scientific reports

OPEN

Characterizing quantum circuits with qubit functional configurations

Zixuan Hu & Sabre Kais✉

We develop a systematic framework for characterizing all quantum circuits with qubit functional configurations. The qubit functional configuration is a mathematical structure that can classify the properties and behaviors of quantum circuits collectively. Major benefits of classifying quantum circuits in this way include: 1. All quantum circuits can be classified into corresponding types; 2. Each type characterizes important properties (such as circuit complexity) of the quantum circuits belonging to it; 3. Each type contains a huge collection of possible quantum circuits allowing systematic investigation of their common properties. We demonstrate the theory's application to analyzing the hardware-efficient ansatzes of variational quantum algorithms. For potential applications, the functional configuration theory may allow systematic understanding and development of quantum algorithms based on their functional configuration types.

The development of quantum algorithms with efficient quantum circuits has been a central part of quantum computation, which has seen enormous progress in the last 30 years in both the theoretical and experimental fronts[1–22]. Over the years, numerous quantum algorithms have been developed targeting a great variety of applications: these include the phase estimation algorithm[23], Shor's factorization algorithm[24], the Harrow-Hassidim-Lloyd algorithm for linear systems[25], the hybrid classical-quantum algorithms[26–29], the quantum machine learning algorithms[30,31], and quantum algorithms for open quantum dynamics[32–37].

Despite the success, the design of efficient quantum circuits to implement new quantum algorithms remains an accidental process, and a systematic way[38,39] to understand how quantum circuits work may lead to improvements in existing algorithms and discovery of new ones. To achieve decisive advantage over classical algorithms, quantum circuits must have the complexity that scales polynomially, not exponentially, with the number of qubits. Mathematically, if we consider all possible quantum circuits without any constraint, then the overwhelming majority of the quantum circuits have exponential scaling[40]. Consequently, designing quantum circuits that can demonstrate the advantage of quantum computation is essentially a process of characterizing and identifying the special cases with polynomial scaling out of the ocean of general cases with exponential scaling. To achieve this we need a systematic framework to classify large numbers of quantum circuits with finite mathematical structures: in this work we develop such a theory based on the "qubit functional configuration". The qubit functionals are first introduced in our previous work[41,42] as objects living in the dual space of the linear space formed by the basis states of the qubit state space. In this work we relate the qubit functionals to how the state vector entries are collectively modified by quantum gates. In particular, any arbitrary quantum circuit can be decomposed into alternating sequences of 1-qubit unitary gates and CNOT gates. Each CNOT sequence prepares the current quantum state into a layer of qubit functional configuration to specify the rule for the next 1-qubit gate sequence on how to collectively modify the state vector entries. All the layers together form a sequence of qubit functional configurations that defines a unique type of quantum circuits with great variety. Characterizing quantum circuits by these qubit functional configuration types has several major benefits: 1. all quantum circuits can be characterized by the corresponding functional configuration types; 2. each type characterizes important properties (such as the circuit complexity) of the quantum circuits belonging to it; 3. each type contains a huge collection of possible quantum circuits allowing systematic investigation of their common properties.

In addition, we demonstrate an application of the theory to the hardware-efficient ansatzes of variational quantum algorithms[43–46] by comparing the qubit functional configurations of two ansatzes used in Refs.[45] and [46]. We also propose a systematic process of creating any arbitrary functional configuration with the help of ancilla qubits.

Department of Chemistry, Department of Physics, Purdue Quantum Science and Engineering Institute, Purdue University, West Lafayette, IN 47907, USA. ✉email: kais@purdue.edu

1

For potential applications, the functional configuration picture may allow us to systematically find a minimal gate decomposition sequence of any given quantum circuit. The qubit functional configuration theory is so far as we know the first of its kind that can characterize huge numbers of quantum circuits together by finite mathematical structures—this may allow systematic understanding of quantum algorithms based on their quantum circuit types and thus benefit the development of new algorithms.

Theory of the qubit functional configuration

The qubit functionals.

An n-qubit quantum state vector can be written as $|\varphi\rangle = \sum_{i=0}^{2^n-1} a_i |i\rangle$, where the basis states $|i\rangle$ can be associated with the bit strings (i) with i in the binary form. For example a 3-qubit state vector has 8 basis states corresponding to 8 bit strings: $(000), (001), (010), \ldots, (111)$. As discussed in Ref.[41] the collection of all these bit strings can be considered as a 3-dimensional linear space V over the binary field $\{0, 1\}$ and any bit string can be expressed as a linear combination of the three basis vectors (100), (010), and (001): $\mathbf{v} = (q_1, q_2, q_3) = q_1(100) + q_2(010) + q_3(001)$, where the coefficients q_i's take values 0 or 1, and the addition "+" is bit-wise addition modulo 2. Now by the theory of linear vector spaces, the dual space V^* of V is formed by the linear functionals over V: $f(\mathbf{v}) = f\big((q_1, q_2, q_3)\big) = g_1 q_1 \oplus g_2 q_2 \oplus g_3 q_3$. Here the three basis functionals are $f^{(1)}\big((q_1, q_2, q_3)\big) = q_1$, $f^{(2)}\big((q_1, q_2, q_3)\big) = q_2$, $f^{(3)}\big((q_1, q_2, q_3)\big) = q_3$, the coefficients g_i's take values 0 or 1, and the "$\oplus$" is addition modulo 2. In Ref.[41] we have associated each functional with a "0" condition that specifies a half-set of the bit strings and studied the quantum condition space generated by these "0" conditions. In this work we call these functionals "qubit functionals" (because they are functionals on the qubit values) and focus on their roles in how the entries of the quantum state vectors are modified collectively by quantum gates.

Roles of elementary gates in modifying state vector entries.

The universality of quantum circuits says that any unitary operation can be decomposed into a sequence of 1-qubit unitaries and CNOT gates. Consider an arbitrary n-qubit quantum state $|\varphi_1\rangle = \sum_{i=0}^{2^n-1} a_i |i\rangle$ as the starting point, if we apply 1-qubit unitaries U_k only without CNOT gates, then at most one U_k can be applied on each qubit q_k without redundancy. This is because U_k's on different q_k's all commute, so if e.g. U_1 has already been applied to q_1, then any additional unitary V_1 applied to q_1 at any point after U_1 will be equivalent to a single gate $W_1 = V_1 U_1$ applied to q_1, thus applying more than one gates on the same qubit is redundant and can be reduced to just one gate. Next to examine the actual effects of 1-qubit unitaries, we study a 3-qubit state $|\varphi_1\rangle = \sum_{i=0}^{7} a_i |i\rangle$ without loss of generality:

$$
\varphi_1 = \begin{matrix} 000 \\ 001 \\ 010 \\ 011 \\ 100 \\ 101 \\ 110 \\ 111 \end{matrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \end{pmatrix} \rightarrow U_1\varphi_1 = \begin{pmatrix} u_1 a_0 + u_2 a_4 \\ u_1 a_1 + u_2 a_5 \\ u_1 a_2 + u_2 a_6 \\ u_1 a_3 + u_2 a_7 \\ u_2 a_0 - u_1 a_4 \\ u_2 a_1 - u_1 a_5 \\ u_2 a_2 - u_1 a_6 \\ u_2 a_3 - u_1 a_7 \end{pmatrix} ; \quad U_2\varphi_1 = \begin{pmatrix} u_1 a_0 + u_2 a_2 \\ u_1 a_1 + u_2 a_3 \\ u_2 a_0 - u_1 a_2 \\ u_2 a_1 - u_1 a_3 \\ u_1 a_4 + u_2 a_6 \\ u_1 a_5 + u_2 a_7 \\ u_2 a_4 - u_1 a_6 \\ u_2 a_5 - u_1 a_7 \end{pmatrix} ; \quad U_3\varphi_1 = \begin{pmatrix} u_1 a_0 + u_2 a_1 \\ u_2 a_0 - u_1 a_1 \\ u_1 a_2 + u_2 a_3 \\ u_2 a_2 - u_1 a_3 \\ u_1 a_4 + u_2 a_5 \\ u_2 a_4 - u_1 a_5 \\ u_1 a_6 + u_2 a_7 \\ u_2 a_6 - u_1 a_7 \end{pmatrix}
$$

(1)

where the 1-qubit unitary U_k has the same basic form $U = \begin{pmatrix} u_1 & u_2 \\ u_2 & -u_1 \end{pmatrix}$ applied to the kth qubit q_k: that is $U_1 = U \otimes I \otimes I, U_2 = I \otimes U \otimes I$, and $U_3 = I \otimes I \otimes U$ (u_1, u_2 are assumed real for cleaner notations; this assumption does not cause any reasoning or results below to lose generality as compared to using complex numbers; $u_1^2 + u_2^2 = 1$). Here we see that U_1 pairs a_0 with a_4, a_1 with a_5, a_2 with a_6, and a_3 with a_7. When we focus on one pair e.g. a_0 and a_4, the effect of U_1 is equivalent to $U \begin{pmatrix} a_0 \\ a_4 \end{pmatrix} = \begin{pmatrix} u_1 a_0 + u_2 a_4 \\ u_2 a_0 - u_1 a_4 \end{pmatrix}$, thus we can say a_0 plays the role of "0" in this 2-dimensional subspace while a_4 plays the role of "1". Now we see the effect of U_1 is dividing the total space into four 2-dimensional subspaces and then modifying the entries in pairs of $[(a_0, a_4), (a_1, a_5), (a_2, a_6), (a_3, a_7)]$ where the 1st entry in each parenthesis is considered "0" and the 2nd entry is considered "1". Similarly, the effect of U_2 can be represented by $[(a_0, a_2), (a_1, a_3), (a_4, a_6), (a_5, a_7)]$, and that of U_3 can be represented by $[(a_0, a_1), (a_2, a_3), (a_4, a_5), (a_6, a_7)]$. Note that U_1, U_2, and U_3 may be applied simultaneously but they keep the individual effects when considering a single gate alone. Now for U_1, $[(a_0, a_4), (a_1, a_5), (a_2, a_6), (a_3, a_7)]$ means that we have separated the total space into two half-spaces spanned by basis states with $q_1 = 0$: $\{|000\rangle \sim a_0, |001\rangle \sim a_1, |010\rangle \sim a_2, |011\rangle \sim a_3\}$ versus $q_1 = 1$: $\{|100\rangle \sim a_4, |101\rangle \sim a_5, |110\rangle \sim a_6, |111\rangle \sim a_7\}$; we then pair each $q_1 = 0$ term with a unique $q_1 = 1$ term and mix them to produce e.g. $U \begin{pmatrix} a_0 \\ a_4 \end{pmatrix} = \begin{pmatrix} u_1 a_0 + u_2 a_4 \\ u_2 a_0 - u_1 a_4 \end{pmatrix}$ or $U \begin{pmatrix} a_1 \\ a_5 \end{pmatrix} = \begin{pmatrix} u_1 a_1 + u_2 a_5 \\ u_2 a_1 - u_1 a_5 \end{pmatrix}$. Finally putting all these pairs back to the proper locations in the total vector, we obtain the vector result of $U_1\varphi_1$ as in Eq. (1). Similarly, U_2 separates the total space into two half-spaces spanned by basis states with $q_2 = 0$ versus $q_2 = 1$, while U_3 separates into half-spaces with $q_3 = 0$ versus $q_3 = 1$.

Now what happens if we apply some CNOT gates before the 1-qubit U_k's?

$$\varphi_1 = \begin{matrix} 000 \\ 001 \\ 010 \\ 011 \\ 100 \\ 101 \\ 110 \\ 111 \end{matrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \end{pmatrix} \xrightarrow{\text{CNOT}_{1\to2}} \begin{matrix} 000 \\ 001 \\ 010 \\ 011 \\ 100 \\ 101 \\ 110 \\ 111 \end{matrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_6 \\ a_7 \\ a_4 \\ a_5 \end{pmatrix} \xrightarrow{\text{CNOT}_{2\to3}} \varphi_1' = \begin{matrix} 000 \\ 001 \\ 010 \\ 011 \\ 100 \\ 101 \\ 110 \\ 111 \end{matrix} \begin{pmatrix} a_0 \\ a_1 \\ a_3 \\ a_2 \\ a_6 \\ a_7 \\ a_5 \\ a_4 \end{pmatrix} \tag{2}$$

In Eq. (2) we apply $\text{CNOT}_{1\to2}$ and then $\text{CNOT}_{2\to3}$ to produce φ_1', where $i \to j$ in the subscripts means q_i is the control and q_j is the target. By studying either the truth table of the CNOT gates or the results from previous works[41,47,48], the $\text{CNOT}_{1\to2}$ calculates $q_1 \oplus q_2$ and stores its value on q_2, and afterwards the $\text{CNOT}_{2\to3}$ calculates $q_1 \oplus q_2 \oplus q_3$ and stores its value on q_3. In other words, after $\text{CNOT}_{1\to2}$ and $\text{CNOT}_{2\to3}$ have been applied, the value on q_2 now represents the value of $q_1 \oplus q_2$, and the value on q_3 now represents the value of $q_1 \oplus q_2 \oplus q_3$, as compared to the state φ_1 before the two CNOT gates are applied. Since both $q_1 \oplus q_2$ and $q_1 \oplus q_2 \oplus q_3$ are linear functionals in V^*, we can understand the vector φ_1' as having a functional configuration of $(f_1 = q_1, f_2 = q_1 \oplus q_2, f_3 = q_1 \oplus q_2 \oplus q_3)$ as compared to φ_1 having $(f_1 = q_1, f_2 = q_2, f_3 = q_3)$. Now by the same reasoning as the previous paragraph we predict that, if we apply 1-qubit unitaries to φ_1' now, U_1 will still separate the total space into two half-spaces spanned by basis states with $q_1 = 0$ versus $q_1 = 1$, but exactly which "0" term is paired with which "1" term will be different. U_2 now will separate into two half-spaces with $q_1 \oplus q_2 = 0$ versus $q_1 \oplus q_2 = 1$, and U_3 will separate into two half-spaces with $q_1 \oplus q_2 \oplus q_3 = 0$ versus $q_1 \oplus q_2 \oplus q_3 = 1$. Indeed this prediction is verified by carrying out the algebra on φ_1' in the same manner as in Eq. (1):

$$\varphi_1' = \begin{pmatrix} a_0 \\ a_1 \\ a_3 \\ a_2 \\ a_6 \\ a_7 \\ a_5 \\ a_4 \end{pmatrix} \to U_1 \varphi_1' = \begin{pmatrix} u_1 a_0 + u_2 a_6 \\ u_1 a_1 + u_2 a_7 \\ u_1 a_3 + u_2 a_5 \\ u_1 a_2 + u_2 a_4 \\ u_2 a_0 - u_1 a_6 \\ u_2 a_1 - u_1 a_7 \\ u_2 a_3 - u_1 a_5 \\ u_2 a_2 - u_1 a_4 \end{pmatrix}; \quad U_2 \varphi_1' = \begin{pmatrix} u_1 a_0 + u_2 a_3 \\ u_1 a_1 + u_2 a_2 \\ u_2 a_0 - u_1 a_3 \\ u_2 a_1 - u_1 a_2 \\ u_1 a_6 + u_2 a_5 \\ u_1 a_7 + u_2 a_4 \\ u_2 a_6 - u_1 a_5 \\ u_2 a_7 - u_1 a_4 \end{pmatrix}; \quad U_3 \varphi_1' = \begin{pmatrix} u_1 a_0 + u_2 a_1 \\ u_2 a_0 - u_1 a_1 \\ u_1 a_3 + u_2 a_2 \\ u_2 a_3 - u_1 a_2 \\ u_1 a_6 + u_2 a_7 \\ u_2 a_6 - u_1 a_7 \\ u_1 a_5 + u_2 a_4 \\ u_2 a_5 - u_1 a_4 \end{pmatrix} \tag{3}$$

In Eq. (3) we see the effect of U_1 is $[(a_0, a_6), (a_1, a_7), (a_2, a_4), (a_3, a_5)]$ with the same "0" entries and "1" entries but different pairing from the previous $[(a_0, a_4), (a_1, a_5), (a_2, a_6), (a_3, a_7)]$. On the other hand, the effect of U_2 is $[(a_0, a_3), (a_1, a_2), (a_6, a_5), (a_7, a_4)]$ where the "0" entries are now $\{a_0, a_1, a_6, a_7\}$—these entries in the original φ_1 vector (not φ_1') correspond to $\{|000\rangle \sim a_0, |001\rangle \sim a_1, |110\rangle \sim a_6, |111\rangle \sim a_7\}$—so indeed for these entries $q_1 \oplus q_2 = 0$. Similarly the effect of U_3 is $[(a_0, a_1), (a_3, a_2), (a_6, a_7), (a_5, a_4)]$ where the "0" entries are $\{a_0, a_3, a_6, a_5\}$ which correspond to $q_1 \oplus q_2 \oplus q_3 = 0$ in the original φ_1 vector.

Characterizing quantum circuits with functional configurations. With these results we are now ready to introduce the idea of using qubit functional configurations to characterize quantum circuits. Starting with an arbitrary initial state $|\varphi_1\rangle = \sum_{i=0}^{7} a_i |i\rangle$, we can consider the qubits as functional holders with the initial qubit functional configuration of $(f_1 = q_1, f_2 = q_2, f_3 = q_3)$. As long as there is no CNOT gate applied, this functional configuration remains in place and specifies a unique rule for all possible 1-qubit unitaries: which entries are considered "0", which are considered "1", and how the "0" and "1" entries are paired for each U_k. As long as the configuration stays in place, a maximum number of $n = 3$ U_k's can be applied with one unitary on each qubit, and any additional 1-qubit unitary would be redundant. After the last U_k under this configuration has been applied and before the first CNOT happens, the state changes to $|\varphi_2\rangle = \sum_{i=0}^{7} b_i |i\rangle$ with new entries in the state vector. Now suppose a sequence of CNOT gates are applied without any 1-qubit unitary in between, right before the next U_k happens, the entries are not modified but only shuffled, and this creates a new layer of functional configuration such as $(f_1 = q_1, f_2 = q_1 \oplus q_2, f_3 = q_1 \oplus q_2 \oplus q_3)$ discussed above. This second layer of functional configuration now in place specifies its own unique rule for all possible 1-qubit unitaries that happen after, before the next CNOT gate is applied. Repeating this process we can analyze any arbitrary quantum circuits as follows:

In Fig. 1 an arbitrary quantum circuit can be decomposed into alternating sequences of 1-qubit U_k and CNOT gates. Each CNOT sequence prepares the current quantum state into a new layer of qubit functional configuration to specify the rule for the next U_k sequence on how the total space should be separated into "0" entries and "1" entries and how the "0" and "1" entries are paired. In particular, any U_k finds the appropriate "0" and "1" entries according to the kth functional f_k and the pairing pattern is defined by the entire functional configuration. After this U_k sequence is completed, right before the next CNOT sequence begins, a quantum state with new entries is created, which can then be used as the initial state for the next iteration. This process can be repeated many times until the end of the circuit where the last layer of functional configuration specifies the rule for the final U_k sequence to create the final state. Together, all the layers of functional configurations form a sequence of configurations that allows us to define a type of quantum circuits.

Note that the decomposition of an arbitrary quantum circuit into 1-qubit gates and CNOT gates is for generating the qubit functional configurations—i.e. it is for theoretical analysis only and thus does not increase the burden of implementing the circuits. In addition, it is well known that[40] such a decomposition will only increase
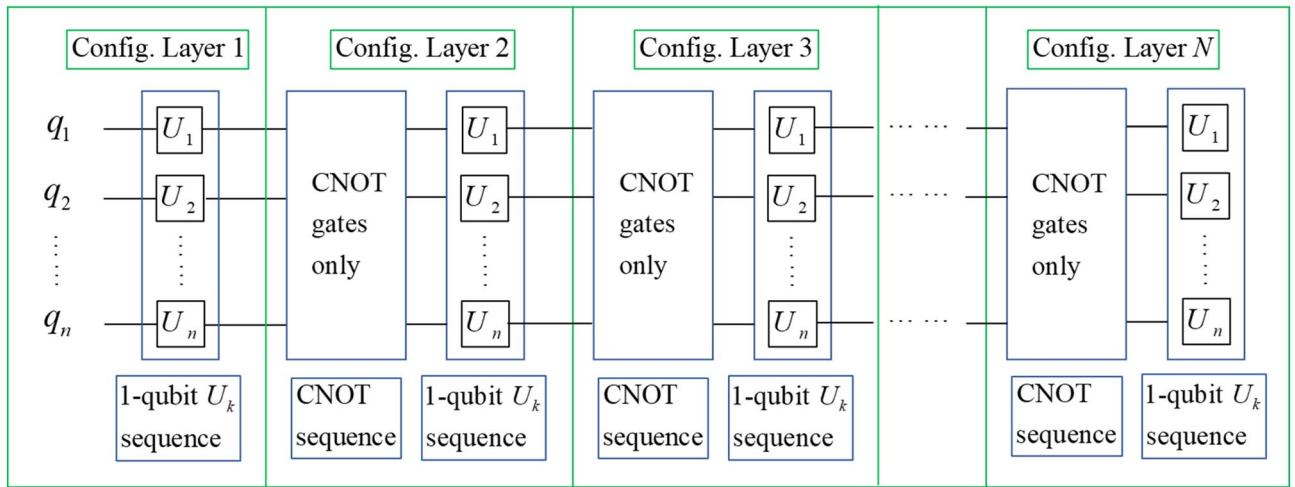
Figure 1. Graphical illustration of an arbitrary quantum circuit analyzed with multiple layers of qubit functional configurations. Note here we use the symbol U_k to mean "any 1-qubit gate on the kth qubit", thus U_k is not a particular gate and may represent different 1-qubit gates across different layers.

the depth of the sequence polynomially, i.e. if the original quantum circuit length is polynomially scaled with the number of qubits, then the decomposition sequence is also polynomially scaled. This means we can classify and characterize any polynomial quantum circuit efficiently with our qubit functional configuration method.

Next we demonstrate the application of the functional configuration picture in Fig. 1 to analyzing the hardware-efficient ansatzes that are widely used in variational quantum algorithms[43–46]. A critical step in a variational quantum algorithm is to select the best ansatzes that are essential to the convergence rate and the quality of final results. Ansatzes are commonly specified by a general structure of quantum circuits and allow multiple continuous parameters to change in the process of variation. The ansatzes can be readily analyzed by our theory because their general structures can be described by the qubit functional configurations which allow the continuous variation of the parameters. To see how this works, the hardware-efficient ansatzes commonly include many identical layers, and within each layer there are two sub-layers: one sub-layer of parameterized 1-qubit unitaries and one sub-layer of two-qubit entanglers. The two-qubit entanglers are usually all CNOT gates with no 1-qubit unitaries in between, such that each entangler sub-layer can be considered as a single layer of qubit functional configuration as described above—this makes the hardware-efficient ansatzes natural application candidates for the qubit functional configuration theory. In the following examples, the ansatzes used in Ref.[45]. and Ref.[46]. are shown in Figs. 2 and 3 respectively, and the corresponding qubit functional configurations are shown in Eqs. (4) and (5) respectively.

The qubit functional configuration of the ansatz in Ref.[45]

$$f_1 = q_2 \oplus q_3 \oplus q_4 \oplus q_5 \oplus q_6, \quad f_2 = q_1 \oplus q_2, \quad f_3 = q_1 \oplus q_2 \oplus q_3, \quad f_4 = q_1 \oplus q_2 \oplus q_3 \oplus q_4$$
$$f_5 = q_1 \oplus q_2 \oplus q_3 \oplus q_4 \oplus q_5, \quad f_6 = q_1 \oplus q_2 \oplus q_3 \oplus q_4 \oplus q_5 \oplus q_6 \tag{4}$$

The qubit functional configuration of the ansatz in Ref.[46]

$$f_1 = q_1 \oplus q_2, \quad f_2 = q_2, \quad f_3 = q_1 \oplus q_2 \oplus q_3, \quad f_4 = q_2 \oplus q_4, \quad f_5 = q_4 \oplus q_5 \oplus q_6, \quad f_6 = q_6 \tag{5}$$

In Eqs. (4) and (5) the differences between the two ansatzes in Figs. 2 and 3 are clearly shown by the corresponding qubit functional configurations. For example, in Eq. (5) f_2 and f_6 still keep their initial values q_2
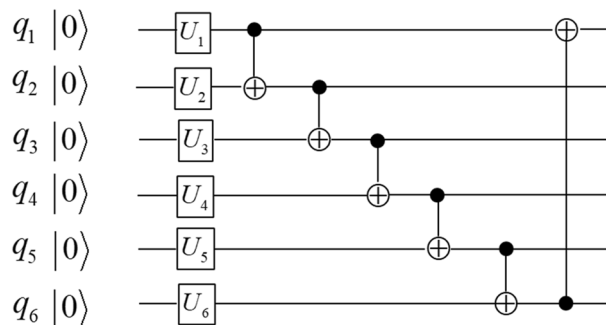
Figure 2. The quantum circuit of one layer of the hardware-efficient ansatz in Ref.[45].
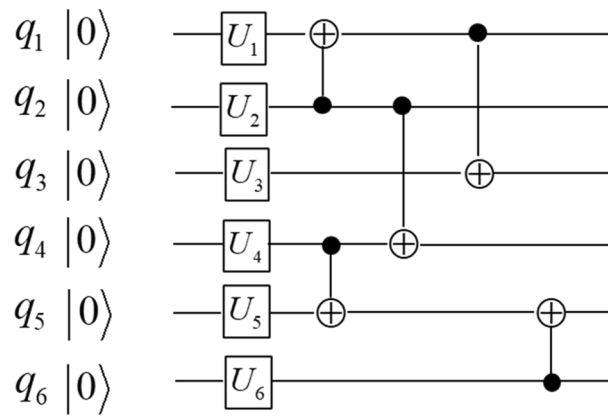
Figure 3. The quantum circuit of one layer of the hardware-efficient ansatz in Ref.[46].

and q_6, while in Eq. (4) all the qubit functionals have been changed. In general, the qubit functionals in Eq. (4) contain more qubit terms than those in Eq. (5), thus we should expect the qubits of the circuit in Fig. 2 to be more interconnected than those in Fig. 3. In this particular case, the ansatzes will repeat the same functional configuration for all the following layers; however, if we apply the theory to other types of quantum circuits, such as the quantum Fourier transform circuit studied in the Supplementary Information (SI) Section S1, each layer may have a different functional configuration and the whole sequence of configuration layers characterize the circuit's properties. So far the comparison of Eqs. (4) and (5) focuses on the structural differences between the two qubit functional configurations. In the SI Section S4 we discuss how such differences would translate into circuit properties when used in conjunction with the "unitary dependence theory" proposed in our recent study[39].

As can be seen from the procedure described in and around Fig. 1, given any quantum circuit, we can generate its qubit functional configuration layers by analyzing the quantum gates one by one. This means the cost of applying the method to a given quantum circuit should be comparable to the circuit length, which can be polynomial or exponential in the number of qubits. Indeed, the qubit functional configuration describes the characteristic properties of a quantum circuit and thus should have the same complexity scaling as the circuit itself.

Studying quantum circuits in terms of the qubit functional configuration types has various benefits:

1. Because the process in Fig. 1 is entirely general, any arbitrary quantum circuit can be analyzed by this picture and thus belongs to one of the types defined by the functional configuration sequence.
2. As explained earlier, for any particular layer of functional configuration, a maximum number of n 1-qubit unitaries can be applied without redundancy, with one unitary for each qubit. In addition, the number of CNOT gates required to achieve any given functional configuration is also bounded. Therefore the total length L of a quantum circuit is closely related to the total number N of functional configuration layers in the circuit. In other words, the functional configuration picture captures the important property of circuit complexity. The exact relation between L and N will be derived in section "Circuit length in relation to the number of layers of functional configurations".
3. More importantly, any particular sequence of the functional configurations contains the characteristic information of how the state vector entries are collectively modified by 1-qubit unitaries, while not fixing the actual U_k sequences—this allows us to group a large collection of quantum circuits together to be classified under a single type defined by the functional configuration sequence. In fact, as the parameters of any 1-qubit unitary can be varied continuously in the complex number domain, there are uncountably infinite number of possible quantum circuits with a given length. However, for a functional configuration sequence of a given length, there are only finite number of possible sequences, thus by this picture we are able to classify an uncountably infinite collection with finite number of types.

Any particular type of quantum circuits has its unique sequence of functional configurations, and each layer of functional configuration defines a unique way by which 1-qubit unitaries work on the state vector entries. As detailed in Ref.[41], the duality between the linear functional space V^* and the bit string space V leads to a one-to-one correspondence between each functional and a unique partition of the bit strings—defined by two half-sets specified by setting "0" and "1" for the functional value respectively. When U_k acts on q_k, the partition of f_k specifies which entries are "0" and which are "1", while the partitions of all other functionals define how the "0" entries are paired with the "1" entries. Consequently if a particular functional f_k appears in one configuration but not in another, then the 1-qubit unitaries will behave differently in the two configurations and thus two quantum circuits of different configuration types have different characteristics. Even if two configurations contain the same collection of functionals, two different permutations of these functionals will in general produce different quantum circuits. As detailed in the Supplementary Information (SI) Section S2, if there is only one layer, then e.g. the configuration (f_1, f_2, f_3) can be considered somewhat equivalent to (f_3, f_1, f_2) because U_k in (f_1, f_2, f_3) obeys the same rules as U_{k+1} in (f_3, f_1, f_2) with $U_{3+1} = U_1$. However, when multiple layers of configurations are present, then each corresponding configuration must permute in the same way for this equivalence to hold:

e.g. $(f_1, f_2, f_3) \rightarrow (f_4, f_5, f_6)$ is equivalent to $(f_3, f_1, f_2) \rightarrow (f_6, f_4, f_5)$ but not to $(f_3, f_1, f_2) \rightarrow (f_4, f_6, f_5)$. Hence we can conclude that the types of quantum circuits as defined by different sequences of functional configurations do not in general overlap with each other. When studying properties of quantum circuits with the gate decomposition sequence one often encounters the problem of having multiple gate sequences that produce the same total unitary operation, which means that gate sequences do not uniquely characterize quantum circuits. In the functional configuration picture however, all these "equivalent" gate sequences must produce the same functional configuration sequence, and the types defined by the functional configuration sequences uniquely characterize quantum circuits. This is another benefit offered by the functional configuration picture. A detailed example of how different gate sequences can produce the same functional configuration is presented in the SI Section S3.

Number of possible functional configuration types. Now a natural question is to ask how many possible functional configuration types there are for a single layer and over a fixed number of layers. For a single layer, because the functional space is the dual of the bit string space with 2^n elements, there are 2^n functionals (for details see Ref.[41]). Excluding the "0" functional, there are $2^n - 1$ to choose from, and n locations to allocate them. The most naïve count would be $(2^n - 1)^n$, but not all of these are valid configurations. To see this first note that it is impossible for duplicate functionals to appear in two or more different locations. This is because e.g. if we have a configuration of $(f_1, f_2 = f_1, f_3..., f_n)$ then the state vector will only have entries associated with the basis states $|00...\rangle$ and $|11...\rangle$ but not $|01...\rangle$ and $|10...\rangle$ (half of all dimensions have been lost for the state vector), and this cannot be always satisfied when the initial state are chosen arbitrarily. It turns out this non-duplication rule is a special example of a more general condition for valid configurations: for a configuration of $(f_1, f_2, f_3..., f_n)$, none of the f_k's for $k = 1, 2, ..., n$ can be the sum of any number of other functionals. This is because otherwise if e.g. $f_7 = f_1 + f_2 + f_5$ then the value of f_7 is no longer free such that the subspace defined by (f_1, f_2, f_5, f_7) becomes (f_1, f_2, f_5) and half of the dimensions have been lost for the state vector. When we view the functionals as members of the linear space V^*, then requiring any f_k is not the sum of any number of other functionals is the same as requiring linear independence for all f_k's for $k = 1, 2, ..., n$ such that $\sum_{k=1}^{n} c_k f_k = 0$ has only the trivial solution of all c_k's being 0. Note that this linear independence is automatically enforced by using CNOT gates to create functional configurations. To see this start with any configuration of $(f_1, f_2, f_3..., f_n)$ with all functionals linearly independent, then applying $CNOT_{j \rightarrow h}$ will replace f_h with $f_j + f_h$ which is still linearly independent from all other functionals. So any CNOT gate cannot break the linear independence when starting with linear independence. The initial configuration of any quantum circuit is $(f_1 = q_1, f_2 = q_2, f_3 = q_3..., f_n = q_n)$ which has linear independence for all functionals, so indeed any CNOT sequence will automatically enforce the linear independence.

Now back to counting the possible configurations for a single layer, considering linear independence there are $2^n - 1$ choices for f_1, $2^n - 2$ for f_2, ..., for f_k there are N_f number of choices:

$$N_f(k, n) = 2^n - 1 - \sum_{i=1}^{k-1} C(k-1, i) = 2^n - 2^{k-1} \tag{6}$$

So the total possible number of configurations for one layer is:

$$N_c(n) = \prod_{k=1}^{n} N_f(k, n) = \prod_{k=1}^{n} \left(2^n - 2^{k-1}\right) \tag{7}$$

Now over multiple layers, for clarity we reinforce the definition of layers of functional configurations. Referring to Fig. 1, start with the initial state φ_1 and the initial configuration $(f_1 = q_1, f_2 = q_2, f_3 = q_3..., f_n = q_n)$, if the first gate applied is a 1-qubit unitary (i.e. the case in Fig. 1), we consider the initial configuration as the first layer; if the first gate is a CNOT gate (i.e. the case obtained by removing the first U_k sequence and combining the first two blocks in Fig. 1), we consider the configuration created by the first CNOT sequence as the first layer. After the first layer has been established, a sequence of 1-qubit unitaries can be applied according to the rule set by this configuration, and as long as there is no CNOT gate, the system stays in the first layer. Right before the next CNOT sequence happens, the state changes to φ_2, and the configuration is reset to $(f_1 = q_1, f_2 = q_2, f_3 = q_3..., f_n = q_n)$. Now the next CNOT sequence applies, and right before the next 1-qubit unitary sequence we create the configuration of the second layer, which stays in place until the next CNOT sequence. Subsequent layers can be defined accordingly. We require that all layers after the first one cannot be the initial configuration $(f_1 = q_1, f_2 = q_2, f_3 = q_3..., f_n = q_n)$—because otherwise no CNOT gate is applied after the reset and we can combine this layer with the previous one—then the total possible number of types defined by N layers of functional configurations is:

$$N_t(n, N) = \frac{N_c(n)(N_c(n) - 1)^{N-1}}{n!} = \frac{\prod_{k=1}^{n} \left(2^n - 2^{k-1}\right) \left[\prod_{k=1}^{n} \left(2^n - 2^{k-1}\right) - 1\right]^{N-1}}{n!} \tag{8}$$

where the $(N_c(n) - 1)^{N-1}$ accounts for "all layers after the first one cannot be the initial configuration", and $n!$ accounts for the equivalence due to permutation of functionals in the first layer. To get a concrete idea of N_t, for a 5-qubit system (like the simplest IBM quantum devices[49] but with unlimited connectivity between qubits) over 1 layer of functional configuration, there are:

$$N_t(5,1) = \frac{\prod_{k=1}^{5}\left(2^5 - 2^{k-1}\right)}{5!} = 83328 \tag{9}$$

Over 3 layers of functional configurations, there are:

$$N_t(5,3) = \frac{\prod_{k=1}^{5}\left(2^5 - 2^{k-1}\right)\left[\prod_{k=1}^{5}\left(2^5 - 2^{k-1}\right) - 1\right]^2}{5!} = 8.3317318 \times 10^{18} \tag{10}$$

So the number of types N_t can be a gigantic number even with quite small values for n and N, but this is reasonable because we are considering all quantum circuits that are ever possible on n qubits and N layers with a finite number of types. In fact, the gigantic number N_t illustrates the immense potential of quantum computation as a small number of qubits can support such an enormous variety of quantum circuits over few layers. In addition, as detailed in the next section (section "Circuit length in relation to the number of layers of functional configurations"), the total number of gates required to realize any one of the N_t possible functional configurations is polynomial in both the qubit number n and the layer number N. This means all of the N_t functional configurations can be reached efficiently with a polynomial circuit if the layer number N is polynomial in n—consequently, although N_t is gigantic, as long as N is polynomial, we are indeed classifying all polynomial circuits that are possible with Eq. (8), such that the classification is relevant to our practical need in quantum computing to design polynomial circuits that can be efficiently implemented.

Circuit length in relation to the number of layers of functional configurations.

How does the number N of layers relate to the length of the quantum circuit? As explained earlier, for any particular layer of functional configuration, a maximum number of n 1-qubit unitaries can be applied without redundancy, with one unitary for each qubit. So for any quantum circuit with N layers of functional configurations, the total number N_U of 1-qubit unitaries is $N \leq N_U \leq nN$ (at least one unitary for each layer). This means that once the circuit type defined by the sequence of functional configurations has been fixed, the length of the circuit can only vary by up to nN. Now if more generally the circuit type is not fixed but the number of layers is fixed, to evaluate how many CNOT gates are possible for one layer, we need to study how many CNOT gates are required to reach an arbitrary functional configuration. Starting with any functional configuration $(f_1, f_2, f_3 \dots, f_n)$, applying $\text{CNOT}_{j \to h}$ will replace f_h with $f_j + f_h$ such that the process of creating functional configurations is dynamical and it is difficult to exactly evaluate how many CNOT gates are required. However, we can calculate an upper bound for the number of CNOT gates by storing the initial configuration of $(f_1 = q_1, f_2 = q_2, f_3 = q_3 \dots, f_n = q_n)$ on n ancilla qubits. To do this we start with the initial configuration $(f_1 = q_1, f_2 = q_2, f_3 = q_3 \dots, f_n = q_n)$ on the computational qubits, prepare the ancilla qubits as $q_{a_1} = |0\rangle$, $q_{a_2} = |0\rangle$, ..., $q_{a_n} = |0\rangle$ and apply the gates $\text{CNOT}_{k \to a_k}$ for $k = 1, \dots, n$ with the computational qubits as controls and the ancilla qubits as targets, such to copy the configuration $(f_{a_1} = q_1, f_{a_2} = q_2, f_{a_3} = q_3 \dots, f_{a_n} = q_n)$ on the ancilla qubits. This way when we modify the configuration on the computational qubits, the ancilla qubits still retain the initial configuration. After this we can create the wanted functionals one by one, e.g. if for the next layer the wanted configuration requires $f_1 = q_2 \oplus q_3 \oplus q_5$, then we can use the corresponding ancilla qubits as controls and the 1st computational qubit as target such as $\text{CNOT}_{a_1 \to 1}$, $\text{CNOT}_{a_2 \to 1}$, $\text{CNOT}_{a_3 \to 1}$, $\text{CNOT}_{a_5 \to 1}$ (order does not matter), and we will create $q_2 \oplus q_3 \oplus q_5$ on the 1st functional location. Note here $\text{CNOT}_{a_1 \to 1}$ is to remove q_1 from f_1. As the ancilla configuration is never changed in this process, we can repeat the same procedure when creating f_2, f_3, and all the way to f_n. This systematic process of creating any functional configuration is illustrated in Fig. 4 with a 3-qubit example.

With this process, for each f_k, the functional that requires the most CNOT gates is the sum of all qubits except q_k, e.g. $f_1 = q_2 \oplus q_3 \oplus \dots \oplus q_n$, which requires n CNOT gates. Therefore the number of CNOT gates required to reach any functional configuration cannot exceed $n \times n = n^2$. The situation is similar for all subsequent layers, hence the number of CNOT gates over N layers cannot exceed n^2N. In practice this upper bound may not be reached due to the linear independence constraint. Overall to achieve any single layer of configuration we need at most n $\text{CNOT}_{k \to a_k}$ gates and n^2 $\text{CNOT}_{a_j \to k}$ gates, and the total over N layers is $(n^2 + n)N$ CNOT gates. The least number we need is one CNOT for all the layers after the first one, so it is $(N - 1)$ gates, and we have $N - 1 \leq N_{\text{CNOT}} \leq (n^2 + n)N$, with the right equality not necessarily achievable. Now we see that even with just 5 qubits and a single layer $N = 1$, the possible length of the quantum circuits is $1 \leq N_U \leq 5$ 1-qubit unitaries and $0 \leq N_{\text{CNOT}} \leq 30$ CNOT gates, so the number of types at 83,328 in Eq. (9) becomes more reasonable. Here we remark that on a case-by-case basis, the procedure to create functional configurations by ancilla qubits may be far less efficient compared to working with the computational qubits only. For example, the configuration $(f_1 = q_1, f_2 = q_1 \oplus q_2, f_3 = q_1 \oplus q_2 \oplus q_3)$ can be easily reached by $\text{CNOT}_{1 \to 2}$ and then $\text{CNOT}_{2 \to 3}$; while using ancilla qubits we first need three $\text{CNOT}_{k \to a_k}$ gates and then $\text{CNOT}_{a_1 \to 2}$, $\text{CNOT}_{a_1 \to 3}$, and $\text{CNOT}_{a_2 \to 3}$, requiring 6 CNOT gates in total. However the ancilla method is more systematic thus to allow us to bound the maximum number of CNOT gates needed. Even with this inefficiency, the upper bound of nN and $(n^2 + n)N$ are both polynomial in n and N such that if N is polynomial in n then the maximum length of the quantum circuits classified by N layers of functional configurations is also polynomial in n: this means that not only the quantum circuits belonging to the same type are either all polynomial or all exponential, but also the quantum circuits of types with the same number of layers are either all polynomial or all exponential. In other words, the functional configuration picture captures the important property of circuit complexity.
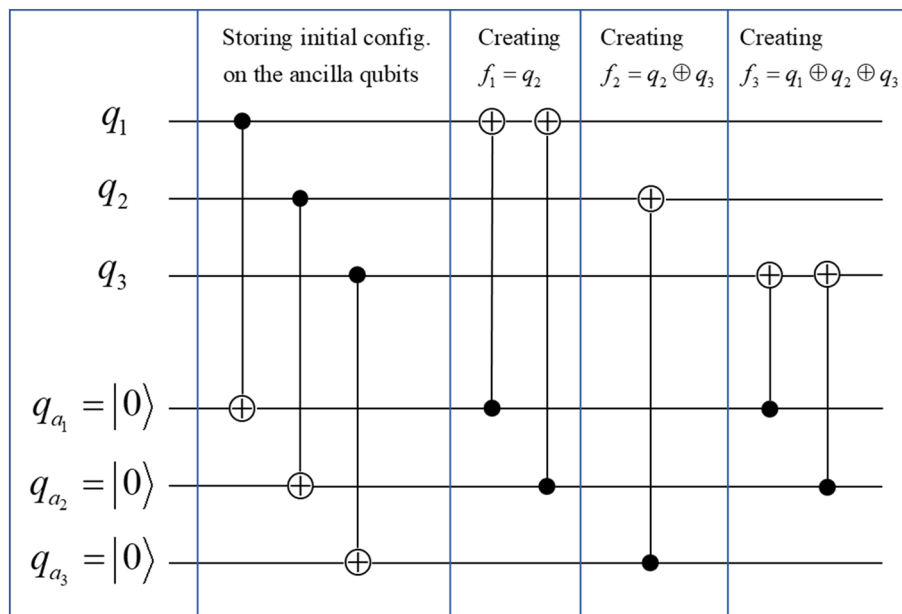
Figure 4. A 3-qubit example of systematically creating any functional configuration by first storing the initial configuration on 3 ancilla qubits and then creating the functionals one by one without changing the ancilla qubits.

Potential applications

The qubit functional configuration picture described in section "Theory of the qubit functional configuration" provides a systematic framework for characterizing quantum circuits. As discussed in section "Characterizing quantum circuits with functional configurations", the framework assigns all possible quantum circuits of a given length into finite types that in general do not overlap. In addition, the characteristic information of how the state vector entries are collectively modified by 1-qubit unitaries is described by the functional configuration sequence. In the following we discuss potential applications provided by this new theory.

Quantum algorithm development. The classification of quantum circuits will allow us to systematically study circuits of the same type or similar types. Numerous quantum algorithms with diverse mechanisms and functionalities have been proposed in the past 30 years, yet these discoveries are mostly made on a case-by-case basis. A systematic study of these algorithms based on the functional configuration types will potentially help us understand why they work, and such knowledge may lead to improvements on existing algorithms and discovery of new algorithms. As demonstrated by the application examples in section "Characterizing quantum circuits with functional configurations", The ansatzes used in variational quantum algorithms[43–46] are natural candidates for classification studies using the qubit functional configuration theory. The hybrid classical-quantum platform of the variational quantum algorithms has led to successful applications such as the variational quantum eigensolver (VQE)[26] and the quantum approximate optimization algorithm (QAOA)[28]. The properties and capabilities of variational quantum algorithms have been studied by descriptors such as expressibility and entangling capability[50] that describe how much of the solution space can be explored by varying the parameters. On the other hand, our qubit functional configuration theory provides a different perspective by classifying various ansatzes into finite mathematical structures such that their behaviors and properties can be studied collectively. In other words, the qubit functional configuration picture may allow us to systematically understand why certain ansatzes have better (or worse) performances when treating different kinds of problems.

The minimal gate decomposition of a quantum circuit. The qubit functional configuration picture may allow us to systematically find a minimal gate decomposition sequence of any given quantum circuit. Without the picture, the gate decomposition of a given quantum circuit is not unique (see section "Characterizing quantum circuits with functional configurations" and the SI Section S3), and there is no systematic approach to find the minimal gate sequence or verify that a known gate sequence is minimal. With the picture, for any given quantum circuit, if one gate decomposition sequence (non-minimal) is known, we will then find the functional configuration sequence that defines the type for the circuit. Now for each layer of functional configuration, the minimal CNOT gate sequence to create this configuration is a well-defined sequence. Joining the minimal CNOT sequence for each configuration layer together and inserting the appropriate 1-qubit unitaries, we will have the minimal overall gate sequence for the quantum circuit. A worked-out example of how this may work is found in the SI SectionS2. Even further, as in Eq. (7) there are only finite number of possible functional configurations for a single layer, thus for small numbers n of qubits, we could even construct a complete dictionary of all

possible functional configurations with the corresponding minimal gate sequences, such that the minimal gate sequence for any arbitrary quantum circuit can be retrieved by searching the functional configuration index.

Conclusion

In this work we have developed a theory of characterizing all quantum circuits with qubit functional configurations. In particular, an arbitrary quantum circuit can be decomposed into alternating sequences of 1-qubit unitary U_k gates and CNOT gates. Each CNOT sequence prepares the current quantum state into a layer of qubit functional configuration to specify the rule for the next U_k sequence on how to collectively modify the state vector entries. All the layers together form a sequence of functional configurations that defines a unique type of quantum circuits with great variety. For uncountably infinite possible quantum circuits, there are only finite number of functional configuration types for a given length, and this number has been evaluated in detail. More importantly, these types uniquely characterize properties of the corresponding quantum circuits. We have demonstrated an application of the theory to the hardware-efficient ansatzes used in variational quantum algorithms by analyzing them in terms of the functional configurations. We have also proposed a systematic procedure of creating any arbitrary functional configuration by storing the initial configuration on ancilla qubits. For potential applications, the functional configuration picture captures the important property of circuit complexity of any given quantum circuit and may allow us to define a minimal sequence for it. So far as we know, the qubit functional configuration theory is the first of its kind that can characterize huge numbers of quantum circuits by finite mathematical structures, and thus may lead to systematic development of efficient quantum circuits that scale polynomially with the qubit number.

Data availability

All data generated or analyzed during this study are included in this published article and its supplementary information files.

References

1. Georgescu, I. M., Ashhab, S. & Nori, F. Quantum simulation. *Rev. Mod. Phys.* **86**(1), 153–185 (2014).
2. Montanaro, A., Quantum algorithms: an overview. *npj Quantum Inf.* **2**(1): 15023 (2016).
3. Cao, Y. *et al.* Quantum chemistry in the age of quantum computing. *Chem. Rev.* **119**(19), 10856–10915 (2019).
4. Albash, T. & Lidar, D. A. Adiabatic quantum computation. *Rev. Mod. Phys.* **90**(1), 015002 (2018).
5. Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum* **2**, 79 (2018).
6. Kais, S., ed. *Quantum Information and Computation for Chemistry.* Quantum Information and Computation for Chemistry. 2014, Wiley.
7. Preskill, J., *Quantum computing 40 years later.* arXiv:2106.10522 [quant-ph] (2021).
8. Arute, F. *et al.* Quantum supremacy using a programmable superconducting processor. *Nature* **574**(7779), 505–510 (2019).
9. Boixo, S. *et al.* Evidence for quantum annealing with more than one hundred qubits. *Nat. Phys.* **10**(3), 218–224 (2014).
10. Linke, N. M. *et al.* Experimental comparison of two quantum computing architectures. *Proc. Natl. Acad. Sci.* **114**(13), 3305 (2017).
11. Carolan, J. *et al.* Universal linear optics. *Science* **349**(6249), 711 (2015).
12. Zhong, H.-S. *et al.* Quantum computational advantage using photons. *Science* **370**(6523), 1460 (2020).
13. Gong, M. *et al.* Quantum walks on a programmable two-dimensional 62-qubit superconducting processor. *Science* **372**(6545), 948 (2021).
14. Blatt, R. & Roos, C. F. Quantum simulations with trapped ions. *Nat. Phys.* **8**(4), 277–284 (2012).
15. Bruzewicz, C. D. *et al.* Trapped-ion quantum computing: Progress and challenges. *Appl. Phys. Rev.* **6**(2), 021314 (2019).
16. Huang, H.-L. *et al.* Superconducting quantum computing: a review. *Sci. China Inf. Sci.* **63**(8), 180501 (2020).
17. Havlíček, V. *et al.* Supervised learning with quantum-enhanced feature spaces. *Nature* **567**(7747), 209–212 (2019).
18. Huang, H.-L. *et al.* Experimental quantum generative adversarial networks for image generation. *Phys. Rev. Appl.* **16**(2), 024051 (2021).
19. Harrigan, M. P. *et al.* Quantum approximate optimization of non-planar graph problems on a planar superconducting processor. *Nat. Phys.* **17**(3), 332–336 (2021).
20. Liu, J. *et al.* Hybrid quantum-classical convolutional neural networks. *Sci. China Phys. Mech. Astron.* **64**(9), 290311 (2021).
21. Cong, I., Choi, S. & Lukin, M. D. Quantum convolutional neural networks. *Nat. Phys.* **15**(12), 1273–1278 (2019).
22. Smart, S. E., Hu, Z., Kais, S. *et al.* Relaxation of stationary states on a quantum computer yields a unique spectroscopic fingerprint of the computer's noise. *Commun. Phys.* **5**, 28. https://doi.org/10.1038/s42005-022-00803-8 (2022).
23. Kitaev, A. Y. Quantum computations: algorithms and error correction. *Russ. Math. Surv.* **52**, 1191 (1997).
24. Shor, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.* **26**(5), 1484–1509 (1997).
25. Harrow, A. W., Hassidim, A. & Lloyd, S. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.* **103**(15), 150502 (2009).
26. Peruzzo, A. *et al.* A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **5**(1), 4213 (2014).
27. Daskin, A. & Kais, S. Decomposition of unitary matrices for finding quantum circuits: application to molecular Hamiltonians. *J. Chem. Phys.* **134**(14), 144112 (2011).
28. Farhi, E., J. Goldstone, and S. Gutmann, *A Quantum Approximate Optimization Algorithm.* arXiv:1411.4028 [quant-ph] (2014).
29. Smart, S. E. & Mazziotti, D. A. Quantum solver of contracted eigenvalue equations for scalable molecular simulations on quantum computing devices. *Phys. Rev. Lett.* **126**(7), 070504 (2021).
30. Biamonte, J. *et al.* Quantum machine learning. *Nature* **549**(7671), 195–202 (2017).
31. Xia, R. & Kais, S. Quantum machine learning for electronic structure calculations. *Nat. Commun.* **9**(1), 4195 (2018).
32. Hu, Z., Xia, R. & Kais, S. A quantum algorithm for evolving open quantum dynamics on quantum computing devices. *Sci. Rep.* **10**(1), 3301 (2020).
33. Wang, H., Ashhab, S. & Nori, F. Quantum algorithm for simulating the dynamics of an open quantum system. *Phys. Rev. A* **83**(6), 062317 (2011).
34. Hu, Z. *et al.* A general quantum algorithm for open quantum dynamics demonstrated with the Fenna-Matthews-Olson complex dynamics. *Quantum* **6**, 726 (2022).

35. Schlimgen, A. W. *et al.* Quantum simulation of open quantum systems using a unitary decomposition of operators. *Phys. Rev. Lett.* **127**(27), 270503 (2021).
36. Zhang, Y. *et al.* Quantum simulation of the radical pair dynamics of the avian compass. *J. Phys. Chem. Lett.* **14**(3), 832–837 (2023).
37. Wang, Y., et al., *Simulation of open quantum system dynamics based on the generalized quantum master equation on quantum computing devices.* arXiv:2209.04956 (2022).
38. Hu, Z. & Kais, S. Characterization of quantum states based on creation complexity. *Adv. Quantum Technol.* **3**(9), 2000043 (2020).
39. Hu, Z., Kais, S. *The unitary dependence theory for characterizing quantum circuits and states.* Communications Physics, 2023. Accepted.
40. Nielsen, M. A., Chuang, I. L. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. 2011: Cambridge University Press. 708.
41. Hu, Z. & Kais, S. The quantum condition space. *Adv. Quantum Technol.* **5**, 2100158. https://doi.org/10.1002/qute.202100158 (2022).
42. Hu, Z., Kais, S. *The wave-particle duality of the qudit quantum space and the quantum wave gates.* arXiv:2207.05213 (2022).
43. Tilly, J., et al. *The Variational Quantum Eigensolver: A Review of Methods and Best Practices.* arXiv:2111.05176 [quant-ph] (2021).
44. Cerezo, M. *et al.* Variational quantum algorithms. *Nat. Rev. Phys.* **3**(9), 625–644 (2021).
45. Wang, Y., Li, G. & Wang, X. Variational quantum gibbs state preparation with a truncated taylor series. *Phys. Rev. Appl.* **16**(5), 054035 (2021).
46. Kandala, A. *et al.* Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* **549**(7671), 242–246 (2017).
47. Amy, M., Maslov, D. & Mosca, M. Polynomial-time T-depth optimization of Clifford+T Circuits via matroid partitioning. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **33**(10), 1476–1489 (2014).
48. Amy, M., Azimzadeh, P. & Mosca, M. On the controlled-NOT complexity of controlled-NOT–phase circuits. *Quantum Sci. Technol.* **4**(1), 015002 (2018).
49. IBM Quantum Experience http://www.research.ibm.com/quantum.
50. Sim, S., Johnson, P. D. & Aspuru-Guzik, A. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Adv. Quantum Technol.* **2**(12), 1900070 (2019).

Acknowledgements

Author contributions

S.K. and Z.H. designed the project. Z.H. developed the theory, carried out the calculations, and prepared the figures. Z.H. and S.K. wrote and reviewed the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-023-31980-3.

Correspondence and requests for materials should be addressed to S.K.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.