Embedding data into quantum circuits II

황용수 @ ETRI

2023.07.20

Outline

- Quantum State Preparation
- Quantum analog-digital conversion
- Ansatz

Reference

- Manuela Weigold et al., Encoding patterns for quantum algorithms, IET Quantum Communication, Vol. 2, Issue 4, pp.141—152 (2021)
- https://pennylane.ai
- M. Cerezo et al., Variational quantum algorithms, nature review physics 3, 625—644 (2021)
- Martin Plesch and Caslav Brukner, Quantum-state preparation with universal gate decomposition, PRA 83, 032302 (2011)
- Xiao-Ming Zhang et al., Quantum State Preparation with Optimal Circuit Depth: Implementations and Applications, PRL 129, 230504 (2022)
- Mikko Mottonen et al., Transformation of quantum states using uniformly controlled rotations, QIC 5, 6, 467—473 (2005)
- Kosuke Mitarai et al., Quantum analog-digital conversion, PRA 99, 012301 (2019)

Quantum State Preparation

Quantum State Preparation

- **Object**: A quantum state $|s\rangle$ has to be prepared on an empty qubit register.
  If the state preparation method is not known that exploits the structure of the state to prepare it efficiently, we have to use a **method for creating an arbitrary state** instead.

QSP #1: Uniformly Controlled Rotation

QIC 5, 6, 467—473 (2005)

Transformation of quantum states using uniformly controlled rotations

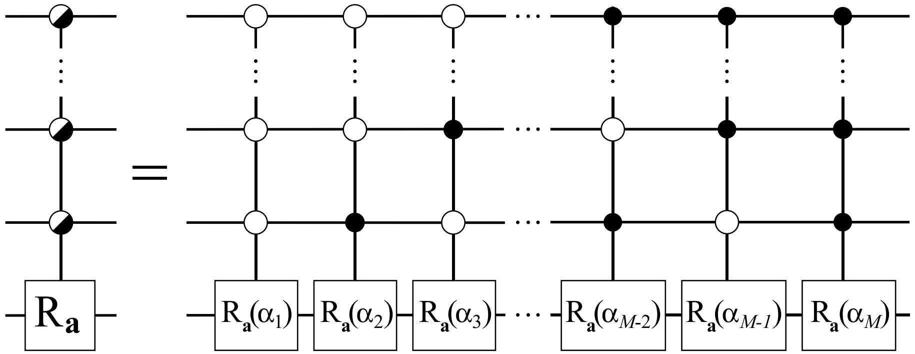
Mikko Möttönen,* Juha J. Vartiainen, Ville Bergholm, and Martti M. Salomaa
Materials Physics Laboratory, P.O. Box 2200,
FIN-02015 Helsinki University of Technology, Finland

(Dated: February 1, 2008)

We consider a unitary transformation which maps any given state of an n-qubit quantum register into another one. This transformation has applications in the initialization of a quantum computer, and also in some quantum algorithms. Employing uniformly controlled rotations, we present a quantum circuit of $2^{n+2} - 4n - 4$ CNOT gates and $2^{n+2} - 5$ one-qubit elementary rotations that effects the state transformation. The complexity of the circuit is noticeably lower than the previously published results. Moreover, we present an analytic expression for the rotation angles needed for the transformation.

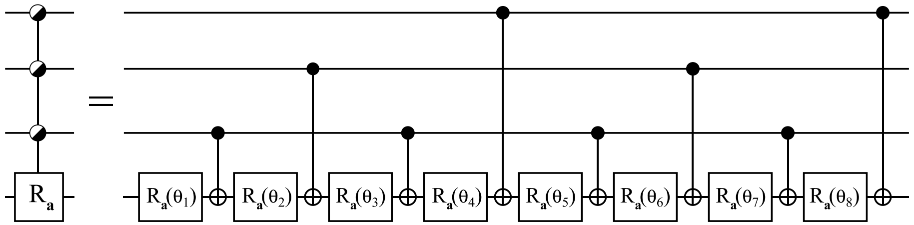
PACS numbers: 03.67.Lx, 03.65.Fd
Keywords: quantum computation, quantum state preparation

QSP #1: Uniformly Controlled Rotation



k-fold Uniformly Controlled Rotation Gate

FIG. 1: Definition of the k-fold uniformly controlled rotation [redacted]^k_m(**a**[redacted]) of qubit m about the axis **a**. The left hand side defines the gate symbol used for the uniformly controlled rotation. T[redacted] en[redacted]neration of the qubits is arbitrary with the exception that the target qubit is the m^{th} one. The black control bits stand[redacted]r va[redacted]ue 1 and the white for 0. Above, $M = 2^k$.



Single- and Two-qubit gate decomposition of *k-fold Uniformly Controlled Rotation Gate*

FIG. 2: Efficient gate decomposition for the uniformly controlled rotation $F^3_4(\mathbf{a}, \boldsymbol{\alpha})$. The relation of the angles $\{\theta_j\}$ to the angles $\{\alpha_j\}$ is shown in Eq. (3).

QSP #1: Uniformly Controlled Rotation

Our algorithm for transforming $|a\rangle = (|a_1|e^{i\omega_1}, |a_2|e^{i\omega_2}, \ldots, |a_N|e^{i\omega_N})^T$ into $|e_1\rangle$ works as follows:

- First we equalize the phases ω_i using a cascade of uniformly controlled z-rotations Ξ_z, rendering the vector real up to the global phase ϕ: $\Xi_z |a\rangle = e^{i\phi} |\hat{a}\rangle$.
- Then we rotate the real state vector $|\hat{a}\rangle$ into the direction of $|e_1\rangle$ using a similar sequence of uniformly controlled y-rotations Ξ_y, thus achieving our goal.

$$\Xi_y \Xi_z |a\rangle = \left(\prod_{j=1}^{n} F_j^{j-1}(\mathbf{y}, \boldsymbol{\alpha}^y_{n-j+1}) \otimes I_{2^{n-j}} \right) \left(\prod_{j=1}^{n} F_j^{j-1}(\mathbf{z}, \boldsymbol{\alpha}^z_{n-j+1}) \otimes I_{2^{n-j}} \right) |a\rangle = e^{i \sum_{j=1}^{N} \omega_j / N} |e_1\rangle . \quad (7)$$

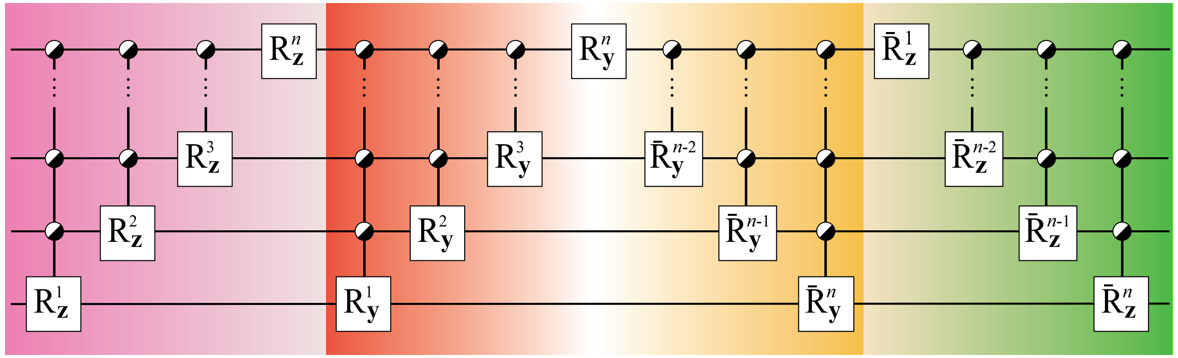


FIG. 3: Gate sequence for state preparation using uniformly controlled rotations. The rotation angles $\{\alpha^q_{j,k}\}$ for the uniformly controlled rotations are given in Eqs. (8) and (5).

QSP #2: Scmidt Decomposition

PRA 83, 032302 (2011)

PHYSICAL REVIEW A **83**, 032302 (2011)

Quantum-state preparation with universal gate decompositions

Martin Plesch[1,2,3] and Časlav Brukner[1,4]
[1] *Faculty of Physics, University of Vienna, Vienna, Austria*
[2] *Faculty of Informatics, Masaryk University, Brno, Czech Republic*
[3] *Institute of Physics, Slovak Academy of Sciences, Bratislava, Slovakia*
[4] *Institute of Quantum Optics and Quantum Information (IQOQI), Austrian Academy of Sciences, Vienna, Austria*
(Received 30 March 2010; published 3 March 2011)

In quantum computation every unitary operation can be decomposed into quantum circuits—a series of single-qubit rotations and a single type entangling two-qubit gates, such as controlled-NOT (CNOT) gates. Two measures are important when judging the complexity of the circuit: the total number of CNOT gates needed to implement it and the depth of the circuit, measured by the minimal number of computation steps needed to perform it. Here we give an explicit and simple quantum circuit scheme for preparation of arbitrary quantum states, which can directly utilize any decomposition scheme for arbitrary full quantum gates, thus connecting the two problems. Our circuit reduces the depth of the best currently known circuit by a factor of 2. It also reduces the total number of CNOT gates from 2^n to $\frac{23}{24}2^n$ in the leading order for even number of qubits. Specifically, the scheme allows us to decrease the upper bound from 11 CNOT gates to 9 and the depth from 11 to 5 steps for four qubits. Our results are expected to help in designing and building small-scale quantum circuits using present technologies.

Scmidt Decomposition

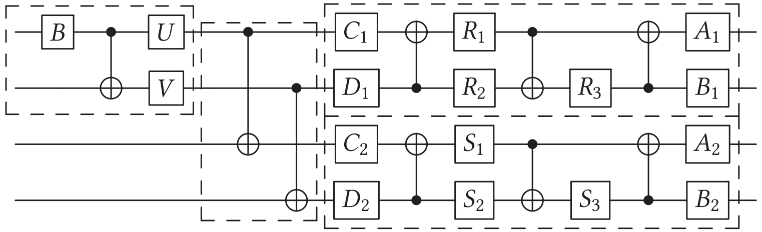


Fig. 56. Circuit for four qubit-state preparation. The four phases of the circuit are indicated in dashed boxes.

Circuit Optimization with circuit identity:
- Z with Ctrl qubit of CNOT
- X with Trgt qubit of CNOT

Evaluation: #CNOT or Circuit Depth

#CNOT: 2^n→**$23/24 \times 2^n$ for even n**
$115/96 \times 2^n$ for odd n

Depth: $23/48 \times 2^n$ for even n
$115/192 \times 2^n$ for odd n

QSP #2: Scmidt Decomposition

- **Schmidt Decomposition**

Theorem 2.2.1 — Schmidt decomposition. Consider quantum systems A and B with dimensions d_A, d_B respectively, and let $d = \min(d_A, d_B)$. Any pure bipartite state $|\Psi\rangle_{AB}$ has a Schmidt decomposition

$$|\Psi\rangle_{AB} = \sum_{i=1}^{d} \lambda_i |u_i\rangle_A |v_i\rangle_B, \tag{2.16}$$

where $\lambda_i \geq 0$ and $\{|u_i\rangle_A\}_i, \{|v_i\rangle_B\}_i$ are orthonormal vector sets. The coefficients λ_i are called the *Schmidt coefficients* and $|u_i\rangle_A, |v_i\rangle_B$ the *Schmidt vectors*.

QSP: Scmidt Decomposition

- To generate a circuit for the creation of a quantum state $|s\rangle$,
  we first need to express the state in terms of two subspaces V and W such that span Hilbert space.
- With the orthonormal basis $\{f_1, \cdots, f_k\} \in V$ and $\{g_1, \cdots, g_k\} \in W$,
  $|s\rangle$ is represented as a linear combination of these basis vectors: $|s\rangle = \sum_{i,j} b_{i,j} \cdot f_i \otimes g_j$.
- The Singular Value Decomposition (SVD) of the matrix $M = \{b_{ij}\}$ is computed as
  $M = (U_1 U_2) \begin{pmatrix} A \\ 0 \end{pmatrix} V^*$.
- The entries of the diagonal matrix A build the set $\{\alpha_1, \cdots, \alpha_m\}$, which defines the Schmidt decomposition of $|s\rangle$, $|s\rangle = \sum_{i=1}^{m} \alpha_i \cdot u_i \otimes v_i, \alpha_i \in \mathbb{R} \geq 0$, where $\sum_{i=1}^{m} \alpha_i = 1$, where $\alpha_1, \cdots, \alpha_m$ are Schmidt coefficients for the Schmidt basis $\{u_i\}, \{v_j\}$.

QSP: Scmidt Decomposition

$|0\rangle^{\otimes n}$

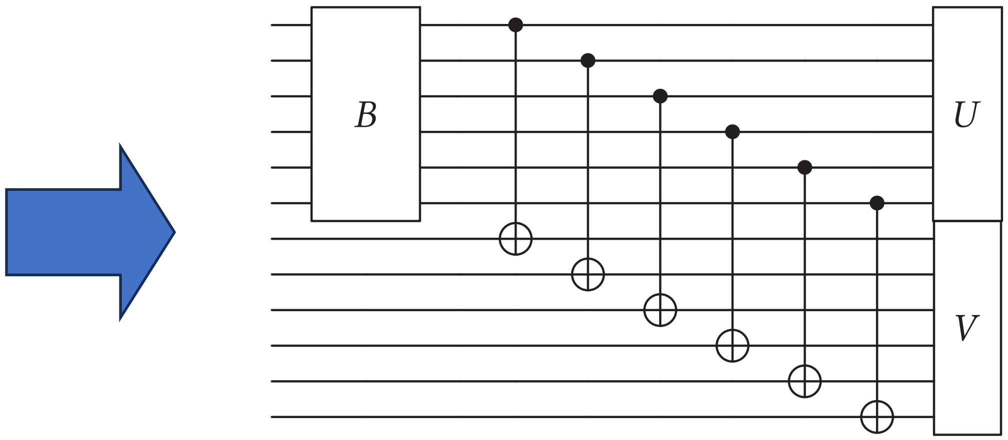


Fig. 53. Schmidt decomposition.

- $(U \otimes V)(CNOT_{n+1}^{1} \otimes \cdots \otimes CNOT_{n+n}^{n})(B \otimes I)|0\rangle^{\otimes n}$
- $= (U \otimes V)(CNOT_{n+1}^{1} \otimes \cdots \otimes CNOT_{n+n}^{n}) \sum_{i=1}^{2^n} b_{i1} |e_i\rangle|e_1\rangle$
- $= (U \otimes V) \sum_{i=1}^{2^n} b_{i1} |e_i\rangle|e_i\rangle$
- $= \sum_{i=1}^{2^n} b_{i1} (U|e_i\rangle)(V|e_i\rangle)$

QSP: Scmidt Decomposition

- To get the precise values of U, V, and B, we write $|s\rangle = \sum_{i,j=1}^{2^n} \alpha_{ij} |e_i\rangle|e_j\rangle$ for some constants $\alpha_{ij} \in \mathbb{C}$, and define $A = \{\alpha_{ij}\}$. Then, $\sum_{i,j=1}^{2^n} \alpha_{ij} |e_i\rangle|e_j\rangle = \sum_{i=1}^{2^n} b_{i1} (U|e_i\rangle)(V|e_i\rangle)$.
- Multiplying $\langle e_i|\langle e_j|$on the left part, then $\alpha_{ij} = \sum_{k=1}^{2^n} b_{k1} u_{ik} v_{jk}$, where $u_{ik} = \langle e_i|U|e_k\rangle$ and $v_{jk} = \langle e_j|V|e_k\rangle$. They respectively corresponds to U_{ik} and V_{jk}

QSP: Sparse Data Structure

- Circuit with Optimal Depth (at the cost of exponential qubits)

PHYSICAL REVIEW LETTERS **129,** 230504 (2022)

Quantum State Preparation with Optimal Circuit Depth: Implementations and Applications

Xiao-Ming Zhang, Tongyang Li, and Xiao Yuan*

Center on Frontiers of Computing Studies, Peking University, Beijing 100871, China and School of Computer Science, Peking University, Beijing 100871, China

(Received 30 January 2022; revised 1 August 2022; accepted 1 November 2022; published 30 November 2022)

Quantum state preparation is an important subroutine for quantum computing. We show that any n-qubit quantum state can be prepared with a $\Theta(n)$-depth circuit using only single- and two-qubit gates, although with a cost of an exponential amount of ancillary qubits. On the other hand, for sparse quantum states with $d \geqslant 2$ nonzero entries, we can reduce the circuit depth to $\Theta(\log(nd))$ with $O(nd \log d)$ ancillary qubits. The algorithm for sparse states is exponentially faster than best-known results and the number of ancillary qubits is nearly optimal and only increases polynomially with the system size. We discuss applications of the results in different quantum computing tasks, such as Hamiltonian simulation, solving linear systems of equations, and realizing quantum random access memories, and find cases with exponential reductions of the circuit depth for all these three tasks. In particular, using our algorithm, we find a family of linear system solving problems enjoying exponential speedups, even compared to the best-known quantum and classical dequantization algorithms.

DOI: 10.1103/PhysRevLett.129.230504

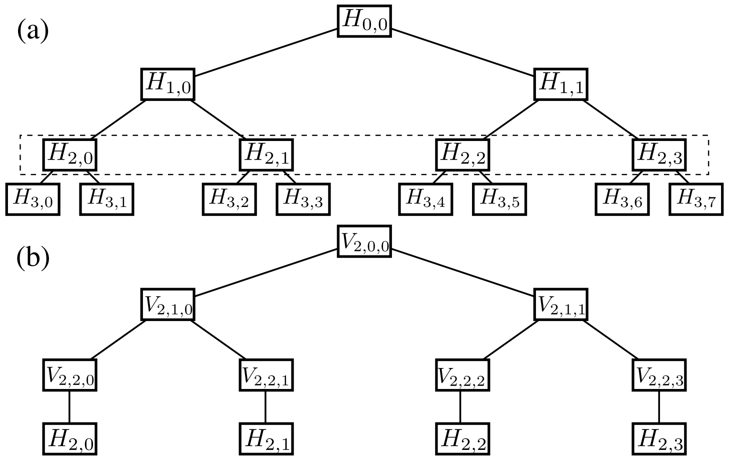


FIG. 1. (a) Layout of binary tree H. Each block represents a qubit. (b) Layout of binary tree V_2, which connects to the second layer of H with dashed box, i.e., H_2. Here, $V_{2,\mathrm{root}}$ is $V_{2,2,0}$. In (a) and (b), CNOT gates are only applied at qubit pairs connected by solid lines. (c) CNOT gate between two distant qubits (black circles) based on pre-shared Bell states (blue circles). $M_{x,z}$ and X, Z represent measurements and Pauli gates [40].

Quantum analog-digital conversion (PRA 99,012301)

Quantum anaog-digital conversion

PHYSICAL REVIEW A **99**, 012301 (2019)

Quantum analog-digital conversion

Kosuke Mitarai,[1,*] Masahiro Kitagawa,[1,2] and Keisuke Fujii[3,4,†]

[1]*Graduate School of Engineering Science, Osaka University, 1-3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan*
[2]*Quantum Information and Quantum Biology Division, Institute for Open and Transdisciplinary Research Initiatives, Osaka University, 1-3 Machikaneyama, Toyonaka, Osaka 560-8531, Japan*
[3]*Graduate School of Science, Kyoto University, Yoshida-Ushinomiya-cho, Sakyo-ku, Kyoto 606-8302, Japan*
[4]*JST, PRESTO, 4-1-8 Honcho, Kawaguchi, Saitama 332-0012, Japan*

(Received 21 June 2018; published 2 January 2019)

Many quantum algorithms, such as the Harrow-Hassidim-Lloyd (HHL) algorithm, depend on oracles that efficiently encode classical data into a quantum state. The encoding of the data can be categorized into two types: analog encoding, where the data are stored as amplitudes of a state, and digital encoding, where they are stored as qubit strings. The former has been utilized to process classical data in an exponentially large space of a quantum system, whereas the latter is required to perform arithmetics on a quantum computer. Quantum algorithms such as HHL achieve quantum speedups with a sophisticated use of these two encodings. In this work, we present algorithms that convert these two encodings to one another. While quantum digital-to-analog conversions have implicitly been used in existing quantum algorithms, we reformulate it and give a generalized protocol that works probabilistically. On the other hand, we propose a deterministic algorithm that performs a quantum analog-to-digital conversion. These algorithms can be utilized to realize high-level quantum algorithms such as a nonlinear transformation of amplitudes of a quantum state. As an example, we construct a "quantum amplitude perceptron," a quantum version of the neural network that hence has a possible application in the area of quantum machine learning.

DOI: 10.1103/PhysRevA.99.012301

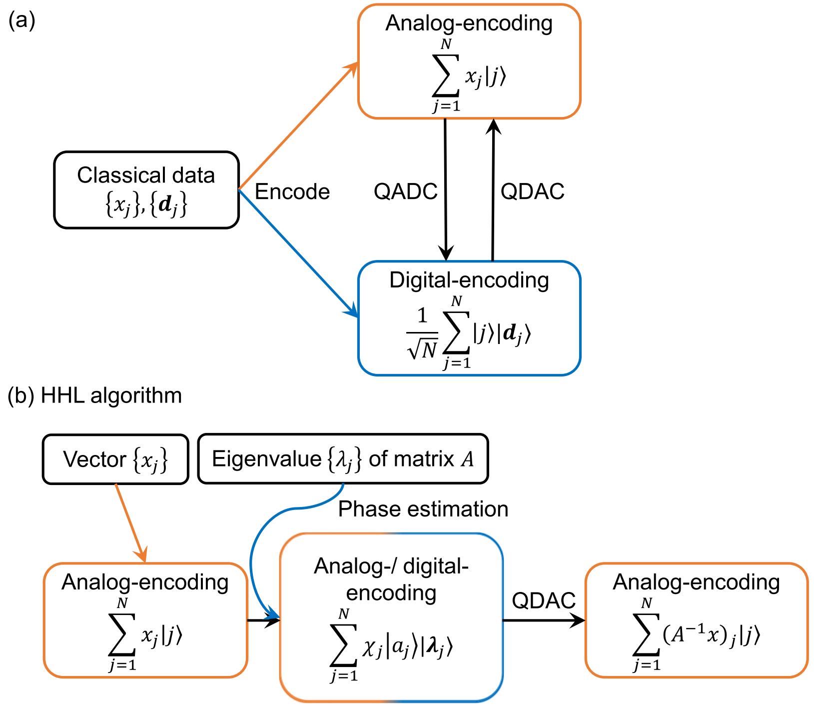


FIG. 1. (a) Schematic sketch of analog encoding and digital encoding. QDAC and QADC mediate these two encodings. (b) A brief flowchart of the HHL algorithm [4]. $\{|a_j\rangle\}$ denote eigenvectors of a Hermitian matrix A, each corresponding to eigenvalues $\{\lambda_j\}$. χ_j are complex numbers such that $\sum_{j=1}^{N} x_j|j\rangle = \sum_{j=1}^{N} \chi_j|a_j\rangle$.

Preliminaries (1/2)

- **Analog encoding**: Data is encoded into analog quantities, complex amplitude of a quantum state with unitary transformation $U_A(\{c_j\})$ as $U_A(\{c_j\})|0\rangle = \sum_j c_j|j\rangle$.
- **Digital encoding**: Data, m bits of binary string, is encoded into qubit strings with unitary transfomation $U_D(\{d_j\})$ as $U_D(\{d_j\})|j\rangle|0\rangle = |j\rangle|d_j\rangle$
- **Phase estimation**: Let U be a unitary operator acting on M-qubit Hilbert space with eigenstates $\{|\psi_j\rangle\}_{j=1}^{2^M}$ and corresponding eigenvalues $\{e^{2\pi i\phi_j}\}_{j=1}^{2^M}$, where $\phi_j \in [0,1)$. Let $\epsilon = 2^{-m}$ for positive integer m. There exists a quantum algorithm transforming $\sum_{j=1}^{2^M} a_j|\psi_j\rangle|0\rangle^{\otimes m} \longrightarrow$ $|\psi_{PE}\rangle \sum_{j=1}^{2^M} a_j|\psi_j\rangle|\tilde{\phi}_j\rangle$ such that $\left|\sum_{k=1}^{m} \tilde{\phi}_j^{(k)} - 2^{-k}\right| \leq \epsilon$ for all j with state fidelity at least 1-poly(ϵ)
- **Amplitude Amplification**: Suppose that we have a unitary operator U that acts on M-qubit Hilbert space as $U|0\rangle^{\otimes M} = \alpha|\psi\rangle|0\rangle + \beta|G\rangle|1\rangle$, where $|\psi\rangle$, $|G\rangle$ are arbitrary (M-1)-qubit states. Then, the prob. of getting $|\psi\rangle|0\rangle$ can be made close to 1 by $O(1/|\alpha|)$ applications of U

Preliminaries (2/2)

- **Quantum arithmetics:** Let a, b be m-bit strings. There exists a quantum algorithm that performs transformation $|a\rangle|b\rangle \rightarrow |a\rangle|a+b\rangle$ or $|a\rangle|b\rangle \rightarrow |a\rangle|ab\rangle$ with O(poly(m)) single- and two-qubit gates.
- **Quantum functions**: Some basic functions such as inverse, trigonometric functions, square root, and inverse trigonometric functions can be calculated to accuracy ϵ. That is, we can perform a transformation $|a\rangle|0\rangle \rightarrow |a\rangle\left|\tilde{f}(a)\right\rangle$ such that $\left|\tilde{f}(a) - f(a)\right| \leq \epsilon$ where $f(a)$ is the objective function, using O(poly($\log_2 1/\epsilon$)) quantum arithmetics

QDAC (Digital-to-Analog)

- **QDAC with ancilla.** There exists a quantum algorithm that performs m-bit QDAC using $O\big(poly(\log_2(1/\epsilon))\big)$ single- and two-qubit gates and one $U_D^\dagger$ where $\epsilon = 2^{-m}$ with $\sum_{j=1}^{N} d_j^2/N$.
- **Procedures**:
  1. (Compute $\varphi_j = \frac{2}{\pi}\cos^{-1} d_j$ by quantum arithmetic)
     $\frac{1}{\sqrt{N}}\sum_{j=1}^{N}|j\rangle|d_j\rangle|0\rangle^{\otimes m} \longrightarrow \frac{1}{\sqrt{N}}\sum_{j=1}^{N}|j\rangle|d_j\rangle|\varphi_j\rangle$, where $\varphi_j = \sum_{k=1}^{m}\varphi_j^{(k)}2^{-k}$.
  2. (Add ancilla $|0\rangle_a$ and Perform controlled rotation $R_y(\pi\varphi_j)$ on the ancilla,
     $$\frac{1}{\sqrt{N}}\sum_{j=1}^{N}|j\rangle|d_j\rangle|\varphi_j\rangle|0\rangle_a \longrightarrow \frac{1}{\sqrt{N}}\sum_{j=1}^{N}|j\rangle|d_j\rangle|\varphi_j\rangle\left(d_j|0\rangle_a + \sqrt{1-d_j^2}|1\rangle_a\right)$$
  3. (Measure ancilla in the computational basis)
     With prob. $\sum_{j=1}^{N} d_j^2/N$, we obtain $C\sum_{j=1}^{N} d_j|j\rangle|d_j\rangle|\varphi_j\rangle|0\rangle_a$ where $C = \sqrt{\left(\sum_{j=1}^{N} d_j^2\right)^{-1}}$
  4. (Uncompute φ_j and apply $U_D^\dagger$) $C\sum_{j=1}^{N} d_j|j\rangle$

QADC (Analog-to-Digital) (0/3)

- For the amplitudes $\{\mathrm{c_j}\}_{\mathrm{j=1}}^{\mathrm{N}}$ of a quantum state, there are three versions of QADC.
- ***Absolute* QADC**: Let $\tilde{r}_j$ denote the m-bit string $\tilde{r}_j^{(1)}, .., \tilde{r}_j^{(m)}$ that best approximates $\left|c_j\right|$ by $\sum_{k=1}^{m} \tilde{r}_j^{(k)} 2^{-k}$. An m-bit *absolute*-QADC operation transforms analog-encoded state $\sum_j^N c_j |j\rangle |0\rangle^{\otimes m}$ to $\frac{1}{\sqrt{N}} \sum_j^N |j\rangle \left|\tilde{r}_j\right\rangle$.
- ***Real* QADC**: Let $\tilde{x}_j$ denote the m-bit string $\tilde{x}_j^{(1)}, .., \tilde{x}_j^{(m)}$ that best approximates the real part of c_j by $\sum_{k=1}^{m} \tilde{x}_j^{(k)} 2^{-k}$. An m-bit *real*-QADC operation transforms analog-encoded state $\sum_j^N c_j |j\rangle |0\rangle^{\otimes m}$ to $\frac{1}{\sqrt{N}} \sum_j^N |j\rangle \left|\tilde{x}_j\right\rangle$.
- ***Imaginary* QADC**: Let $\tilde{y}_j$ denote the m-bit string $\tilde{y}_j^{(1)}, .., \tilde{y}_j^{(m)}$ that best approximates the imaginary part of c_j by $\sum_{k=1}^{m} \tilde{y}_j^{(k)} 2^{-k}$. An m-bit *imaginary*-QADC operation transforms analog-encoded state $\sum_j^N c_j |j\rangle |0\rangle^{\otimes m}$ to $\frac{1}{\sqrt{N}} \sum_j^N |j\rangle \left|\tilde{y}_j\right\rangle$.

QADC (Analog-to-Digital) (1/3)

- ***Absolute* QADC.** There exists an m-bit absolute-QADC algorithm that runs using $O(1/\epsilon)$ controlled U_A gates and $O\left(\left((\log_2 N)\right)^2/\epsilon\right)$ single- and two-qubit gates with output state fidelity $1 - O\big(poly(\epsilon)\big)$ where $\epsilon = 2^{-m}$.
- **Procedures:**
  1. (Prepare address qubits) $\frac{1}{\sqrt{N}}\sum_{k=1}^{N}|k\rangle_{ad}$
  2. (CNOT from address qubits to ancilla qubits A)
     $\frac{1}{\sqrt{N}}\sum_{k=1}^{N}|k\rangle_{ad}\,|k\rangle_{A}$
  3. (Prepare analog-encoded state in data qubits)
     $\sum_{j=1} c_j|j\rangle_{data}$
  4. (SWAP test with another ancilla B)
     $\equiv \frac{1}{\sqrt{N}}\sum_k|k\rangle_{ad}|\Psi_k\rangle_{data,A,B}$

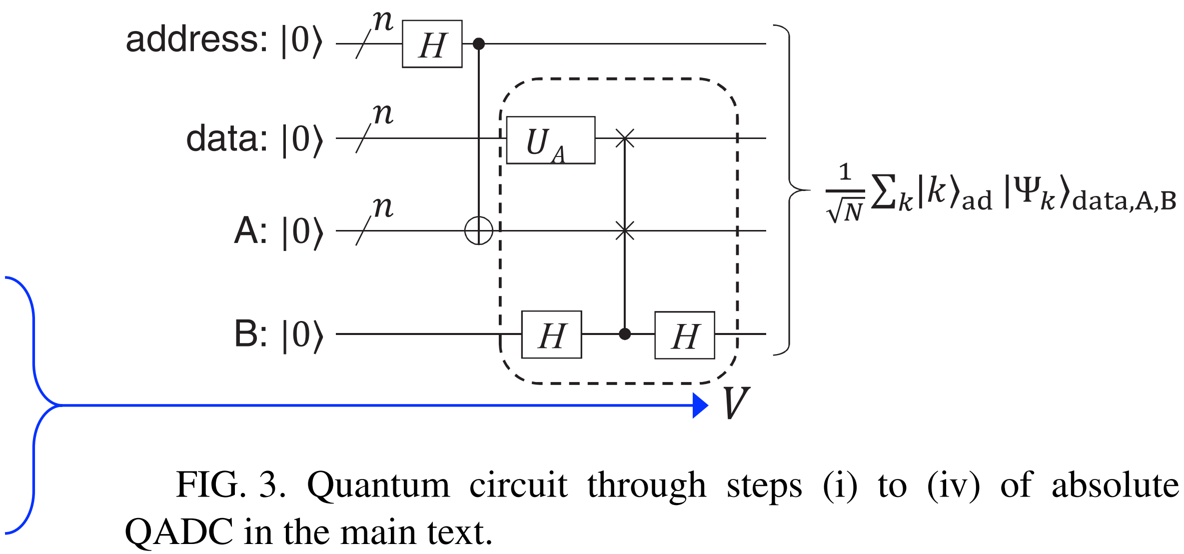


FIG. 3. Quantum circuit through steps (i) to (iv) of absolute QADC in the main text.

V: It extracts an absolute value r_k of amplitude c_k

QADC (Analog-to-Digital) (2/3)

- **Procedures (cont'd):**
  5. (Construct a gate G)
     $G = V \cdot CNOT_{ad\to A} \cdot S_0 \cdot CNOT_{ad\to A} \cdot V^\dagger \cdot Z_B$,
     where $S_0 = I - 2(|0\rangle\langle 0|)_{data,A,B}$ and Z_B is Pauli-Z on B

     $G\frac{1}{\sqrt{N}}\sum_k |k\rangle_{ad}|\Psi_k\rangle_{data,A,B} = \frac{1}{\sqrt{N}}\sum_k |k\rangle_{ad}\left(G_k|\Psi_k\rangle_{data,A,B}\right)$,
     where $G_k = VS_kV^\dagger Z_B$ and $S_k = I - 2(|0\rangle\langle 0|)_{data,B}\otimes(|k\rangle\langle k|)_A$

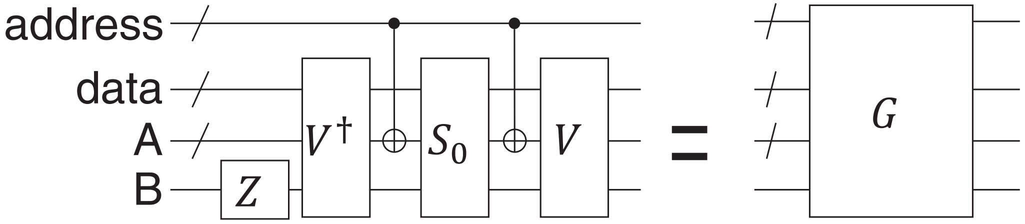


FIG. 4. Definition of gate G in absolute QADC.

  6. (Introduce Register qubits and Phase estimation of G)

$$\equiv \frac{1}{\sqrt{N}}\sum_k |k\rangle_{ad}\left|\Psi_{k,AE}\right\rangle_{reg',data,A,B}$$

where $\left|\Psi_{k,AE}\right\rangle_{reg',data,A,B} =$

$$\frac{1}{\sqrt{2}}\left(|\theta_k\rangle_{reg'}|\Psi_{k+}\rangle_{data,A,B} + |1-\theta_k\rangle_{reg'}|\Psi_{k-}\rangle_{data,A,B}\right)$$

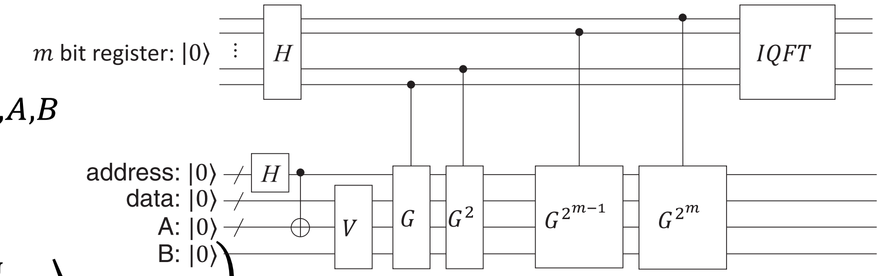


FIG. 5. Step (vi) of the absolute-QADC algorithm. The phase estimation is performed to encode the analog-encoded value x_j into qubit bit strings. IQFT: inverse quantum Fourier transformation [25].

QADC (Analog-to-Digital) (3/3)

- **Procedures (cont'd):**
  7. (On another register, calculate $r_k = \sqrt{2(\sin \pi\theta_k)^2 - 1}$)
  $$\frac{1}{\sqrt{N}}\sum_k |k\rangle_{ad} |\tilde{r}_k\rangle_{reg} \left|\Psi_{k,AE}\right\rangle_{reg',data,A,B}$$
  8. (Uncompute the data, A, B, and reg')
  $$\frac{1}{\sqrt{N}}\sum_k |k\rangle_{ad} |\tilde{r}_k\rangle_{reg} |0\rangle_{reg',data,A,B}$$

Digital-encoded state !!

Real- and *Imaginary*-QADC work similarly, but with little modifications such as taking Hadamard test.

Ansatz

Glimpse of VQA

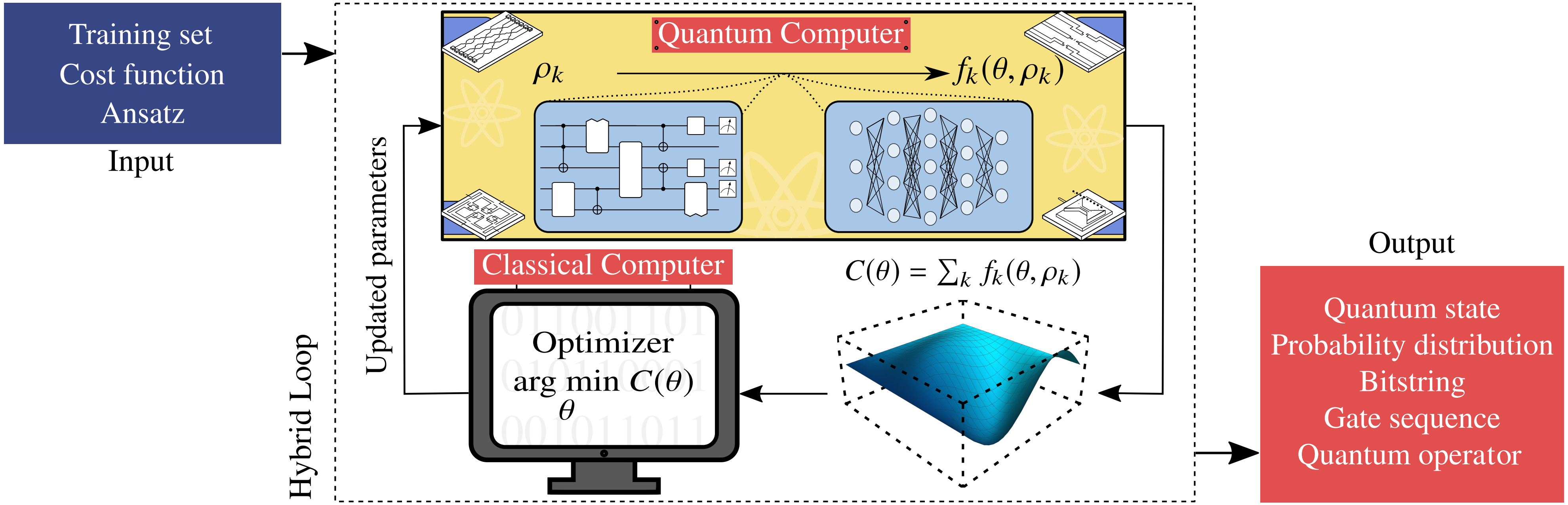


FIG. 1. **Schematic diagram of a Variational Quantum Algorithm (VQA)**. The inputs to a VQA are: a cost function $C(\boldsymbol{\theta})$, with $\boldsymbol{\theta}$ a set of parameters that encodes the solution to the problem, an ansatz whose parameters are trained to minimize the cost, and (possibly) a set of training data $\{\rho_k\}$ used during the optimization. Here, the cost can often be expressed in the form in Eq. (3), for some set of functions $\{f_k\}$. Also, the ansatz is shown as a parameterized quantum circuit (on the left), which is analogous to a neural network (also shown schematically on the right). At each iteration of the loop one uses a quantum computer to efficiently estimate the cost (or its gradients). This information is fed into a classical computer that leverages the power of optimizers to navigate the cost landscape $C(\boldsymbol{\theta})$ and solve the optimization problem in Eq. (1). Once a termination condition is met, the VQA outputs an estimate of the solution to the problem. The form of the output depends on the precise task at hand. The red box indicates some of the most common types of outputs.

Glimpse of VQA

- **Variational method** in quantum theory is a method for **finding low energy states of a quantum system**. The rough idea of the method is that one defines a **trial wave function** (sometimes called an **ansatz**) as a function of some parameters, and then one finds the values of these parameters that minimize the expectation value of the energy with respect to these parameters.
- The minimized ansatz is then an approximation to the lowest energy eigenstate, and the expectation value serves as an upper bound on the energy of the ground state.

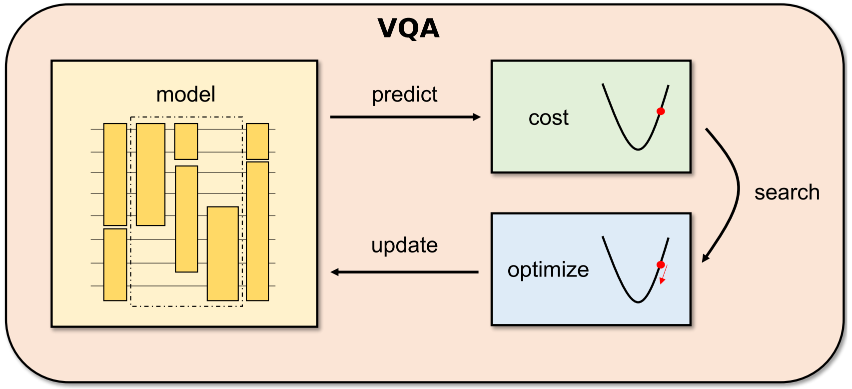


Fig. 1. Schematic of variational quantum algorithm, the model is designed based on quantum gates on quantum computer, and the optimization progress is on classical computer, the classical computer optimizes and updates the parameters in the trainable layer of the model.

<source: arXiv:2212.0491>

Ansatz

- In the context of variational circuits, an ansatz describes **a subroutine consisting of a sequence of gates applied to specific wires (qubits)**. Similar to the architecture of a neural network, this only defines a base structure, while **the types of gates and/or their free parameters can be optimized by the variational proceudre**.
- Generically speaking the form of the ansatz dictates what the parameters are, and hence, how they can be trained to minimize the cost.

- **Problem-inspired ansatz**: Ansatz tailored to the information about the problem.
- **Problem-agnostic ansatz**: They can be used even when no relevant information about the problem is available.

Ansatz

- Parameters are encoded into a unitary $U(\theta)$ that is applied to the input state,
  $U(\theta) = U_L(\theta_L)\ \cdots U_1(\theta_1)$ with
  $U_l(\theta_l) = \prod_m e^{-i\theta_m H_m} W_m$.
  Here W_m is an unparametrized unitary and H_m is a Hermitian operator.

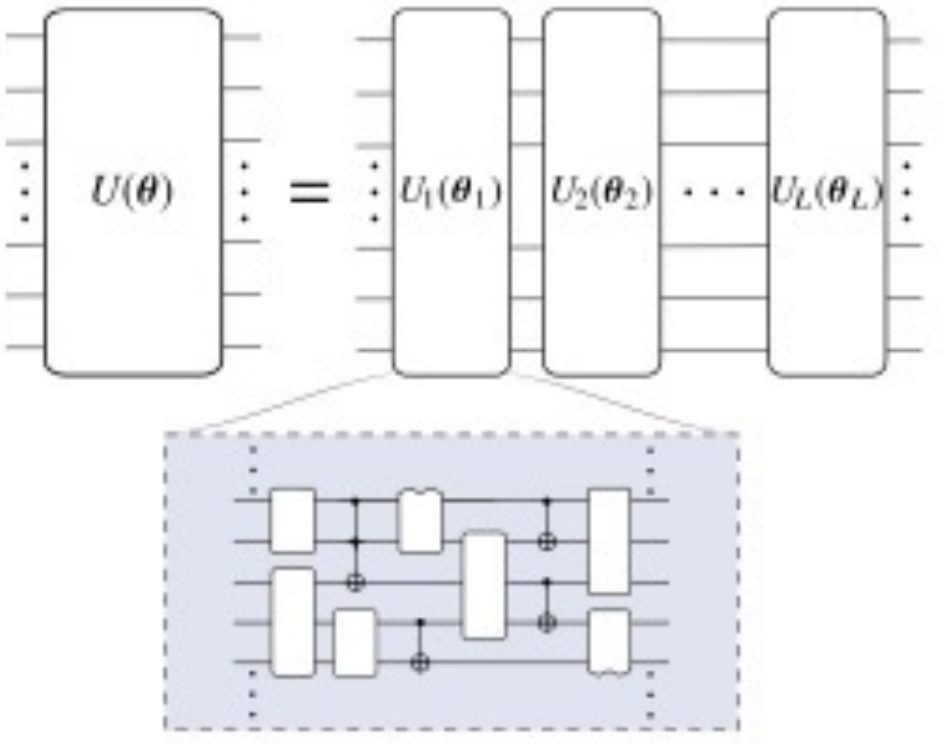
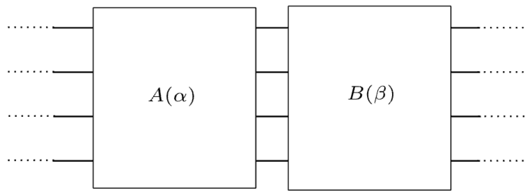


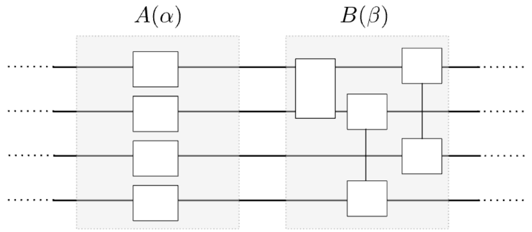
FIG. 2. **Schematic diagram of an ansatz.** The unitary $U(\boldsymbol{\theta})$, with $\boldsymbol{\theta}$ a set of parameters, can be expressed as a product of L unitaries $U_l(\boldsymbol{\theta}_l)$ sequentially acting on an input state. As indicated, each unitary $U_l(\boldsymbol{\theta}_l)$ can in turn be decomposed into a sequence of parametrized and unparametrized gates.

Types of Ansatz #1: Layered Gate Ansatz

- A layer is a sequence of gates that is repeated. The number of repetitions of a layer forms a hyperparameter of the variational circuit. The layer can be decomposed into two overall unitaries A and B.

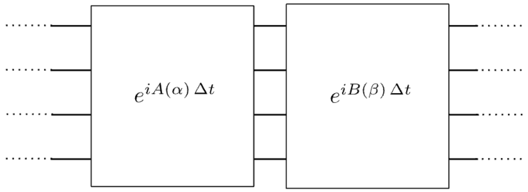


- Block A contains single-qubit gates applied to every subsystem or wire (qubits). Block B consists of both single-wire gates as well as entangling gates



Types of Ansatz #2: Alternating operator Ansatz

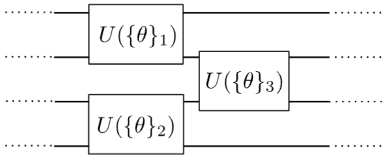
- We use layers of two blocks, but the difference is that here we apply the unitaries representing the Hamiltonians A and B which are evolved for a short time Δt.



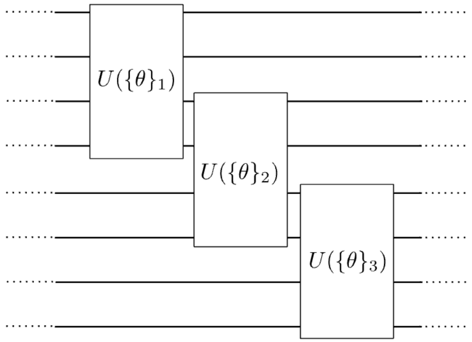
- The idea of this ansatz is based on analogies to adiabatic quantum computing, in which the system starts in the ground state of A and adiabatically evolves to the ground state of B. Quickly alternating applications of A and B for very short time Δt can be used as a heuristic to approximate this evolution

Types of Ansatz #3: Tensor network ansatz

- Gate sequence inspired by tensor networks. The simplest one is a tree architecture that consecutively entangles subsets of qubits.



- Another tensor network is based on matrix product states. The circuit unitaries can be decomposed in different ways, and their size corresponds to the “bond dimension” of the matrix product state – the higher the bond dimension, the more complex the circuit ansatz.



Types of Ansatz #4: Hardware efficient ansatz

- The hardware efficient ansatz is a generic name used for ansatzes that are aimed at reducing the circuit depth needed to implement $U(\theta)$ when using a given quantum hardware.
- One uses unitaries W_m and $e^{-i\theta_m H_m}$ that are taken from **a gate set determined from the connectivity and interactions specific to a quantum hardware** which avoids the circuit depth overhead arising from translating an arbitrary unitary into the sequence of native gates

Ansatz Expressibility

- Given the wide range of ansatzes one can use, a relevant question is whether a given architecture can prepare a target state by optimizing its parameters.
- Two ways to judge the quality of an ansatz: **expressibility** and **entangling capability**
- An ansatz is *expressible* if the circuit can be used to uniformly explore the entire space of a quantum state. One way to quantify the **expressibility** of an ansatz $U(\theta)$ is to compare the distribution of states obtained from $U(\theta)$ to the maximally expressive uniform (Haar) distribution of states U_{Haar}, $A^{(t)}(U) = \int dU_{Haar} U_{Haar}^{\otimes t} |0\rangle\langle 0| \left(U_{Haar}^{\dagger}\right)^{\otimes t} - \int dU U^{\otimes t} |0\rangle\langle 0| \left(U^{\dagger}\right)^{\otimes t}$.
- A measure of **entangling capability** for ansatz quantifies the average entanglement of states produced from randomly sampling the circuit parameter θ.

Quantum architecture search (QAS)

ARTICLE OPEN

Check for updates

Quantum circuit architecture search for variational quantum algorithms

Yuxuan Du[1,2]✉, Tao Huang[2,6], Shan You[3], Min-Hsiu Hsieh[4,5]✉ and Dacheng Tao[1,2]✉

Variational quantum algorithms (VQAs) are expected to be a path to quantum advantages on noisy intermediate-scale quantum devices. However, both empirical and theoretical results exhibit that the deployed ansatz heavily affects the performance of VQAs such that an ansatz with a larger number of quantum gates enables a stronger expressivity, while the accumulated noise may render a poor trainability. To maximally improve the robustness and trainability of VQAs, here we devise a resource and runtime efficient scheme termed quantum architecture search (QAS). In particular, given a learning task, QAS automatically seeks a near-optimal ansatz (i.e., circuit architecture) to balance benefits and side-effects brought by adding more noisy quantum gates to achieve a good performance. We implement QAS on both the numerical simulator and real quantum hardware, via the IBM cloud, to accomplish data classification and quantum chemistry tasks. In the problems studied, numerical and experimental results show that QAS cannot only alleviate the influence of quantum noise and barren plateaus but also outperforms VQAs with pre-selected ansatze.

npj Quantum Information (2022)8:62 ; https://doi.org/10.1038/s41534-022-00570-y

Quantum architecture search (QAS)

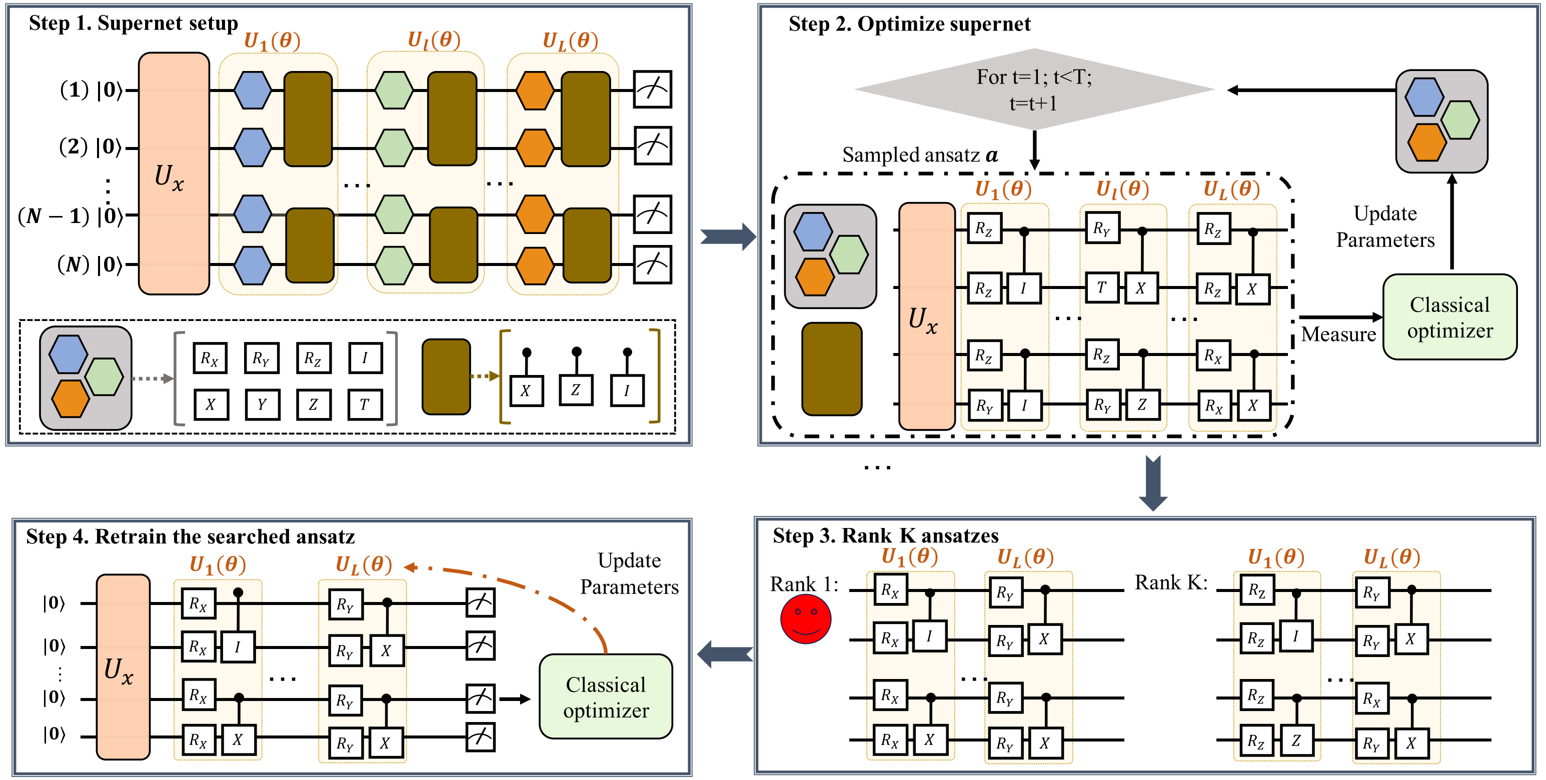
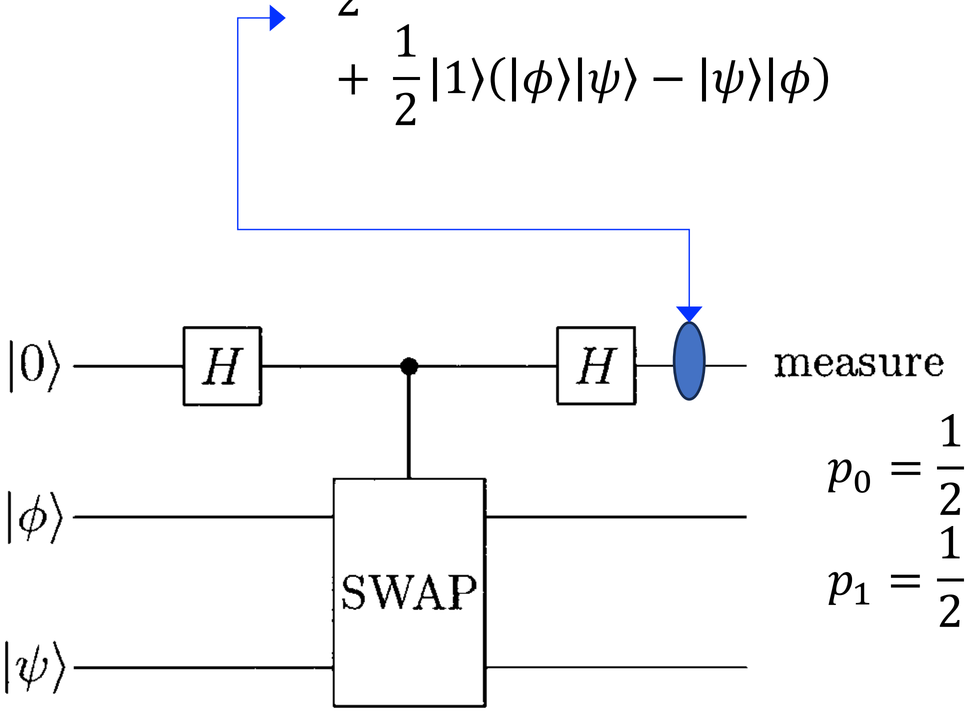


Fig. 1 Paradigm of the quantum architecture search scheme (QAS). In Step 1, QAS sets up supernet $\mathcal{A}$, which defines the ansatze pool $\mathcal{S}$ to be searched and parameterizes each ansatz in $\mathcal{S}$ via the specified weight sharing strategy. All possible single-qubit gates are highlighted by hexagons and two-qubit gates are highlighted by the brown rectangle. The unitary U_x refers to the data encoding layer. In Step 2, QAS optimizes the trainable parameters for all candidate ansatzes. Given the specified learning task $\mathcal{L}$, QAS iteratively samples an ansatz $\boldsymbol{a}^{(t)} \in \mathcal{S}$ from $\mathcal{A}$ and optimizes its trainable parameters to minimize $\mathcal{L}$. $\mathcal{A}$ correlates parameters among different ansatzes via weight sharing strategy. After T iterations, QAS moves to Step 3 and exploits the trained parameters $\boldsymbol{\theta}^{(T)}$ and the predefined $\mathcal{L}$ to compare the performance among K ansatze. The ansatz with the best performance is selected as the output, indicated by a red smiley face. Last, in Step 4, QAS utilizes the searched ansatz and the parameters $\boldsymbol{\theta}^{(T)}$ to retrain the quantum solver with few iterations.

Appendix

SWAP Test

$$\frac{1}{2}|0\rangle(|\phi\rangle|\psi\rangle + |\psi\rangle|\phi\rangle)$$
$$+ \frac{1}{2}|1\rangle(|\phi\rangle|\psi\rangle - |\psi\rangle|\phi\rangle)$$



$$p_0 = \frac{1}{2}(1 + |\langle\phi|\psi\rangle|^2)$$
$$p_1 = \frac{1}{2}(1 - |\langle\phi|\psi\rangle|^2)$$

FIG. 1. Quantum circuit to test if $|\phi\rangle = |\psi\rangle$ or $|\langle\phi|\psi\rangle| \leq \delta$.

VOLUME 87, NUMBER 16 PHYSICAL REVIEW LETTERS 15 OCTOBER 2001

Quantum Fingerprinting

Harry Buhrman,[1,*] Richard Cleve,[2,†] John Watrous,[2,‡] and Ronald de Wolf[1,§]
[1]*CWI, P.O. Box 94709, Amsterdam, The Netherlands*
and University of Amsterdam, Amsterdam, The Netherlands
[2]*Department of Computer Science, University of Calgary, Calgary, Alberta, Canada T2N 1N4*
(Received 19 April 2001; published 26 September 2001)

Classical fingerprinting associates with each string a shorter string (its *fingerprint*), such that any two distinct strings can be distinguished with small error by comparing their fingerprints alone. The fingerprints cannot be made exponentially smaller than the original strings unless the parties preparing the fingerprints have access to correlated random sources. We show that fingerprints consisting of *quantum* information *can* be made exponentially smaller than the original strings without any correlations or entanglement between the parties. This implies an exponential quantum/classical gap for the equality problem in the simultaneous message passing model of communication complexity.