Capturing non-Markovian dynamics on near-term quantum computers *Phys. Rev. Res.* **3** 013182
- [28] Pauls J A, Zhang Y, Berman G P and Kais S 2013 Quantum coherence and entanglement in the avian compass *Phys. Rev. E* **87** 062704
- [29] Xia R and Kais S 2018 Quantum machine learning for electronic structure calculations *Nat. Commun.* **9** 1–6
- [30] Biamonte J, Wittek P, Pancotti N, Rebentrost P, Nathan W and Lloyd S 2017 Quantum machine learning *Nature* **549** 195–202
- [31] Schuld M and Killoran N 2019 Quantum machine learning in feature Hilbert spaces *Phys. Rev. Lett.* **122** 040504
- [32] Huggins W, Patil P, Mitchell B, Whaley K B and Miles Stoudenmire E 2019 Towards quantum machine learning with tensor networks *Quantum Sci. Technol.* **4** 024001
- [33] Dixit V, Selvarajan R, Aldwairi T, Koshka Y, Novotny M A, Humble T S, Alam M A and Kais S 2021 Training a quantum annealing based restricted Boltzmann machine on cybersecurity data *IEEE Trans. Emerg. Top. Comput. Intell.* **1-12**
- [34] Roy S, Hu Z, Kais S and Bermel P 2021 Enhancement of photovoltaic current through dark states in donor–acceptor pairs of tungsten-based transition metal di-chalcogenides *Adv. Funct. Mater.* **31** 2100387
- [35] Sajjan M, Sureshbabu S H and Kais S 2021 Quantum machine-learning for eigenstate filtration in two-dimensional materials (arXiv:2105.09488)
- [36] Wilson B A, Kudyshev Z A, Kildishev A V, Kais S, Shalaev V M and Boltasseva A 2021 Machine learning framework for quantum sampling of highly-constrained, continuous optimization problems (arXiv:2105.02396)
- [37] Leshno M, Lin V Y, Pinkus A and Schocken S 1993 Multilayer feedforward networks with a nonpolynomial activation function can approximate any function *Neural Netw.* **6** 861–7
- [38] Daskin A 2018 A simple quantum neural net with a periodic activation function *IEEE Int. Conf. Systems, Man, and Cybernetics (SMC)* (Piscataway, NJ: IEEE) pp 2887–91
- [39] Peruš M 2000 Neural networks as a basis for quantum associative networks *Neural Netw. World* **10** 1001–13
- [40] Zak M and Williams C P 1998 Quantum neural nets *Int. J. Theor. Phys.* **37** 651–84
- [41] Cao Y, Guerreschi G G and Aspuru-Guzik A 2017 Quantum neuron: an elementary building block for machine learning on quantum computers (arXiv:1711.11240)
- [42] Schuld M, Sinayskiy I and Petruccione F 2015 Simulating a perceptron on a quantum computer *Phys. Lett. A* **379** 660–3
- [43] Coppersmith D 2002 An approximate Fourier transform useful in quantum factoring (arXiv:quant-ph/0201067)
- [44] Weinstein Y S, Pravia M A, Fortunato E M, Lloyd S and Cory D G 2001 Implementation of the quantum Fourier transform *Phys. Rev. Lett.* **86** 1889
- [45] Gilyén A, Su Y, Low G H and Nathan W 2019 Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics *Proc. 51st Annual ACM SIGACT Symp. Theory of Computing* pp 193–204
- [46] Barenco A, Bennett C H, Cleve R, DiVincenzo D P, Margolus N, Shor P, Sleator T, Smolin J A and Weinfurter H 1995 Elementary gates for quantum computation *Phys. Rev. A* **52** 3457
- [47] Aaronson S 2019 Shadow tomography of quantum states *SIAM J. Comput.* **49** STOC18–368