Quantum Computing Method for Solving Electromagnetic Problems Based on the Finite Element Method

Jianan Zhang, *Member, IEEE*, Feng Feng, *Senior Member, IEEE*, and Qi-Jun Zhang, *Fellow, IEEE*

***Abstract*— Applying quantum computation to solve electromagnetic (EM) problems is still at an early age. Recently, an initial study on applying quantum computation to solve the finite element method (FEM) equations in the EM domain has been made. This article makes a further development beyond the initial study. Specifically, we first develop an approach to systematically prepare the quantum state to represent the right-hand side (RHS) vector of the finite element equation in EM problems. Then, to reduce the number of gates needed in the quantum state preparation process, we propose a quantum-gate-reduction method, which explores the fact that the FEM cells in the input port are a small portion of the total cells in the 3-D EM structure and that the number of gates needed for state preparation depends on the RHS vector's sparsity pattern. Based on the proposed quantum-gate-reduction method, we further derive the upper and lower bounds analytically for the number of gates needed in the quantum state preparation circuit. Furthermore, to deal with the large condition number of the finite element matrix in EM, we leverage a matrix preconditioner to modify the original linear equations, so as to reduce the number of qubits required in using quantum computation to solve EM problems. Two EM examples are used to illustrate how the proposed quantum computing method can be used to find solutions to EM problems.**

***Index Terms*— Electromagnetic (EM), finite element method (FEM), Harrow–Hassidim–Lloyd (HHL), quantum computing.**

I. Introduction

BEING one of the most thriving topics for solving large-scale problems, quantum computing (or quantum computation) has attracted increasing attention from the computational community in recent years [1], [2], [3]. In certain cases, quantum algorithms provide exponential speedups over their classical counterparts running on classical computers. Quantum computing opens many new opportunities for simulating complex physical systems, one of which is electromagnetic (EM) structures. Despite the intensive research on quantum computing in recent years, the study of quantum simulation of EM structures is sparse in the literature. In [4] and [5], a quantum computing algorithm based on the transmission line matrix (TLM) method has been presented for EM simulation. In [1], the possibility of applying quantum computing to solve the finite element method (FEM) equations in EM [6] has been investigated. Later on, in [7], quantum computation has been applied to solve matrix equations arising from the application of the method of moments (MoM) to solving EM problems.

The initial study in [1] was conducted based on the Harrow–Hassidim–Lloyd (HHL) algorithm [8], developed recently in the quantum computation community for solving the quantum version of linear equations [9]. In theory, the HHL algorithm can solve linear equations exponentially faster than classical computation. In the last decade, HHL has promoted the development of a plethora of approaches to solving mathematical tasks [10], [11], [12], [13], [14]. The study of [1] reformulates the finite element equation into a new quantum computation format, which opens the opportunity for solving finite element equations of EM problems by the HHL algorithm. Recently, an application of the sparse approximate inverse (SPAI) preconditioner in conjuction with the HHL algorithm to compute the EM scattering cross section of an arbitrary target has been investigated in [15]. However, how to efficiently set up the state preparation quantum circuit for the right-hand side (RHS) vector in the FEM for EM problems remains unexplored.

Efficient quantum state preparation is an indispensible component for applying quantum computation to solving EM problems. In the last decade, several methods have been reported to construct the quantum circuit for quantum state preparation [16], [17]. However, these studies focus on the preparation of the quantum state for a full vector. In the finite element linear equations for EM problems, the RHS vector is evaluated based on the EM excitation in the input port of the 3-D EM structure. Our investigations show that the gate count will increase proportionally with the vector size if we prepare the quantum state for the RHS vector (i.e., EM excitation vector) as a full vector, which means that the exponential speedup of HHL will vanish in the very first step. To maintain

Manuscript received 17 February 2023; revised 14 May 2023 and 22 June 2023; accepted 30 June 2023. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under Grant RGPIN-2017-06420, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20220808, in part by the National Natural Science Foundation of China under Grant 62101382, and in part by the Startup Research Fund of Southeast University under Grant RF1028623061. *(Corresponding author: Qi-Jun Zhang.)*

Jianan Zhang is with the State Key Laboratory of Millimeter Waves, Southeast University, Nanjing 210096, China (e-mail: jiananzhang@seu.edu.cn).

Feng Feng is with the School of Microelectronics, Tianjin University, Tianjin 300072, China (e-mail: ff@tju.edu.cn).

Qi-Jun Zhang is with the Department of Electronics, Carleton University, Ottawa, ON K1S 5B6, Canada (e-mail: qjz@doe.carleton.ca).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMTT.2023.3297406.

Digital Object Identifier 10.1109/TMTT.2023.3297406

HHL's exponential speedup when being applied to solve EM problems, it is crucial to develop a quantum circuit in which the gate count increases (no worse than) logarithmically with respect to vector size, which remains an open topic in the literature.

In this article, we make a further development beyond the work of [1]. In particular, we develop an approach to systematically prepare the quantum state to represent the EM excitation vector of the finite element equation in EM. To improve the quantum state preparation efficiency, we further propose a quantum-gate-reduction method to reduce the number of gates needed in the state preparation of the EM excitation vector. The proposed method explores the fact that the EM excitation vector in EM problems is always sparse and that the number of gates needed for state preparation depends on the EM excitation vector's sparsity pattern. Based on the proposed quantum-gate-reduction method, we derive the upper and lower bounds analytically for the number of gates needed in the state preparation process to theoretically demonstrate that the gate count increases logarithmically with respect to the EM excitation vector size. Because of this, the HHL's exponential speedup is maintained even if the state preparation procedure for the EM excitation vector is included. Furthermore, to deal with the high condition number of the finite element matrix, we leverage the SPAI preconditioner to modify the linear equation in the FEM formulation of EM problems. In this way, we reduce the number of qubits required in applying quantum computation to solving EM problems. Two EM examples are presented to demonstrate how the proposed quantum computing method can be applied to find solutions to EM problems.

II. Preliminaries

A. Qubits and Quantum States

The quantum bit (or qubit for short) is the fundamental unit of information in quantum computing. A qubit has a state, called quantum state. In quantum mechanics, we represent states as vectors. Two possible states for a qubit are $|0\rangle$ and $|1\rangle$, which are usually identified as column vectors

$$|0\rangle \triangleq \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{1}$$

and

$$|1\rangle \triangleq \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{2}$$

respectively, where notation "$|\rangle$" is the standard notation (called *Dirac* notation) to describe the state of qubits in quantum mechanics [2]. For example, a column vector and a row vector are written in *Dirac* notation as $|u\rangle$ and $\langle u|$, respectively. The inner and outer products of two vectors $|u\rangle$ and $|v\rangle$ are denoted by $\langle u|v\rangle$ and $|u\rangle\langle v|$, respectively.

A qubit can be in a state other than $|0\rangle$ and $|1\rangle$, by forming linear combinations of states (called superpositions) as follows:

$$|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle \triangleq \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \tag{3}$$

where α and β are the complex coefficients, with $|\alpha|^2 + |\beta|^2 = 1$. When we examine/observe a qubit, we get the result 0 or 1 with probabilities $|\alpha|^2$ or $|\beta|^2$, respectively. A two-qubit system has four computational basis states denoted as $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$. The quantum state of two qubits can be described by the state vector

$$|\varphi\rangle = \alpha_0|00\rangle + \alpha_1|01\rangle + \alpha_2|10\rangle + \alpha_3|11\rangle \tag{4}$$

which is sometimes, for notational convenience, written as follows:

$$|\varphi\rangle = \sum_{i=0}^{3} \alpha_i |i\rangle \tag{5}$$

where $|i\rangle$ is a vector of size 4, $i = 0, 1, 2, 3$, and $\sum_i |\alpha_i|^2 = 1$. One may note that the notations in the set of basis vectors in (4) are the binary representations of the notations in the set of basis vectors in (5). Let $N = 2^n$; then, an n-qubit quantum state can be expressed in a similar manner as follows:

$$|\psi\rangle = \sum_{i=0}^{N-1} \alpha_i |i\rangle \tag{6}$$

where $|i\rangle$ is a vector of size N and $i = 0, 1, \ldots, N-1$. Notably, the total number of coefficients to describe an n-qubit quantum state is N, and the coefficients satisfy

$$\sum_{i=0}^{N-1} |\alpha_i|^2 = 1. \tag{7}$$

We can represent the quantum state $|\psi\rangle$ with a vector as shown below

$$|\psi\rangle \triangleq \begin{bmatrix} |\alpha_0|e^{i\theta_0} \\ |\alpha_1|e^{i\theta_1} \\ \vdots \\ |\alpha_{N-1}|e^{i\theta_{N-1}} \end{bmatrix}. \tag{8}$$

It should be noticeable that in quantum computation, it is the relative difference between the angles (or phases) for different coefficients that is important. We refer to a phase $e^{i\theta_g}$ as a *global phase* if it is applied to the entire state, not just one term in the superposition. A global phase change has no impact on quantum measurements, which means that the measurement statistics obtained by measuring two states with a global phase difference are exactly identical. In such cases, we say that the two states are *equal, up to global phase* [2].

B. Quantum Operators and Quantum Circuits

In quantum computation, the whole computation process is described by quantum states evolving under the action of a series of quantum gates/operators. Each quantum operator (gate) is mathematically represented by a square unitary matrix of size $N \times N$, where $N = 2^n$ if one applies the operator to n qubits. Taking the one-qubit Hadamard operator as an example, the matrix to represent the Hadamard operator is defined as follows:

$$H \triangleq \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{9}$$

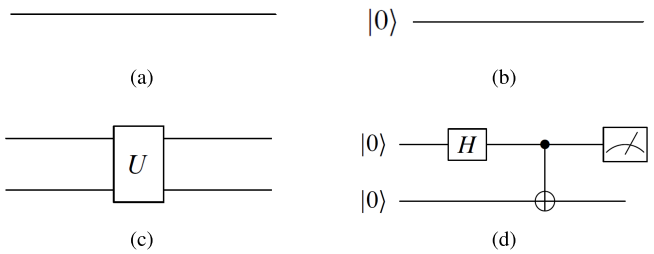


Fig. 1. (a) Circuit diagram of a quantum circuit wire with no operator on it. (b) Circuit diagram of the initial prepared state. (c) General circuit diagram of binary operators, where the letter U represents the specific operator. (d) Example of a two-qubit quantum circuit diagram [2].

Another two important operators, which will be used in Section III and Appendixes B and C, are the elementary rotation gates $R_y(\theta)$ and $R_z(\theta)$. Mathematically, the $R_y(\theta)$ gate can be represented by the following 2×2 matrix:

$$R_y(\theta) \triangleq \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{bmatrix} \tag{10}$$

while the $R_z(\theta)$ gate can be mathematically represented by

$$R_z(\theta) \triangleq \begin{bmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{bmatrix}. \tag{11}$$

Applying a quantum operator to a quantum state is equivalent to mathematically multiplying the state vector by the matrix representing the operator.

Quantum circuits are typically depicted by circuit diagrams, which are constructed and read from left to right. The construction of a quantum circuit diagram begins with the circuit wire represented by a line, as shown in Fig. 1(a). A line with no operator means that the qubit remains in the state in which it was previously prepared. The initial prepared state is denoted with a ket and label on the left of the wire, as shown in Fig. 1(b). A commonly used quantum operator is denoted by a box containing the letter that represents that operator straddling the line. For example, a binary gate is denoted as an operator box spanning two quantum wires, as illustrated in Fig. 1(c). An example of a two-qubit quantum circuit diagram is given by Fig. 1(d), which consists of a Hadamard operator, a controlled-NOT operator, and a measurement operator [represented by the "meter" symbol in Fig. 1(d)] applied to two qubits initially prepared in the ground state $|0\rangle$. The measurement operator converts a single qubit state $|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle$ into a probabilistic classical bit, which is 0 with probability $|\alpha|^2$, or 1 with probability $|\beta|^2$ [3].

III. Proposed Quantum Computing Method for Solving EM Problems Based on the FEM

This section presents the proposed quantum computing method for solving finite element equations in the EM domain in detail. We first reformulate the EM finite element equation into a quantum computation format. Then, we incorporate the SPAI preconditioner into the reformulation to deal with the large condition number of the EM finite element matrix. Next, we present a systematic method for efficient quantum state preparation of the RHS vector, followed by deriving the number of gates needed in the state preparation circuit. Finally, we illustrate the procedure of solving the preconditioned reformulated FEM equation using the HHL algorithm.

A. Reformulation of the Finite Element Equations in EM Problems for Quantum Computation

The final result of the finite element formulation for many problems in EM as well as in other engineering fields is a set of linear equations that can be written as follows [6]:

$$\boldsymbol{K}\boldsymbol{\phi} = \boldsymbol{c} \tag{12}$$

where $\boldsymbol{c}$ is the known vector describing the specific excitation, $\boldsymbol{K}$ is the finite element matrix, and $\boldsymbol{\phi}$ is the vector containing all the unknowns [i.e., degrees of freedom (DoFs)] used to approximate the electric fields in the solution domain.

Let M be the number of rows (or columns) of the matrix $\boldsymbol{K}$. The application of HHL to solving linear equations requires the system matrix to be Hermitian. In addition, the number of rows of the matrix should be an integer power of 2 to allow the use of a certain number of qubits to represent the RHS and the solution vectors. Since the finite element matrix $\boldsymbol{K}$ is not Hermitian directly and the number of rows of $\boldsymbol{K}$ is typically not an integer power of 2, we reformulate (12) into an equivalent equation where the system matrix is in Hermitian format. Let $\widetilde{\boldsymbol{A}}$ be the new matrix of the reformulated equation. In order to allow further formalization of the matrix in Sections III-B and III-D, we propose to formulate $\widetilde{\boldsymbol{A}}$ as follows:

$$\widetilde{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{0}_{M\times M} & \boldsymbol{K} & \boldsymbol{0}_{M\times m} \\ \boldsymbol{K}^{\dagger} & \boldsymbol{0}_{M\times M} & \boldsymbol{0}_{M\times m} \\ \boldsymbol{0}_{m\times M} & \boldsymbol{0}_{m\times M} & \widetilde{\boldsymbol{\Lambda}}_{m\times m} \end{bmatrix} \tag{13}$$

where the symbol $\dagger$ represents complex conjugate transpose, and m is given by

$$m = 2^{\lceil \log_2 M \rceil + 1} - 2M. \tag{14}$$

The expression $\lceil \log_2 M \rceil$ denotes rounding to the smallest integer that is greater than $\log_2 M$. The symbol $\widetilde{\boldsymbol{\Lambda}}$ represents an $m \times m$ diagonal matrix, defined as follows:

$$\widetilde{\boldsymbol{\Lambda}} = \begin{bmatrix} \widetilde{\lambda}_{\min} & & & & \\ & -\widetilde{\lambda}_{\min} & & & \\ & & \ddots & & \\ & & & \widetilde{\lambda}_{\min} & \\ & & & & -\widetilde{\lambda}_{\min} \end{bmatrix} \tag{15}$$

where $\widetilde{\lambda}_{\min}$ is the smallest positive eigenvalue of the $2M \times 2M$ Hermitian matrix $\widetilde{\boldsymbol{K}}$, defined as follows:

$$\widetilde{\boldsymbol{K}} = \begin{bmatrix} \boldsymbol{0}_{M\times M} & \boldsymbol{K} \\ \boldsymbol{K}^{\dagger} & \boldsymbol{0}_{M\times M} \end{bmatrix}. \tag{16}$$

Note that the matrix $\widetilde{\boldsymbol{\Lambda}}$ is a repetition of the smallest eigenvalue $\widetilde{\lambda}_{\min}$ for several times across the entire diagonal. In other words, $\widetilde{\boldsymbol{\Lambda}}$ only contains the smallest positive eigenvalue of matrix $\widetilde{\boldsymbol{K}}$, instead of all its eigenvalues. For a sparse matrix $\widetilde{\boldsymbol{K}}$, computing its minimum eigenvalue is much easier than computing all of its eigenvalues. For example, the value of $\lambda_{\min}$ can be precomputed very efficiently using classical numerical methods, such as the Arnoldi algorithm [18]. The "eigs" function in MATLAB is an example of implementation.

Let N be the number of rows (or columns) of $\widetilde{\boldsymbol{A}}$. According to (13), the value of N should be

$$N = 2^{\lceil \log_2 M \rceil + 1}. \tag{17}$$

Notably, the reformulated matrix $\widetilde{\boldsymbol{A}}$ is Hermitian, and its number of rows (or columns) is an integer power of 2, which satisfies the format requirements of the HHL algorithm.

Next, we formulate the RHS vector of the new equation as follows:

$$\widetilde{\boldsymbol{b}} = \left[\boldsymbol{c}^T \quad \boldsymbol{0}_{1\times M} \quad \underbrace{0 \quad 0 \quad \ldots \quad 0 \quad 0}_{m \text{ zeros}} \right]^{\mathrm{T}}. \tag{18}$$

Let us define $\boldsymbol{x}$ as a vector of N unknown elements. Then, the original finite element equation (12) can be reformulated into the following format:

$$\widetilde{\boldsymbol{A}}\boldsymbol{x} = \widetilde{\boldsymbol{b}} \tag{19}$$

where the second M elements (out of a total of N elements) of the N-dimensional vector $\boldsymbol{x}$ represent the electric field solution vector $\boldsymbol{\phi}$ and the N-dimensional vector $\widetilde{\boldsymbol{b}}$ contains the EM excitation vector $\boldsymbol{c}$.

We intend to propose a quantum computing method based on the HHL algorithm to solve (19). The performance of the HHL algorithm highly relies on the condition number of the reformulated matrix $\widetilde{\boldsymbol{A}}$. For HHL to achieve an exponential speedup, the condition number can scale at most polylogarithmically with the size of $\widetilde{\boldsymbol{A}}$ [15]. However, in most situations, the matrix $\boldsymbol{K}$ resulting from EM problems, especially that with a large dimension, is usually relatively ill conditioned [6]. The main reason for this phenomenon can be the existence of "bad aspect ratio" elements due to mesh refinement [19], [20]. As a consequence, the reformulated matrix $\widetilde{\boldsymbol{A}}$ is not well conditioned, either. This will lead to a large number of qubits in the use of HHL and reduce the HHL algorithm's efficiency. Since $\widetilde{\boldsymbol{A}}$ usually has a large condition number, we implement the SPAI preconditioner to the Hermitian matrix to improve the efficiency of the HHL algorithm according to the existing literature, as illustrated in Section III-B.

B. Preconditioning of the Reformulated Equation With SPAI

We address the above problem using the matrix preconditioning technique. We aim to find a preconditioning matrix (i.e., preconditioner) $\boldsymbol{P}$ and solve the preconditioned linear equation

$$\boldsymbol{PK}\boldsymbol{\phi} = \boldsymbol{Pc} \tag{20}$$

rather than solving the original linear equation (12) directly.

Commonly used preconditioners in the EM community include the diagonal, block diagonal, incomplete lower–upper (LU), symmetric successive overrelaxation (SSOR), and SPAI preconditioners [21], [22]. Among these preconditioning techniques, we adopt the SPAI technique [23], [24], [25] as the basis to develop our preconditioning method. The SPAI technique guarantees that the condition number of $\boldsymbol{PK}$ is much lower than that of $\boldsymbol{K}$, while the preconditioned matrix $\boldsymbol{PK}$ has good sparsity [15]. In particular, $\boldsymbol{P}$ is found by minimizing [23]

$$\left\| \boldsymbol{K}^T \boldsymbol{P}^T - \boldsymbol{I} \right\|_F^2 = \sum_{k=1}^{M} \left\| \left(\boldsymbol{K}^T \boldsymbol{P}^T - \boldsymbol{I} \right) \boldsymbol{e}_k \right\|_2^2 \tag{21}$$

where the expressions $\|\cdot\|_F$ and $\|\cdot\|_2$ denote the Frobenius norm and the L^2 norm, respectively; $\boldsymbol{I}$ represents the identity matrix; and $\boldsymbol{e}_k$ ($k = 1, \ldots, M$) is the kth column of the identity matrix.

One can separate (21) into M independent least squares problems [23]

$$\min_{\boldsymbol{p}'_k} \left\| \boldsymbol{K}^T \boldsymbol{p}'_k - \boldsymbol{e}_k \right\|_2, \quad k = 1, \ldots, M \tag{22}$$

where $\boldsymbol{p}'_k$ represents the kth column of $\boldsymbol{P}^T$. The SPAI technique imposes an initial sparsity constraint on $\boldsymbol{P}$ (e.g., to be the same as $\boldsymbol{K}$) and updates the sparsity pattern iteratively. Since $\boldsymbol{p}'_k$ is allocated with a fixed sparsity pattern, one can eliminate many rows in $\boldsymbol{p}'_k$ and the corresponding columns in $\boldsymbol{K}^T$ when solving the least squares problem (22). Since the finite element matrix $\boldsymbol{K}$ itself is sparse, one can further eliminate many rows in $\boldsymbol{K}^T$ when solving (22). This results in M least squares problems of a much smaller size than that of (22) as follows:

$$\min_{\hat{\boldsymbol{p}}'_k} \left\| \widehat{\boldsymbol{K}'} \hat{\boldsymbol{p}}'_k - \hat{\boldsymbol{e}}_k \right\|_2, \quad k = 1, \ldots, M \tag{23}$$

where $\widehat{\boldsymbol{K}'}$, $\hat{\boldsymbol{p}}'_k$, and $\hat{\boldsymbol{e}}_k$ represent the $\boldsymbol{K}^T$, $\boldsymbol{p}'_k$, and $\boldsymbol{e}_k$ after size reduction, respectively. More details of the SPAI technique are provided in Appendix A. One should notice that in simple cases where the conditional number of $\boldsymbol{K}$ is small enough to be handled by the standard HHL algorithm, the preconditioner can be simply set as the identity matrix, i.e., $\boldsymbol{P} = \boldsymbol{I}_{M\times M}$.

Once the preconditioner $\boldsymbol{P}$ is found, we formulate a preconditioned Hermitian matrix, $\boldsymbol{A}$, of size $N \times N$, as follows:

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{0}_{M\times M} & \boldsymbol{PK} & \boldsymbol{0}_{M\times m} \\ (\boldsymbol{PK})^{\dagger} & \boldsymbol{0}_{M\times M} & \boldsymbol{0}_{M\times m} \\ \boldsymbol{0}_{m\times M} & \boldsymbol{0}_{m\times M} & \boldsymbol{\Lambda}_{m\times m} \end{bmatrix} \tag{24}$$

where $\boldsymbol{\Lambda}$ represents an $m \times m$ diagonal matrix, defined as follows:

$$\boldsymbol{\Lambda} = \begin{bmatrix} \lambda_{\min} & & & & \\ & -\lambda_{\min} & & & \\ & & \ddots & & \\ & & & \lambda_{\min} & \\ & & & & -\lambda_{\min} \end{bmatrix} \tag{25}$$

where $\lambda_{\min}$ is the smallest positive eigenvalue of the $2M \times 2M$ preconditioned Hermitian matrix $\widetilde{\boldsymbol{K}}_{\text{precond}}$, defined as follows:

$$\widetilde{\boldsymbol{K}}_{\text{precond}} = \begin{bmatrix} \boldsymbol{0}_{M\times M} & \boldsymbol{PK} \\ (\boldsymbol{PK})^{\dagger} & \boldsymbol{0}_{M\times M} \end{bmatrix}. \tag{26}$$

Define the RHS vector of the preconditioned equation as follows:

$$\boldsymbol{b} = \begin{bmatrix} (\boldsymbol{Pc})^{\mathrm{T}} & \boldsymbol{0}_{1\times M} & \underbrace{0 \quad 0 \quad \ldots \quad 0 \quad 0}_{m \text{ zeros}} \end{bmatrix}^{\mathrm{T}}. \tag{27}$$

Now, we have the preconditioned finite element equation for solving EM problems

$$\boldsymbol{Ax} = \boldsymbol{b} \tag{28}$$

where the second M elements (out of a total of N elements) of the vector $\boldsymbol{x}$ represent the electric field solution vector $\boldsymbol{\phi}$. In Section III-C, we describe how to prepare the quantum state for representing the RHS vector $\boldsymbol{b}$.

C. Proposed Quantum State Preparation for the RHS Vector $\boldsymbol{b}$ in the Preconditioned Finite Element Equation

One crucial step for applying quantum computation to solve (28) is to set up the quantum circuit to prepare the quantum state to represent the RHS vector $\boldsymbol{b}$. The HHL's exponential speedup assumes the availability of an "efficient" preparation of the quantum state for $\boldsymbol{b}$ [8]. In the quantum linear systems algorithms literature, "efficient" is taken to be "polylogarithmic" in the system size N [9]. This section along with Section III-D propose a systematic method to construct such a quantum circuit.

Suppose that $\boldsymbol{b}$ is an N-dimensional complex-valued vector, i.e., $\boldsymbol{b} = [b_1\ b_2\ \cdots\ b_N]^{\mathrm{T}}$. Let n be the number of qubits used to represent the RHS vector $\boldsymbol{b}$, i.e., $n = \log_2 N$. To facilitate understanding, we first use the three-qubit case (i.e., $n = 3$) as an example to illustrate how to prepare the quantum state for $\boldsymbol{b}$. Then, we generalize the formulation for the n-qubit case. Notice that the magnitude of $\boldsymbol{b}$ is not necessarily 1. Since quantum states are normalized, we use $\bar{\boldsymbol{b}} = [\bar{b}_1, \bar{b}_2, \ldots, \bar{b}_8]^{\mathrm{T}}$ to represent the normalized vector of $\boldsymbol{b}$ as follows (in EM applications, there is always an input port with EM excitation, so the vector $\boldsymbol{b}$ always contains at least one nonzero entry):

$$\bar{\boldsymbol{b}} = \frac{\boldsymbol{b}}{\|\boldsymbol{b}\|_2}. \tag{29}$$

Let $|\bar{b}\rangle$ represent the quantum state corresponding to $\bar{\boldsymbol{b}}$. The quantum circuit for preparing $|\bar{b}\rangle$ consists of a magnitude preparation circuit followed by a phase preparation circuit, illustrated as follows.

1) Magnitude Preparation for Three-Qubit $|\bar{b}\rangle$: First, let us make the following definitions:

$$\xi_1 = \arctan\left|\frac{b_2}{b_1}\right| \tag{30}$$

$$\xi_2 = \arctan\left|\frac{b_4}{b_3}\right| \tag{31}$$

$$\xi_3 = \arctan\left|\frac{b_6}{b_5}\right| \tag{32}$$

$$\xi_4 = \arctan\left|\frac{b_8}{b_7}\right| \tag{33}$$

$$\Phi_1 = \arctan\sqrt{\frac{|b_3|^2 + |b_4|^2}{|b_1|^2 + |b_2|^2}} \tag{34}$$

$$\Phi_2 = \arctan\sqrt{\frac{|b_7|^2 + |b_8|^2}{|b_5|^2 + |b_6|^2}} \tag{35}$$

$$\Psi_1 = \arctan\left|\frac{\sum_{k=5}^{8}|b_k|^2}{\sum_{k=1}^{4}|b_k|^2}\right|. \tag{36}$$

Then, the quantum circuit to prepare the magnitudes of $|\bar{b}\rangle$, i.e., $|\bar{b}_1|, |\bar{b}_2|, \ldots, |\bar{b}_8|$, is shown in Fig. 2. Taking the one-qubit case [as shown in (3)] as an illustration, the rotation gate $R_y(\Psi)$ reallocates the proportions of the state $|\psi\rangle$ being $|0\rangle$ and $|1\rangle$ according to the phase parameter Ψ. In other words, by applying $R_y(\Psi)$ to the ground state $|0\rangle$, we can get the correct magnitudes α and β as desired (refer to Appendixes B and C for more details).

2) Phase Preparation for Three-Qubit $|\bar{b}\rangle$: Let θ_k be the phase of b_k, i.e., (if $b_i = 0$, we let $\theta_i = 0$)

$$\theta_1 = \angle b_1 \tag{37}$$

$$\theta_2 = \angle b_2 \tag{38}$$

$$\vdots$$

$$\theta_8 = \angle b_8. \tag{39}$$

We make the following definitions:

$$\bar{\xi}_1 = \frac{1}{2}(\theta_2 - \theta_1) \tag{40}$$

$$\bar{\xi}_2 = \frac{1}{2}(\theta_4 - \theta_3) \tag{41}$$

$$\bar{\xi}_3 = \frac{1}{2}(\theta_6 - \theta_5) \tag{42}$$

$$\bar{\xi}_4 = \frac{1}{2}(\theta_8 - \theta_7) \tag{43}$$

$$\bar{\Phi}_1 = \frac{1}{4}[(\theta_3 + \theta_4) - (\theta_1 + \theta_2)] \tag{44}$$

$$\bar{\Phi}_2 = \frac{1}{4}[(\theta_7 + \theta_8) - (\theta_5 + \theta_6)] \tag{45}$$

$$\bar{\Psi}_1 = \frac{1}{8}\left(\sum_{k=5}^{8}\theta_k - \sum_{k=1}^{4}\theta_k\right). \tag{46}$$

Then, the quantum circuit to prepare the phase of $|\bar{b}\rangle$ is shown in Fig. 3. Taking the one-qubit case [as shown in (3)] as an illustration, the application of the rotation gate $R_z(\bar{\Psi})$ to state $|\psi\rangle$ allows us to change the phase difference between the two coefficients α and β. In other words, by applying

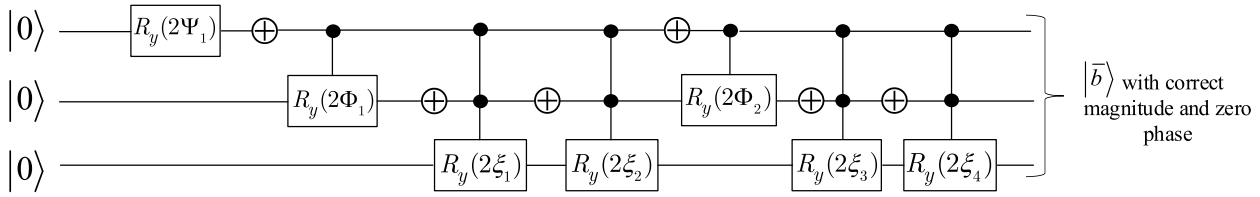


Fig. 2. Quantum circuit to implement the magnitude preparation of $|\bar{b}\rangle$: the three-qubit example. The symbol $\oplus$ represents the NOT operator. The black dot(s) upon each R_y gate means that the gate is controlled by the corresponding qubit(s). For example, the two black dots above the gate $R_y(2\xi_1)$ means that this R_y gate applied on the third qubit is controlled by the top two qubits, conditional on these two qubits' states being $|00\rangle$.

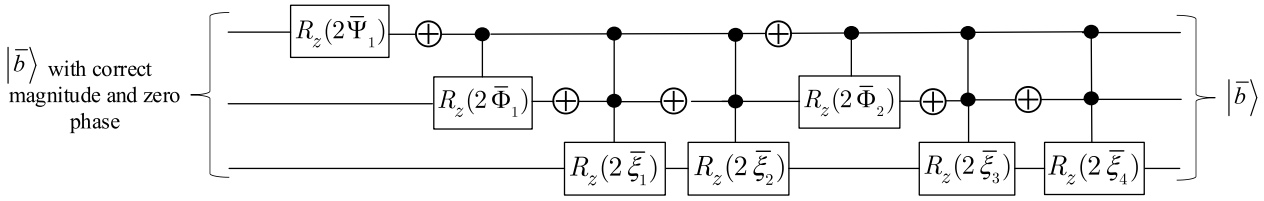


Fig. 3. Quantum circuit to implement the phase preparation of $|\bar{b}\rangle$: the three-qubit example. The black dot(s) upon each R_z gate means that the gate is controlled by the corresponding qubit(s). For example, the two black dots above the gate $R_z(2\bar{\xi}_2)$ mean that this R_z gate applied on the third qubit is controlled by the top two qubits, conditional on these two qubits' states being $|01\rangle$.

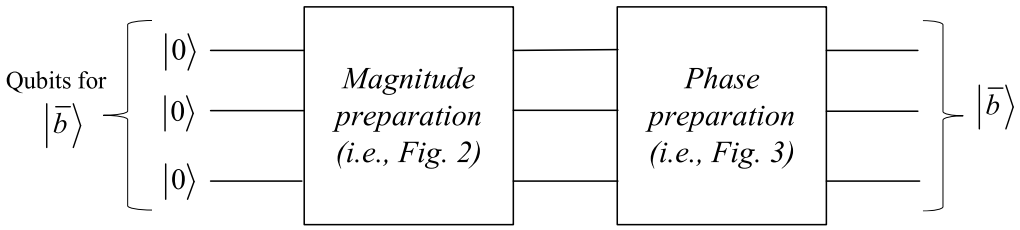


Fig. 4. Overall quantum circuit to prepare a three-qubit $|\bar{b}\rangle$, including magnitude and phase.

$R_z(\bar{\Psi})$ to the state obtained from the magnitude preparation circuit, we can get the desired phase difference between the coefficients. The overall circuit to prepare the quantum state $|\bar{b}\rangle$, including magnitude and phase, is shown in Fig. 4. Appendixes B and C illustrate the quantum computation procedures of preparing $|\bar{b}\rangle$ for the one-qubit and two-qubit cases, respectively.

Now, we generalize the above formulation to the n-qubit case. To ease the generalization procedure, we slightly change the indices of the elements in $\boldsymbol{b}$ and unify the symbols ξ, Φ, and Ψ as one symbol Φ with respective superscripts. Specifically, let $\boldsymbol{b}$ represent an N-dimensional complex-valued vector, i.e., $\boldsymbol{b} = [b_0\ b_1\ \cdots\ b_{N-1}]$, where $N = 2^n$. Let the phase of b_j be denoted by θ_j ($j = 0, 1, \ldots, N-1$). Then, $\boldsymbol{b}$ can be represented as follows:

$$\boldsymbol{b} = \begin{bmatrix} |b_0|e^{i\theta_0} \\ |b_1|e^{i\theta_1} \\ \vdots \\ |b_{N-1}|e^{i\theta_{N-1}} \end{bmatrix}. \tag{47}$$

From here onward in this section, we use the symbols i and l ($i, l \in \{0, 1, \ldots, n-1\}$) to denote the indices of qubits and use the symbols j and k ($j, k \in \{0, 1, \ldots, N-1\}$) to denote a decimal integer in the range of from 0 to $N-1$. The value of k can be converted into the binary representation as follows:

$$k = \begin{bmatrix} k_0\ k_1\ \cdots\ k_{l-1} \end{bmatrix}_2 \tag{48}$$

where $1 \le l \le n-1$ and k_i ($k_i \in \{0, 1\}$) denotes the ith element in the binary representation of k. An example of the relationship between k and k_i for $l = 3$ is shown in Table I.

We define the symbol $\Phi_k^{(l)}$ as the phase parameter for the kth rotation gate (i.e., R_y) at the lth qubit, where

$$k = \begin{cases} 0, & \text{if } l = 0 \\ \displaystyle\sum_{i=0}^{l-1} k_i 2^{l-1-i}, & \text{if } l = 1, 2, \ldots, n-1. \end{cases} \tag{49}$$

Note that (49) simply relates the value of k with its binary representation $[k_0\ k_1\ \cdots\ k_{l-1}]_2$. For a given l, there are, in total, 2^l possible values (i.e., $k \in \{0, 1, \ldots, 2^l - 1\}$) for k whose binary representation can be written as (48). Then, we derive the value of $\Phi_k^{(l)}$ as follows:

$$\Phi_k^{(l)} = \arctan \sqrt{\frac{\sum_{j=I_1}^{I_1+I_2} |b_j|^2}{\sum_{j=I_0}^{I_0+I_2} |b_j|^2}} \tag{50}$$

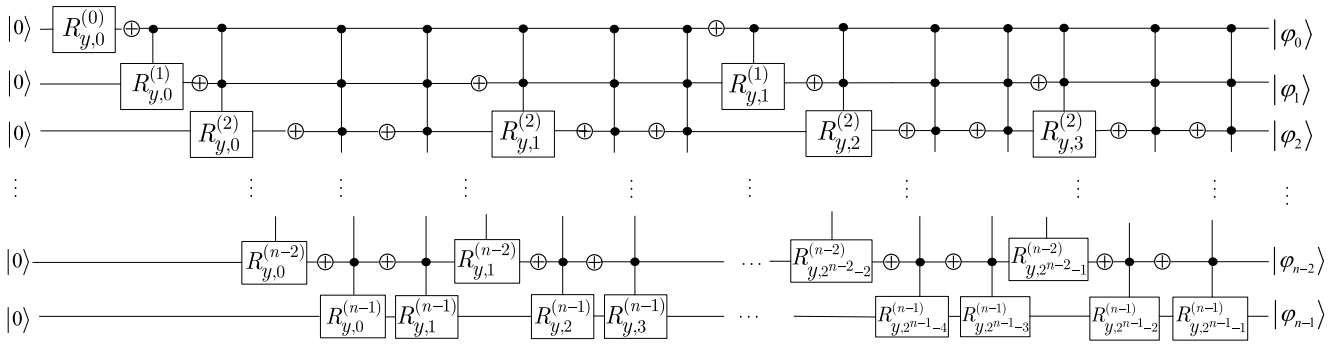


Fig. 5. Quantum circuit for preparing the magnitude of an n-qubit $|\bar{b}\rangle$, where the final state $|\varphi_0\varphi_1\dots\varphi_{n-1}\rangle$ represents $|\bar{b}\rangle$ with correct magnitude and zero phase. The symbol $R_{y,k}^{(l)}$ inside each block represents the rotation gate $R_y(2\Phi_k^{(l)})$.

TABLE I
RELATIONSHIP BETWEEN k AND k_i WHEN $l = 3$: PRESENTED FOR THE UNDERSTANDING OF (48) AND (49)

k	k_0	k_1	k_2
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

where

$$I_0 = \begin{cases} 0, & \text{if } l = 0 \\ \sum_{i=0}^{l-1} k_i 2^{n-1-i}, & \text{if } l = 1, 2, \dots, n-1 \end{cases} \tag{51}$$

$$I_1 = I_0 + 2^{n-1-l} \tag{52}$$

$$I_2 = 2^{n-1-l} - 1. \tag{53}$$

We define another symbol $\bar{\Phi}_k^{(l)}$ as the phase parameter for the kth rotation gate (i.e., R_z) at the lth qubit, where the definition of k is the same as that in (49). The value of $\bar{\Phi}_k^{(l)}$ is derived as follows:

$$\bar{\Phi}_k^{(l)} = \frac{1}{2^{n-l}} \left[\sum_{j=I_1}^{I_1+I_2} \theta_j - \sum_{j=I_0}^{I_0+I_2} \theta_j \right], \quad l = 0, 1, \dots, n-1 \tag{54}$$

where I_0, I_1, and I_2 are defined in (51)–(53), respectively. Similar to the three-qubit case, the quantum circuit for preparing an n-qubit quantum state $|\bar{b}\rangle$ consists of a magnitude preparation circuit followed by a phase preparation circuit. They are constructed as follows.

3) Magnitude Preparation for n-Qubit $|\bar{b}\rangle$: The magnitude preparation circuit is set up step by step as follows.

Step M1) Initialize the states on all the n qubits as $|0\rangle$.

Step M2) For each $l \in \{0, 1, \dots, n-1\}$ and $k \in \{0, 1, \dots, 2^l - 1\}$, find the phase parameter $\Phi_k^{(l)}$ according to (50).

Step M3) Apply the kth rotation gate $R_y(2\Phi_k^{(l)})$ to the lth qubit, where $l \in \{0, 1, \dots, n-1\}$ and $k \in \{0, 1, \dots, 2^l - 1\}$.

Step M4) If $l > 0$, then the kth rotation gate $R_y(\Phi_k^{(l)})$ should be controlled by the qubits $l' \in \{0, 1, \dots, l-1\}$, conditional on these qubits' states being $|k_0\rangle \otimes |k_1\rangle \otimes \cdots \otimes |k_{l-1}\rangle$. Here, the values of $k_0, k_1, \dots, k_{l-1}$ are determined by the binary representation of k, as shown in Table I.

Fig. 5 shows the quantum circuit for preparing the magnitude of an n-qubit $|\bar{b}\rangle$.

4) Phase Preparation for n-Qubit $|\bar{b}\rangle$: The phase preparation circuit is set up step by step as follows.

Step P1) Get the states of all the n qubits from the magnitude preparation circuit.

Step P2) For each $l \in \{0, 1, \dots, n-1\}$ and $k \in \{0, 1, \dots, 2^l - 1\}$, find the phase parameter $\bar{\Phi}_k^{(l)}$ according to (54).

Step P3) Apply the kth rotation gate $R_z(2\bar{\Phi}_k^{(l)})$ to the lth qubit, where $l \in \{0, 1, \dots, n-1\}$ and $k \in \{0, 1, \dots, 2^l - 1\}$.

Step P4) If $l > 0$, then the kth rotation gate $R_z(\bar{\Phi}_k^{(l)})$ should be controlled by the qubits $l' \in \{0, 1, \dots, l-1\}$, conditional on these qubits' states being $|k_0\rangle \otimes |k_1\rangle \otimes \cdots \otimes |k_{l-1}\rangle$. Here, the values of $k_0, k_1, \dots, k_{l-1}$ are determined by the binary representation of k, as shown in Table I.

Fig. 6 shows the quantum circuit for preparing the phase of an n-qubit $|\bar{b}\rangle$.

D. Proposed Quantum Gate Reduction for Efficient State Preparation of the Preconditioned RHS Vector $\boldsymbol{b}$

This section proposes a quantum-gate-reduction method to prepare the quantum state $|\bar{b}\rangle$ efficiently, i.e., to have the gate count in the quantum circuit increase logarithmically in the vector size N. In EM problems, the RHS vector $\boldsymbol{b}$ is evaluated based on the EM excitation in the input port of the 3-D EM structure. In the solution domain of 3-D FEM, typically only a

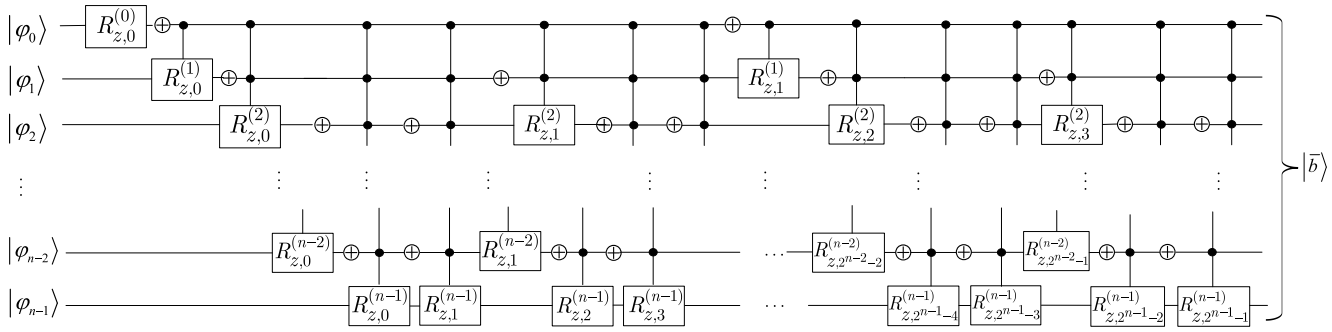


Fig. 6. Quantum circuit for preparing the phase of an n-qubit $|\bar{b}\rangle$. The input to the circuit is obtained from the output of the magnitude preparation circuit, i.e., Fig. 5. The symbol $R_{z,k}^{(l)}$ inside each block represents the rotation gate $R_z(2\bar{\Phi}_k^{(l)})$.

few cells (and, therefore, a small portion of DoFs) are located in the input port of the 3-D EM structure. Thus, the RHS vector $\boldsymbol{b}$ in EM problems is always sparse. We notice that the number of gates needed to prepare $|\bar{b}\rangle$ depends on the sparsity of the vector $\boldsymbol{b}$. If $\boldsymbol{b}$ is sparse, many $\Phi_k^{(l)}$ and $\bar{\Phi}_k^{(l)}$ will be zeros. The R_y or R_z gates can be omitted in the state preparation circuit for such zero-valued cases. Therefore, it is natural to conjecture that the number of gates needed to prepare the quantum state $|\bar{b}\rangle$ is proportional to the number of nonzeros in $\boldsymbol{b}$ (more details are provided in the next subsection).

However, our investigations show that if we directly prepare the state $|\bar{b}\rangle$ based on the procedure described in Section III-C, the number of gates needed for a sparse vector $\boldsymbol{b}$ is fewer than that needed for a dense $\boldsymbol{b}$, but still much larger than the number of nonzeros in $\boldsymbol{b}$. Our further investigations show that the number of gates needed to prepare $|\bar{b}\rangle$ is proportional to the number of nonzeros in $\boldsymbol{b}$ only when the nonzero elements of $\boldsymbol{b}$ are all located at the top (or the bottom) of vector $\boldsymbol{b}$. However, in the default 3-D FEM, the DoFs in the input port are not arranged in a specific order. Therefore, we propose a method to renumber the DoFs in the input port with respect to the DoFs in other cells of the 3-D EM structure, which is achieved by permutating the finite element matrix in a specific way. By using the proposed method, the number of gates needed for the preparation of $|\bar{b}\rangle$ can be reduced to achieve an efficient state preparation of the RHS vector $\boldsymbol{b}$.

Let d represent the number of nonzeros in $\boldsymbol{Pc}$. Let $\{\boldsymbol{W}_1, \boldsymbol{W}_2, \ldots, \boldsymbol{W}_{n_w}\}$ represent the set of permutation matrices used to swap all the nonzero elements of $\boldsymbol{Pc}$ to the top locations, where n_w $(0 \leq n_w \leq d)$ is the total number of row permutations that need to be done for such purpose. At the worst case, $n_w = d$, which means that, in total, d row permutations need to be done. At the best case, $n_w = 0$, which means that the nonzero elements are already at the top locations of vector $\boldsymbol{Pc}$, and thus, no row permutations need to be done. Then, solving (20) is equivalent to solving

$$\boldsymbol{W}_{n_w}(\cdots(\boldsymbol{W}_2(\boldsymbol{W}_1\boldsymbol{PK})))\boldsymbol{\phi} = \boldsymbol{W}_{n_w}(\cdots(\boldsymbol{W}_2(\boldsymbol{W}_1\boldsymbol{Pc}))). \tag{55}$$

Let $\Gamma(\cdot)$ denote the above series of row permutation operations on $\boldsymbol{PK}$ and $\boldsymbol{Pc}$, i.e.,

$$\Gamma(\cdot) = \boldsymbol{W}_{n_w}(\cdots(\boldsymbol{W}_2(\boldsymbol{W}_1(\cdot)))). \tag{56}$$

Next, we formulate a new Hermitian matrix, $\boldsymbol{A}_{\text{perm}}$, as follows:

$$\boldsymbol{A}_{\text{perm}} = \begin{bmatrix} \boldsymbol{0}_{M\times M} & \Gamma(\boldsymbol{PK}) & \boldsymbol{0}_{M\times m} \\ (\Gamma(\boldsymbol{PK}))^{\dagger} & \boldsymbol{0}_{M\times M} & \boldsymbol{0}_{M\times m} \\ \boldsymbol{0}_{m\times M} & \boldsymbol{0}_{m\times M} & \boldsymbol{\Lambda}_{m\times m} \end{bmatrix}. \tag{57}$$

We further define a new RHS vector $\boldsymbol{b}_{\text{perm}}$ as follows:

$$\boldsymbol{b}_{\text{perm}} = \begin{bmatrix} (\Gamma(\boldsymbol{Pc}))^{\text{T}} & \boldsymbol{0}_{1\times M} & \underbrace{0 \quad 0 \quad \ldots \quad 0 \quad 0}_{m \text{ zeros}} \end{bmatrix}^{\text{T}}. \tag{58}$$

Now, solving (28) is equivalent to solving

$$\boldsymbol{A}_{\text{perm}}\boldsymbol{x} = \boldsymbol{b}_{\text{perm}}. \tag{59}$$

Note that the permutation matrices $\{\boldsymbol{W}_1, \boldsymbol{W}_2, \ldots, \boldsymbol{W}_{n_w}\}$ allow all the nonzero elements of $\boldsymbol{b}$ to be swapped to the top of vector $\boldsymbol{b}$. Such permutations on the rows do not affect the final solution to EM problems using the proposed quantum computing method but will greatly reduce the number of gates needed for preparing the state $|\bar{b}\rangle$, as analyzed in Section III-E, and demonstrated in Section IV-B.

E. Analytical Upper and Lower Bounds for the Number of Gates Needed in the RHS State Preparation

In this section, we first empirically show the relationship between the number of gates needed in the RHS state preparation circuit and the number of nonzeros in the RHS vector $\boldsymbol{b}$. Then, inspired by these empirical results, we derive analytically the upper and lower bounds of the number of gates needed in preparing $|\bar{b}\rangle$, to theoretically demonstrate that the gate count increases logarithmically with respect to the EM excitation vector size N.

Let d denote the number of nonzeros in the sparse vector $\boldsymbol{b}$. Let us for now assume that d is an integer power of 2, i.e., $d = 2^q$, where q is an integer in the range $1 \leq q \leq n$ and n represents the number of qubits used to prepare $|\bar{b}\rangle$. Let us further assume that all the nonzero entries in $\boldsymbol{b}$ have been swapped to the top of $\boldsymbol{b}$. We perform the following experiments.

For a given pair of $N \in \{8, 16, \ldots, 256\}$ and $d \in \{2, 4, \ldots, 256\}$, we generate a random sparse vector $\boldsymbol{b}$ (whose nonzero elements are all at its top positions) and find the value of $\Phi_k^{(l)}$ according to (50), where $n = \log_2 N$. Since $\boldsymbol{b}$ is sparse,

TABLE II
NUMBER OF GATES REQUIRED IN MAGNITUDE PREPARATION VERSUS THE MATRIX SIZE N AND THE NUMBER OF NONZEROS d

$N_{\text{req}}^{\text{mag}}$	N=8	N=16	N=32	N=64	N=128	N=256
d = 2	1	1	1	1	1	1
d = 4	3	3	3	3	3	3
d = 8	7	7	7	7	7	7
d = 16	N/A	15	15	15	15	15
d = 32	N/A	N/A	31	31	31	31
d = 64	N/A	N/A	N/A	63	63	63
d = 128	N/A	N/A	N/A	N/A	127	127
d = 256	N/A	N/A	N/A	N/A	N/A	255

TABLE III
NUMBER OF GATES REQUIRED IN PHASE PREPARATION VERSUS THE MATRIX SIZE N AND THE NUMBER OF NONZEROS d

$N_{\text{req}}^{\text{phase}}$	N=8	N=16	N=32	N=64	N=128	N=256
d = 2	3	4	5	6	7	8
d = 4	4	5	6	7	8	9
d = 8	7	8	9	10	11	12
d = 16	N/A	15	16	17	18	19
d = 32	N/A	N/A	31	32	33	34
d = 64	N/A	N/A	N/A	63	64	65
d = 128	N/A	N/A	N/A	N/A	127	128
d = 256	N/A	N/A	N/A	N/A	N/A	255

many $\Phi_k^{(l)}$ will be zeros. We refer to an R_y gate with zero phase parameter (i.e., $\Phi_k^{(l)} = 0$) as a *redundant gate*, which can be omitted in the magnitude preparation circuit, and refer to an R_y gate with nonzero phase parameter (i.e., $\Phi_k^{(l)} \neq 0$) as a *required gate*, which cannot be omitted. Similarly, we find the value of $\bar{\Phi}_k^{(l)}$ according to (54) and refer to an R_z gate with $\bar{\Phi}_k^{(l)} \neq 0$ and $\bar{\Phi}_k^{(l)} = 0$ as a *required gate* and a *redundant gate*, respectively.

Let $N_{\text{req}}^{\text{mag}}$ and $N_{\text{req}}^{\text{phase}}$ represent the number of required gates for magnitude preparation and phase preparation, respectively. Tables II and III show, respectively, the value of $N_{\text{req}}^{\text{mag}}$ versus N and d, and that of $N_{\text{req}}^{\text{phase}}$ versus N and d. From Table II, one can clearly observe that $N_{\text{req}}^{\text{mag}}$ is proportional to (approximately one time) the number of nonzeros d. The insight from Table III appears to be not as clear as that in Table II. However, as d increases (e.g., when $d \geq 16$), one can still observe a tendency that $N_{\text{req}}^{\text{phase}}$ is proportional to (approximately one time) d.

The above experiments reveal that the number of required gates is proportional to the number of nonzeros in $\boldsymbol{b}$. In the following descriptions, we formally derive the number of gates required for preparing $|\bar{b}\rangle$ and provide the upper and lower bounds of the number of required gates for preparing $|\bar{b}\rangle$ when the number of nonzeros d is not an integer power of 2.

1) Number of Gates Needed for Magnitude Preparation: First, let us derive the number of required gates in the magnitude preparation, $N_{\text{req}}^{\text{mag}}$, with respect to the matrix size N and the number of nonzeros d. It can be found from (49) and (51) that

$$I_0 = 2^{n-l}k. \tag{60}$$

Substituting this into (52), we obtain

$$I_1 = 2^{n-l}k + 2^{n-l-1}. \tag{61}$$

From (50), we can see that whether the value of $\Phi_k^{(l)}$ is zero depends on the values of b_j, where $j \in \{I_1, I_1 + 1, \ldots, I_1 + I_2\}$. Considering that only the first d elements of $\boldsymbol{b}$ are nonzeros, while the remaining $N - d$ elements are all zeros, we conclude that the value of $\Phi_k^{(l)}$ will not be zero when the following condition is satisfied:

$$I_1 = 2^{n-l}k + 2^{n-l-1} \leq d - 1. \tag{62}$$

This should be natural, since if $I_1 > d-1$, all the values of b_j in the numerator $\sum_{j=I_1}^{I_1+I_2} |b_j|^2$ will be zero, and thus, $\Phi_k^{(l)}$ will be zero. So, the number of required gates in the magnitude preparation circuit is the number of combinations of l and k that satisfy the condition of (62).

The condition of (62) is equivalent to

$$2^{n-l}k + 2^{n-l-1} \leq 2^q - 1. \tag{63}$$

From (63), we obtain the range of k for a given l as follows:

$$k \leq 2^{q+l-n} - 2^{-1} - 2^{l-n}. \tag{64}$$

Let the maximum value of k be denoted by $k_{\max}$. Then, it can be derived from (64) that (considering that k is an integer and $l - n \leq -1$)

$$\begin{aligned} k_{\max} &= \left\lceil 2^{q+l-n} - \left(2^{-1} + 2^{l-n}\right) \right\rceil \\ &= 2^{q+l-n} - 1. \end{aligned} \tag{65}$$

On the other hand, the value of l should satisfy the following condition due to the constraint that $k \geq 0$:

$$2^{q+l-n} - 2^{-1} - 2^{l-n} \geq 0 \tag{66}$$

which is equivalent to

$$l \geq n - 1 - \log_2\left(2^q - 1\right). \tag{67}$$

Let the minimum value of l be $l_{\min}$, greater than which $\Phi_k^{(l)} \neq 0$ for $k = 0, 1, \ldots, k_{\max}$. Then, from (67), we can derive the value of $l_{\min}$ as follows:

$$\begin{aligned} l_{\min} &= \left\lfloor n - 1 - \log_2\left(2^q - 1\right) \right\rfloor \\ &= n - 1 - \left\lceil \log_2\left(2^q - 1\right) \right\rceil \\ &= n - q. \end{aligned} \tag{68}$$

The insight from (65) and (68) is that for a given $l \in \{n - q, n - q + 1, \ldots, n - 1\}$, there are 2^{q+l-n} required gates (i.e., $k = 0, 1, \ldots, 2^{q+l-n} - 1$) for the lth qubit. Accordingly, we can show the value of $N_{\text{req}}^{\text{mag}}$ for given n and q as follows:

$$\begin{aligned} N_{\text{req}}^{\text{mag}} &= \sum_{l=n-q}^{n-1} 2^{q+l-n} \\ &= 2^0 + 2^1 + 2^2 + \cdots + 2^{q-1} \\ &= 2^q - 1. \end{aligned} \tag{69}$$

2) Number of Gates Needed for Phase Preparation: Next, we derive the number of required gates in the phase preparation, $N_{\text{req}}^{\text{phase}}$, with respect to the matrix size N and the number of nonzeros d. From (54), one can find that whether the value of $\bar{\Phi}_k^{(l)}$ is zero depends on the values of b_j, where $j \in \{I_1, I_1 + 1, \ldots, I_1 + I_2\} \cup \{I_0, I_0 + 1, \ldots, I_0 + I_2\}$. Since $l \leq n-1$, we have $I_1 = I_0 + 2^{n-l-1} > I_0$. Consequently, we can conclude that the value of $\bar{\Phi}_k^{(l)}$ will not be zero if the following condition is satisfied:

$$I_0 = 2^{n-l}k \leq 2^q - 1. \tag{70}$$

This should be natural, since if $I_0 > 2^q - 1$, all the values of b_j, where $j \in \{I_1, I_1 + 1, \ldots, I_1 + I_2\} \cup \{I_0, I_0 + 1, \ldots, I_0 + I_2\}$, will be zero, and thus, $\bar{\Phi}_k^{(l)}$ will be zero. So, the total number of required gates in the phase preparation circuit is the number of combinations of l and k that satisfy the condition of (70).

From (70), we can obtain the range of k for a given l as follows:

$$k \leq 2^{q+l-n} - 2^{l-n}. \tag{71}$$

Note that the value of l can be any integer in the set $\{0, 1, \ldots, n-1\}$, since $2^{q+l-n} - 2^{l-n} > 0$ always holds. Let the maximum value of k be denoted by $k_{\max}$. Based on (71), we derive the value of $k_{\max}$ in two cases.

Case 1: When $q + l - n \leq -1$, i.e., $l \leq n - q - 1$, we have $2^{-n} \leq 2^{l-n} < 2^{q+l-n} \leq 1$. In this case, we can derive the value of $k_{\max}$ as follows:

$$k_{\max} = \left\lceil 2^{q+l-n} - 2^{l-n} \right\rceil = 0. \tag{72}$$

Therefore, there is only one required gate (i.e., $k = 0$) for the lth qubit when $0 \leq l \leq n - q - 1$.

Case 2: When $q + l - n \geq 0$, i.e., $l \geq n - q$, we have $2^{q+l-n} \geq 1$ and $0 < 2^{l-n} < 1$. In this case, we can derive the value of $k_{\max}$ as follows:

$$k_{\max} = \left\lceil 2^{q+l-n} - 2^{l-n} \right\rceil = 2^{q+l-n} - 1. \tag{73}$$

Therefore, there are 2^{q+l-n} required gates (i.e., $k = 0, 1, \ldots, 2^{q+l-n} - 1$) for the lth qubit when $n - q \leq l \leq n - 1$.

Now, combining both cases, for given n and q, we can show the total number of combinations of l and k that satisfy (70), i.e., the value of $N_{\text{req}}^{\text{phase}}$, as follows:

$$\begin{aligned} N_{\text{req}}^{\text{phase}} &= 1 \times \left[(n - q - 1) + 1\right] + \sum_{l=n-q}^{n-1} 2^{q+l-n} \\ &= 1 \times \left[(n - q - 1) + 1\right] + \left(2^0 + 2^1 + 2^2 + \cdots + 2^{q-1}\right) \\ &= n - q + 2^q - 1. \end{aligned} \tag{74}$$

3) Total Number of Gates Needed for Preparing $|\bar{b}\rangle$: Combining (69) and (74), we can find the total number of required gates for the preparation of $|\bar{b}\rangle$ as follows:

$$\begin{aligned} N_{\text{req}}^{\text{total}} &= N_{\text{req}}^{\text{mag}} + N_{\text{req}}^{\text{phase}} \\ &= n - q + 2\left(2^q - 1\right) \\ &= 2d - \log_2 d + \log_2 N - 2. \end{aligned} \tag{75}$$

Note that the above equation reveals the analytical relation between the number of gates needed to prepare the state $|\bar{b}\rangle$ and the sparsity (and size) of the EM excitation vector $\boldsymbol{b}$.

By using our proposed method, the gate count can be minimized, and we have theoretically demonstrated that the gate count increases logarithmically in N and proportionally with d. When using the FEM to solve EM problems, the sparsity of $\boldsymbol{b}$ only depends on the number of FEM cells in the input port and does not depend on the meshes between the input and output ports. That said, as the 3-D structure in the middle becomes more complex, denser meshes are typically required to accurately approximate the electric fields in the middle part of the solution domain, resulting in bigger matrix $\boldsymbol{A}$ and bigger vector $\boldsymbol{b}$ in the FEM. Even under this situation, the number of cells on the input port remains constant, and therefore, the sparsity of $\boldsymbol{b}$ remains constant. Because of this, our proposed method guarantees that even if this state preparation procedure for the EM excitation vector $\boldsymbol{b}$ is included, the HHL is still exponentially faster than its classical counterpart.

4) Upper and Lower Bounds for N_{req}^{total}: In most practical cases, the number of nonzero elements in $\boldsymbol{b}$ (i.e., the value of d) is not necessarily an integer power of 2. In such cases, our formula (75) provides the upper and lower bounds for the total number of required gates in the preparation circuit for $|\bar{b}\rangle$. Formally, for an N-dimensional RHS vector $\boldsymbol{b}$ with d nonzero elements on its top locations, where $N = 2^n$ and $0 \leq d \leq N$, the total number of required gates to prepare the corresponding quantum state $|\bar{b}\rangle$, $N_{\text{req}}^{\text{total}}$, is upper bounded by $N_{\text{req}}^{\text{up}}$ and lower bounded by $N_{\text{req}}^{\text{low}}$, i.e.,

$$N_{\text{req}}^{\text{low}} \leq N_{\text{req}}^{\text{total}} \leq N_{\text{req}}^{\text{up}}$$

where

$$N_{\text{req}}^{\text{up}} = 2^{\lceil \log_2 d \rceil + 1} - \lceil \log_2 d \rceil + \log_2 N - 2 \tag{76}$$

and

$$N_{\text{req}}^{\text{low}} = 2^{\lfloor \log_2 d \rfloor + 1} - \lfloor \log_2 d \rfloor + \log_2 N - 2. \tag{77}$$

F. Solving the Preconditioned Reformulated FEM Equations Using Quantum Computation

In this section, we solve the reformulated finite element equation after preconditioning and permutation [i.e., (59)] with the HHL algorithm. In order to perform quantum computation, we define three sets of qubits, as illustrated in Fig. 7: an ancilla qubit (also called the ancilla register), a set of L qubits (also called the work register), and a set of n qubits (also called the I/O register).

The quantum computation procedure begins with preparing the n-qubit state $|\bar{b}_{\text{perm}}\rangle$ through a magnitude preparation circuit (denoted by P_1 in Fig. 7) and a phase preparation circuit (denoted by P_2 in Fig. 7) on the I/O register to represent the EM excitation vector $\boldsymbol{b}_{\text{perm}}$. At the same time, we initialize both the ancilla and the work registers in the $|0\rangle$ state.

Next, we apply the quantum operator $e^{i\boldsymbol{A}_{\text{perm}}t2^r}$ to the I/O register [2], [3], [26] for different values of r, where t is a user-defined hyperparameter and $r = 0, 1, \ldots, L-1$. These operations, together with a set of Hadamard gates [2] and quantum Fourier transform operator on the work register, form the quantum phase estimation (QPE) operation, which produces the eigenvalues of $\boldsymbol{A}_{\text{perm}}$ in binary format in the work

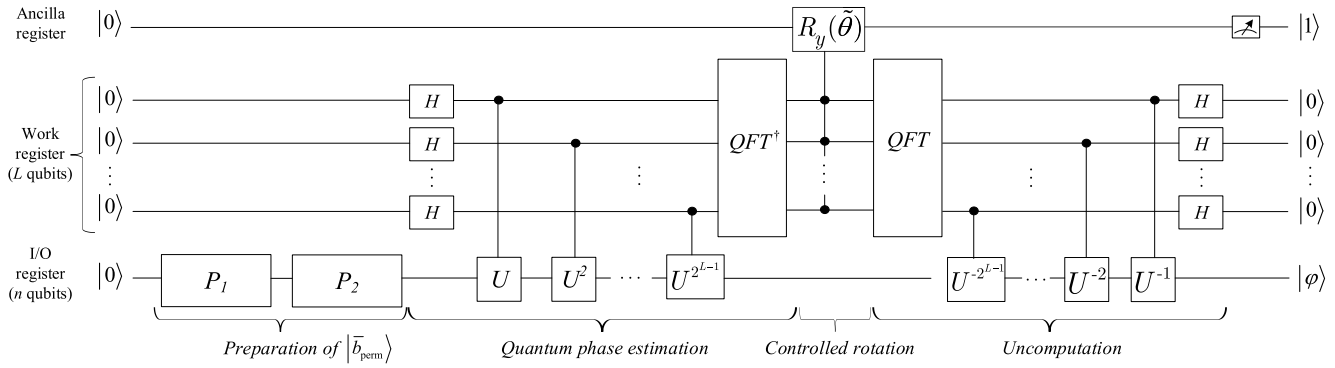


Fig. 7. Schematic of the HHL algorithm with the proposed quantum state preparation circuit. Each square block represents a specific quantum operator/circuit. The symbols H, U, QFT, and R_y represent the Hadamard operator, the Hamiltonian operator, the quantum Fourier transform operator, and the rotation operator, respectively. The symbols P_1 and P_2 denote the magnitude preparation circuit (i.e., Fig. 5) and the phase preparation circuit (i.e., Fig. 6), respectively.

register. For explanation purpose, let λ_j and $|u_j\rangle$ represent the jth eigenvalue and the corresponding eigenvector of $\boldsymbol{A}_{\text{perm}}$, respectively. Let $\tilde{\lambda}_j$ be the estimated value of λ_j on the work register.

Following the phase estimation, we perform a controlled rotation on the ancilla register to rotate the ancilla qubit from $|0\rangle$ to $\sin(\tilde{\theta})|0\rangle + \cos(\tilde{\theta})|1\rangle$, where $\tilde{\theta}$ is found from $\tilde{\theta} = \arccos(C/\tilde{\lambda}_j)$ and C is a user-defined hyperparameter. Note that the phase parameter of the controlled rotation gate, $\tilde{\theta}$, depends on the estimated eigenvalue $\tilde{\lambda}_j$ on the work register.

Finally, the work register is uncomputed by the algorithm back to $|0\rangle$, while the ancilla and the I/O registers combined will have the following state [8]:

$$\sum_{j=1}^{N} \gamma_j \left[\sqrt{1 - \frac{C^2}{\tilde{\lambda}_j^2}} |0\rangle_{\text{ancilla}} + \frac{C}{\tilde{\lambda}_j} |1\rangle_{\text{ancilla}} \right] \otimes |u_j\rangle_{\text{IO}} \tag{78}$$

where the symbol $\otimes$ denotes tensor product and $\gamma_j = \langle u_j | \bar{b}_{\text{perm}} \rangle$.

Let $|\varphi\rangle$ represent the final state of the I/O register. According to (78), when we examine the ancilla register and post-selects on the $|1\rangle$ outcome, the state of the I/O register will be proportional to an estimation of the EM solution vector $|x\rangle$

$$|\varphi\rangle = \sum_j \frac{\gamma_j C}{\tilde{\lambda}_j} |u_j\rangle_{\text{IO}} \approx C|x\rangle. \tag{79}$$

The above computation procedure produces a quantum-mechanical representation $|x\rangle$ of the solution of the unknown vector $\boldsymbol{x}$. When we examine $|x\rangle$, we obtain the result $|i\rangle$ with probability $|x_i|^2$. In order to evaluate $|x_i|$, the quantum computation process of HHL is repeated many times as the Monte Carlo analysis.

G. Discussions

The quantum solution provided by (79) is accurate except for the binary quantization of the eigenvalues λ_j. However, the solution obtained by retrieving the values in (79) from quantum computer into classical computer will not be as perfect as the theoretical solution. The reasons for such imperfectness are as follows. First, the QPE subroutine in HHL finds the approximated eigenvalues of the system matrix through quantization. In doing this, the precise eigenvalues are approximated by its binary representations, which inherently lead to the loss of accuracy to a certain extent. The accuracy level depends on the number of qubits in the work register (denoted by L). Increasing the value of L improves the eigenvalues' accuracy but leads to more quantum computing resources. Second, the quantum state $|x\rangle$ obtained from HHL encodes the entries of $\boldsymbol{x}$ in its amplitudes precisely. However, due to the probabilistic nature of quantum computing, to obtain the values of the entries in $\boldsymbol{x}$, one needs to perform Monte Carlo analysis to obtain statistics and do approximations. In doing this, HHL algorithm is repeated for a large number of times, and the probabilities obtained from the statistics are taken as good approximations to (the magnitude square of) the entries in $\boldsymbol{x}$. In general, the accuracy of the solution obtained from the HHL circuit tends to increase, as the number of circuit repetitions increases, until a certain point where the solution converges and further repetitions do not significantly improve the accuracy [2]. Moreover, as the number of circuit repetitions increases, the simulation time will also increase [27]. Finding the optimal number of circuit repetitions to balance solution accuracy and compute time is an important aspect of using the HHL algorithm to solve practical EM problems, which remains an open subject in the literature. In many cases, the optimal number of circuit repetitions needs to be found by a trial-and-error process [2], [27].

Next, we discuss the robustness of the proposed algorithm against noise. The proposed algorithm is implemented through the Google's *Cirq* library [28], which serves as a noiseless quantum computing simulator. That is to say, the state preparations of the RHS vector, unitary operations, and output state measurements are all implemented by the noiseless gates provided by *Cirq*. However, practical quantum computers are vulnerable to noise, thereby affecting the solutions' quality obtained by the proposed method. We leave the evaluation of the proposed algorithm's robustness against noise as an important future work. Moreover, we leave it an open research topic to take advantages of the error mitigation techniques [29], [30],

[31], [32], [33] to mitigate the effects of noise on the proposed method when performed on real quantum hardware.

Although the proposed method was developed in the context of EMs, it is applicable to finite element problems in other fields as long as the final format of the problem to be solved is a set of linear equations. If the system matrix is well conditioned, the preconditioner $\boldsymbol{P}$ can be simply set as the identity matrix. If the RHS vector $\boldsymbol{b}$ is a full vector, the proposed quantum-gate-reduction method described in Section III-D can be omitted.

Current noisy intermediate-scale quantum (NISQ) devices have a limited number of qubits and limited circuit depth. The variational quantum algorithms (VQAs) are a class of quantum algorithms that aim to achieve a practical quantum advantage over classical algorithms on NISQ devices [35], [36], [37]. Despite its great potential of handling the constraints of near-term quantum computers, VQAs' performance relies on the availability of an efficient ansatz. Each computational problem has a problem-specific ansatz, and there is no generic ansatz for all the computational problems. The development of a new ansatz for a new computational task is typically nontrivial and requires a lot of human experience. To the best of the authors' knowledge, there is no ansatz reported in the literature for solving FEM linear equations in 3-D EM problems. Future noiseless large-scale quantum computers are foreseen to have a much larger number of available qubits and much higher circuit depths than today's NISQ devices. However, how to leverage the power of such much matured quantum computers to accelerate the process of solving large-scale EM problems remains an open subject in the literature. Our work builds a theoretical basis for achieving quantum advantages on much matured quantum computers in the future.

Specifically, for solving a linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ arising from the FEM in EM problems, the comparison of runtime complexity between the best classical algorithm, i.e., conjugate gradient (CG), and the HHL algorithm is $\mathcal{O}(Ns\kappa \log(1/\epsilon))$ versus $\mathcal{O}(\log(N)s^2\kappa^2/\epsilon)$, where N is the dimension of $\boldsymbol{A}$, s is the sparsity (i.e., the maximum number of nonzero entries per row) of $\boldsymbol{A}$, κ is the condition number of $\boldsymbol{A}$, and ϵ is the desired precision [2], [8], [9]. In other words, HHL solves the quantum linear system problem exponentially faster than CG with respect to N. However, there are several critical requirements for the HHL to achieve this exponential speedup, e.g., the following hold: 1) a well-conditioned system matrix with constant sparsity; 2) an "efficient" preparation of $|\bar{b}\rangle$; and 3) an "efficient" readout of the solution $|x\rangle$ [34]. This study tackles with the aforementioned important caveats associated with HHL in the following three aspects. First, we propose to apply the SPAI preconditioner to the EM system matrix to meet the first requirement of HHL. The application of the SPAI preconditioner makes the EM-based system matrix better conditioned, while not increasing too much the sparsity of the EM-based system matrix. Second, we propose an "efficient" quantum state preparation method to meet the second requirement of HHL. Our proposed method guarantees that even if this state preparation procedure for the EM excitation vector $\boldsymbol{b}$ is included, the HHL is still exponentially faster than its classical counterpart. Third, although, in this study, we read out all the entries of $\boldsymbol{x}$ for illustrative purposes, the solution of interest to many EM problems (e.g., calculating the S-parameters) is in a linear combination format well suited to the solution format of the HHL algorithm [38].

Finally, we discuss how the finite element matrix (referred to as global matrix hereafter) in EM fits the constant sparsity requirement of the HHL algorithm. The global matrix in EM problems is obtained by assembling the local matrices over all the FEM cells. Suppose that tetrahedron elements [39] are used in the discretization process, each DoF on a local edge/face in a tetrahedron corresponds to one row of the global matrix. For a given edge in a tetrahedron under consideration, the number of nonzeros in the row corresponding to this edge is not directly related to the number of tetrahedrons/cells in the whole solution space but related to the number of neighbor tetrahedrons of the tetrahedron under consideration. As the number of cells increases (which means that N increases), the number of neighbor tetrahedrons typically does not increase. Therefore, the maximum number of nonzeros in each row (i.e., sparsity s) does not increase. In other words, the sparsity of our EM-based matrix $\boldsymbol{A}$ does not increase, as the matrix size N increases. Because of this, finite element matrices in EM problems fit the HHL's requirement of constant sparsity, so that the HHL's computational benefits can be achieved when it is being applied to solve EM problems including the new gate-reduction algorithm in preparing the quantum state for $\boldsymbol{b}$.

IV. Examples

A. 2-D Electrostatic Problem

We first consider a 2-D electrostatic problem [40], with the aim of solving the Poisson's equation

$$-\Delta\phi(x, y) = \frac{\rho}{\varepsilon} \tag{80}$$

with boundary condition

$$\phi(x, y) = 0 \tag{81}$$

where the symbol Δ denotes the Laplace operator, while the symbols ϕ, ε, and ρ represent the electric scalar potential, permittivity, and electric charge density, respectively.

We consider a simple case by setting $\rho/\varepsilon = 1$. The solution domain of this 2-D problem is $[-1, 1]^2$, and we divide the whole solution domain into 200 cells, as depicted in Fig. 8. This division results in a total of $M = 121$ DoFs in the finite element equation. Consequently, we have the matrix $\widetilde{\boldsymbol{A}}$ of size $N \times N = 256 \times 256$ and an eight-qubit quantum state $|\bar{b}\rangle$, i.e., $n = 8$. The condition number of $\widetilde{\boldsymbol{A}}$ is 58.48, which is small enough to be handled by the standard HHL algorithm. Thus, we simply set the preconditioner $\boldsymbol{P}$ as $\boldsymbol{P} = \boldsymbol{I}_{121\times121}$. In this study, we prepare the eight-qubit RHS state $|\bar{b}\rangle$ based on the quantum state preparation method described in Section III-C. We should note that the RHS vector $\boldsymbol{b}$ in this example is real-valued. Therefore, we only have the magnitude preparation circuit, and the phase preparation circuit can be omitted. For the magnitude preparation circuit, we have $l \in \{0, 1, \ldots, 7\}$ and $k \in \{0, 1, \ldots, 2^l - 1\}$. Had the vector $\boldsymbol{b}$ been dense, the

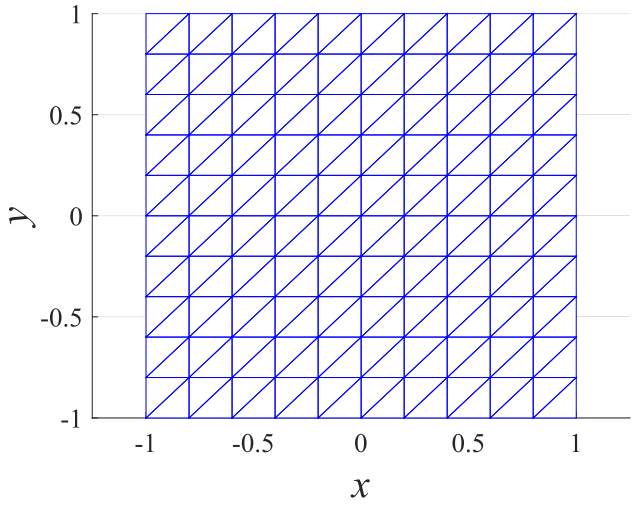


Fig. 8. FEM mesh generated for the 2-D electrostatic problem with 121 DoFs. This results in a 256 × 256 matrix $\boldsymbol{A}$ and an eight-qubit quantum state $|\bar{b}\rangle$ as inputs to the proposed quantum computing method.

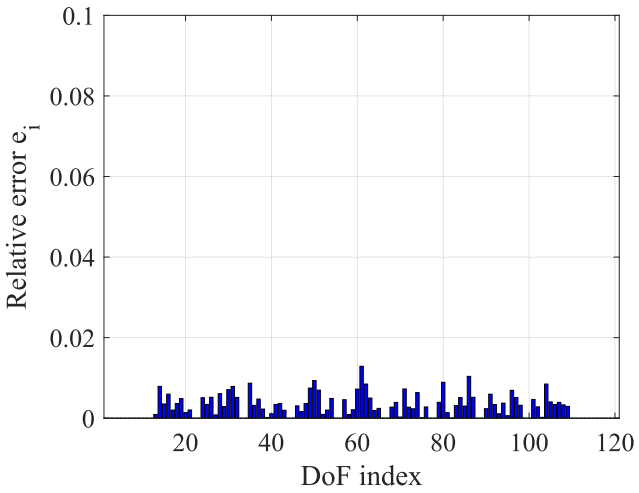
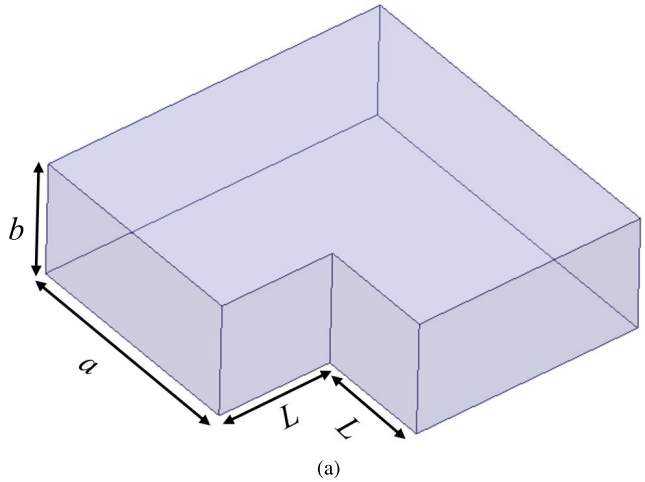


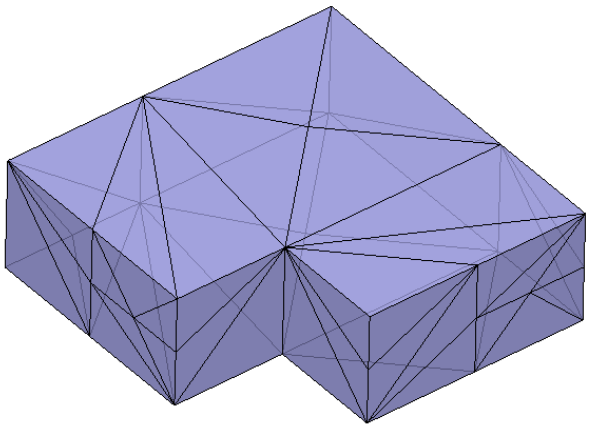
Fig. 9. Relative errors of the 121 DoFs: a comparison between the quantum solution and the classical solution for the 2-D electrostatic problem.

number of gates needed for preparing the state $|\bar{b}\rangle$ would have been 255. In this example, the number of nonzero elements in $\boldsymbol{b}$ is 81, i.e., $d = 81$. With the existing state preparation method in the quantum literature [16], the number of gates needed to prepare the state $|\bar{b}\rangle$ is 88. By swapping the 81 nonzero elements to the top locations of vector $\boldsymbol{b}$ using the proposed gate-reduction method, the number of gates needed to prepare the quantum state $|\bar{b}_{\text{perm}}\rangle$ is reduced to 80.

We program the reformulation of the finite element equation and the codes for preparing the quantum state $|\bar{b}_{\text{perm}}\rangle$ and combine this program with HHL based on the codes provided in [2]. We write the program in Python utilizing the *Cirq* library developed by Google [28]. In order to evaluate $|x_i|$, HHL has to be performed many times as the Monte Carlo analysis. In typical cases, $\boldsymbol{x}$ is read out in a linear combination format, which requires us to run HHL about 40 000 times. In this study, we read out all the individual components of $\boldsymbol{x}$ for illustrative purposes.



(a)



(b)

Fig. 10. Structure and mesh for the 3-D example. (a) Structure of the waveguide bend defined for finite element simulation, where $a = 8.636$ (mm), $b = 4.318$ (mm), and $L = 4.318$ (mm). (b) FEM mesh generated to perform the EM simulation.

We do a comparison between the quantum solution and the classical solution with respect to the relative error of each DoF of interest, e_i, which is defined as follows:

$$e_i = \frac{\left|\left(\left|x_i^{\text{QC}}\right| - \left|x_i^{\text{CC}}\right|\right)\right|}{\sqrt{\frac{1}{|\mathcal{I}_{\text{in}}|}\sum_{i\in\mathcal{I}_{\text{in}}}\left|x_i^{\text{CC}}\right|^2}} \quad \forall i \in \mathcal{I}_{\text{in}} \tag{82}$$

where $|x_i^{\text{QC}}|$ and $|x_i^{\text{CC}}|$ represent the magnitude of the ith component in the quantum solution and that in the classical solution, respectively. The symbol $\mathcal{I}_{\text{in}}$ represents the set containing all the indices for the DoFs of interest, while $|\mathcal{I}_{\text{in}}|$ denotes the number of elements in $\mathcal{I}_{\text{in}}$. The relative error of each DoF of interest is shown in Fig. 9. It is observed that the relative errors for all the DoFs of interest are quite small.

B. 3-D EM Wave Propagation Problem

Next, we consider a 3-D EM problem, i.e., the EM wave propagation in a waveguide bend, as shown in Fig. 10(a). The cross section of the bend is 8.636 × 4.318 mm. We are interested in calculating the reflection coefficient (i.e., S_{11}) at a frequency of 27.5 GHz using the 3-D FEM. The mesh

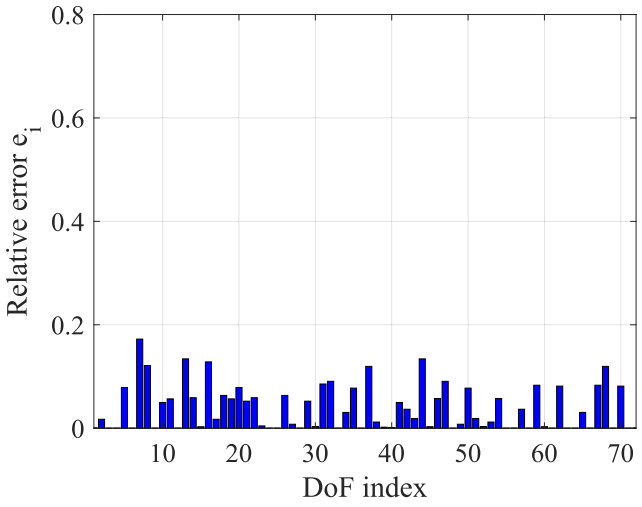


Fig. 11. Relative errors of the 72 DoFs of interest: a comparison between the quantum solution and the classical solution for the 3-D EM problem.

for the FEM is obtained from the HFSS simulator, as shown in Fig. 10(b). We mesh the whole solution domain into 53 cells, which results in a total of $M = 496$ DoFs in the finite element equation. The condition number of $\boldsymbol{K}$ is 2.2528×10^8. Without performing the proposed preconditioning technique, the standard HHL algorithm requires 29 qubits (i.e., $L = 29$) in the work register to perform QPE. This makes it memory-intensive to simulate the HHL circuit. We apply the proposed SPAI-based preconditioning technique and solve the resultant preconditioned FEM equations with HHL. For the SPAI technique, we set $\varepsilon = 0.4$ and $s = 6$. After preconditioning, we have the preconditioned matrix $\boldsymbol{A}$ of size $N \times N = 1024 \times 1024$ and a ten-qubit quantum state $|\bar{b}\rangle$, i.e., $n = 10$. The condition number of the preconditioned system matrix $\boldsymbol{A}$ is 343.96, which requires only seven qubits (i.e., $L = 7$) in the work register to perform QPE. This demonstrates the effectiveness of the proposed preconditioning method in reducing the number of qubits needed in applying HHL to solve EM problems.

We next apply the proposed quantum-gate-reduction method to move all the nonzeros in $\boldsymbol{b}$ to the top of vector $\boldsymbol{b}$, forming the permuted RHS vector $\boldsymbol{b}_{\text{perm}}$. The row indices of $\boldsymbol{A}$ are swapped accordingly to obtain the permuted system matrix $\boldsymbol{A}_{\text{perm}}$. The quantum circuit for preparing the quantum state $|\bar{b}_{\text{perm}}\rangle$ is based on Figs. 5 and 6. For this example, the number of qubits in the I/O register is 10. The RHS vector $\boldsymbol{b}$ in this example is complex-valued. Therefore, in the magnitude and phase preparation circuits, we have $l \in \{0, 1, \ldots, 9\}$ and $k \in \{0, 1, \ldots, 2^l - 1\}$. Had the vector $\boldsymbol{b}$ been dense, the number of gates needed for preparing the state $|\bar{b}\rangle$ would have been 2046. In this example, the number of nonzero elements in $\boldsymbol{b}$ is 149, i.e., $d = 149$. With the existing state preparation method in the quantum literature [16], the number of gates needed to prepare the state $|\bar{b}\rangle$ is 513. By swapping the 149 nonzero elements to the top locations of vector $\boldsymbol{b}$ using the proposed quantum-gate-reduction method, the number of gates needed to prepare the quantum state $|\bar{b}_{\text{perm}}\rangle$ is reduced to 303. Note that this number is inside the range between the upper bound $N_{\text{req}}^{\text{up}} = 512$ and the lower bound $N_{\text{req}}^{\text{low}} = 257$ as given by our derived formulas (76) and (77). Moreover, the proposed gate-reduction method guarantees that even if the state preparation procedure for the EM excitation vector $\boldsymbol{b}$ is included, the HHL algorithm is still exponentially faster than its classical counterpart.

For illustrative purposes, we read out all the individual components of $\boldsymbol{x}$ by running the HHL algorithm five million times. The evaluation of the reflection coefficient only relies on the DoFs on the input port. For the considered mesh, there are 72 DoFs on the input port. Therefore, among all the second 496 components of $\boldsymbol{x}$, we are interested in the 72 components corresponding to the DoFs on the input port, i.e., $|\mathcal{I}_{\text{in}}| = 72$. In Fig. 11, we compare the quantum and classical solutions for these 72 DoFs of interest with the metric of relative error defined in (82). We can see that the relative errors for all the DoFs of interest are small.

V. Conclusion

In this article, we have described a quantum computing method for solving EM problems through finite element equations. We have provided a systematic approach to construct the quantum circuit for the state preparation of the RHS vector in applying the FEM to solving EM problems. Moreover, to achieve an efficient quantum state preparation, we have proposed a specific method to reduce the number of gates needed in the quantum state preparation process, by exploring the fact that in EM problems, the RHS vector is always sparse. The upper and lower bounds for the number of gates needed in state preparation have also been derived. To tackle with the poorly conditioned finite element matrix in EM, a matrix preconditioner has been adopted to precondition the finite element linear equation, reducing the number of qubits required in using quantum computation to solve EM problems. The performance of the proposed method has been verified by two EM examples.

Considering that the hardware development of quantum computers is at an early stage, we have used small-scale EM examples to demonstrate the formulation in this article. Due to the limited number of qubits and circuit depth of current quantum computers, applying quantum computing (e.g., the HHL method) to solve small-scale EM problems offers no benefits compared with classical computing today. However, the significant ongoing progress in quantum hardware development will allow the future quantum computers to be capable of solving large-scale EM problems. For large-scale EM problems, the HHL method will (in theory) achieve an exponential speedup over the best-known classical computing method [2], [8], [9]. The main scope of this article is to explore how to formulate EM problems into a quantum computation format, which can be leveraged in the future to achieve a quantum speedup over classical solvers. From this perspective, we consider our study as an initial but important step toward achieving quantum advantages over classical EM solvers in the future when much matured quantum computing hardware becomes available. Applying the proposed quantum method to more practical EM problems demands continuous development in quantum computing technology and its integration into EM simulations.

Algorithm 1 SPAI Algorithm [23]

Input: $\boldsymbol{K}'$, initial sparsity pattern $\boldsymbol{S}_{\text{ini}}$, ε, s

for $k = 1, \ldots, M$ **do**
- Determine the initial sparsity pattern $\mathcal{J}$ for $\boldsymbol{p}'_k$ based on $\boldsymbol{S}_{\text{ini}}$;
- Find the row indices $\mathcal{I}$ of the corresponding nonzero entries in $\boldsymbol{K}'$;
- Compute the solution $\boldsymbol{p}'_k$ to the least squares problem (87) and its residual $\boldsymbol{r}$ given by (91);
- **while** $\|\boldsymbol{r}\|_2 > \varepsilon$ **do**
  - Set $\mathcal{L}$ to be the set of indices l for which $\boldsymbol{r}(l) \neq 0$;
  - Set $\widetilde{\mathcal{J}}$ equal to the set of all new column indices of $\boldsymbol{K}'$ that appear in all $\mathcal{L}$ rows but not in $\mathcal{J}$;
  - For each $j \in \widetilde{\mathcal{J}}$ compute ρ_j^2 according to (93);
  - Reduce $\widetilde{\mathcal{J}}$ to only contain the s column indices j with the smallest ρ_j;
  - Augment $\mathcal{J}$ as $\mathcal{J} = \mathcal{J} \cup \widetilde{\mathcal{J}}$;
  - Determine the new row indices $\mathcal{I}$ and solve the new least squares problem (87);
  - Compute the new residual $\boldsymbol{r} = \boldsymbol{K}'\boldsymbol{p}'_k - \boldsymbol{e}_k$.

Output: The SPAI preconditioner $\boldsymbol{P}$

APPENDIX A
COMPUTATION OF THE SPAI PRECONDITIONER

Let us first consider computing a sparse approximate left inverse $\boldsymbol{P}$ for a given initial sparsity structure (denoted by $\boldsymbol{S}_{\text{ini}}$), which is described by the nonzero entries of $\boldsymbol{S}_{\text{ini}}$. To align with the vector and matrix notations used in this article, we compute the transpose of $\boldsymbol{P}$ instead of computing $\boldsymbol{P}$ itself directly. Let $\boldsymbol{K}'$ denote the transpose of $\boldsymbol{K}$, i.e.,

$$\boldsymbol{K}' = \boldsymbol{K}^T. \tag{83}$$

Let $\boldsymbol{P}'$ denote the transpose of the unknown preconditioner $\boldsymbol{P}$. The SPAI technique minimizes $\|\boldsymbol{K}^T\boldsymbol{P}' - \boldsymbol{I}\|$ in the Frobenius norm to obtain an explicit approximate inverse $\boldsymbol{P}'$ of $\boldsymbol{K}'$. In particular, $\boldsymbol{P}'$ is found by minimizing [23]

$$\left\|\boldsymbol{K}^T\boldsymbol{P}' - \boldsymbol{I}\right\|_F^2 = \sum_{k=1}^{M}\left\|\left(\boldsymbol{K}^T\boldsymbol{P}' - \boldsymbol{I}\right)\boldsymbol{e}_k\right\|_2^2 \tag{84}$$

where the expressions $\|\cdot\|_F$ and $\|\cdot\|_2$ denote the Frobenius norm and the L^2 norm, respectively; $\boldsymbol{I}$ represents the identity matrix; and $\boldsymbol{e}_k$ ($k = 1, \ldots, M$) is the kth column of the identity matrix.

One can separate (84) into M independent least squares problems [23]

$$\min_{\boldsymbol{p}'_k}\|\boldsymbol{K}^T\boldsymbol{p}'_k - \boldsymbol{e}_k\|_2, \quad k = 1, \ldots, M \tag{85}$$

where $\boldsymbol{p}'_k$ represents the kth column of $\boldsymbol{P}'$. Thus, we can solve (85) in parallel to obtain an approximate inverse $\boldsymbol{P}'$ of $\boldsymbol{K}'$.

Let $p'_k(j)$ be the jth element of $\boldsymbol{p}'_k$, where $j = 1, \ldots, M$. Let $\mathcal{J}$ be the set containing the indices of the nonzeros in $\boldsymbol{p}'_k$, i.e., $\mathcal{J} = \{j | j = 1, \ldots, M, p'_k(j) \neq 0\}$. Let $\mathcal{I}$ be the set containing the row indices i, such that $\boldsymbol{K}'(i, \mathcal{J})$ is not identically zero. These definitions allow us to eliminate all zero rows in the submatrix $\boldsymbol{K}'(\cdot, \mathcal{J})$, resulting in the following submatrix of a very small size:

$$\widehat{\boldsymbol{K}'} = \boldsymbol{K}'(\mathcal{I}, \mathcal{J}). \tag{86}$$

If we further define $\hat{\boldsymbol{e}}_k = \boldsymbol{e}_k(\mathcal{I})$, then we have the following reduced-sized least squares problem equivalent to (22) [23]:

$$\min_{\hat{\boldsymbol{p}}'_k}\left\|\widehat{\boldsymbol{K}'}\hat{\boldsymbol{p}}'_k - \hat{\boldsymbol{e}}_k\right\|_2. \tag{87}$$

The size of this least squares problem is $|\mathcal{I}| \times |\mathcal{J}|$, which is much smaller than that of (22), since $\boldsymbol{K}'$ and $\boldsymbol{P}'$ are both highly sparse. We can find the solution to (87) as follows [23]:

$$\hat{\boldsymbol{p}}'_k = \widehat{\boldsymbol{R}}^{-1}\hat{\boldsymbol{\eta}}(1 : |\mathcal{J}|) \tag{88}$$

where $\widehat{\boldsymbol{R}} = \boldsymbol{R}(1 : |\mathcal{J}|, \cdot)$, $\hat{\boldsymbol{\eta}} = \boldsymbol{Q}^T\hat{\boldsymbol{e}}_k$, $\boldsymbol{Q}$ and $\boldsymbol{R}$ are obtained from executing the QR decomposition of $\widehat{\boldsymbol{K}}'$, i.e.,

$$\widehat{\boldsymbol{K}}' = \boldsymbol{Q}\boldsymbol{R}. \tag{89}$$

Once $\hat{\boldsymbol{p}}'_k$ is obtained from (88) for each $k = 1, \ldots, M$, we let $\boldsymbol{p}'_k(\mathcal{J}) = \hat{\boldsymbol{p}}_k$ and obtain the approximate inverse $\boldsymbol{P}$ for the given initial sparsity structure as follows:

$$\boldsymbol{P} = \left[\boldsymbol{p}'_1 \quad \boldsymbol{p}'_2 \quad \cdots \quad \boldsymbol{p}'_M\right]^{\mathrm{T}}. \tag{90}$$

Now, we elaborate on the improvement of $\boldsymbol{P}$ by augmenting its sparsity structure. Define the residual for the kth column of $\boldsymbol{P}'$ as follows:

$$\boldsymbol{r} = \boldsymbol{K}'\boldsymbol{p}'_k - \boldsymbol{e}_k. \tag{91}$$

Assume that $\|\boldsymbol{r}\|_2$ is not equal to zero, we will augment the set of indices contained in $\mathcal{J}$ to reduce $\|\boldsymbol{r}\|_2$. Let ε be the prescribed tolerance for $\boldsymbol{r}$. Denote by $\mathcal{L}$ the set containing the indices l for which $\boldsymbol{r}(l) \neq 0$. For each $l \in \mathcal{L}$, define $\mathcal{N}_l$ as the corresponding index set, which consists of the column indices of nonzero entries in the lth row of $\boldsymbol{K}'$ that are not yet contained in $\mathcal{J}$. Let $\widetilde{\mathcal{J}}$ be the set containing all the potential new candidate indices that can be added to $\mathcal{J}$; then

$$\widetilde{\mathcal{J}} = \bigcup_{l\in\mathcal{L}}\mathcal{N}_l. \tag{92}$$

To find the new indices $j \in \widetilde{\mathcal{J}}$ that lead to the most profitable reduction in $\|\boldsymbol{r}\|_2$, we compute an auxiliary quantity ρ_j^2 for each $j \in \widetilde{\mathcal{J}}$ as follows [23]:

$$\rho_j^2 = \|\boldsymbol{r}\|_2^2 - \frac{\left(\boldsymbol{r}^T\boldsymbol{K}'\boldsymbol{e}_j\right)^2}{\left\|\boldsymbol{K}'\boldsymbol{e}_j\right\|_2^2} \tag{93}$$

where $\boldsymbol{e}_j$ is the jth column of identity matrix. Let s be the number of nonzero entries added to $\boldsymbol{p}'_k$ each time we update its sparsity pattern. Next, we select from $\widetilde{\mathcal{J}}$ the s indices with the smallest ρ_j and add them to $\mathcal{J}$. With the augmented set of indices $\mathcal{J}$, we solve the reduced-size least squares problem (87) again. This improves the kth row of the preconditioner $\boldsymbol{P}$. We repeat this procedure for $\boldsymbol{p}'_k$ ($k = 1, \ldots, M$) until the norm of the residual is lower than the prescribed tolerance ε. The overall SPAI algorithm is shown in Algorithm 1.

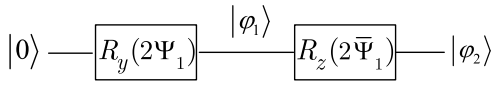


Fig. 12. Quantum circuit for preparing $|\bar{b}\rangle$: one-qubit case.

Appendix B
Illustration of the Quantum Computation Procedure for Preparing $|\bar{b}\rangle$: One-Qubit Case

In this section, we illustrate the quantum computation procedure for preparing the state $|\bar{b}\rangle$ in the one-qubit case. In this case, we have

$$\boldsymbol{b} = \begin{bmatrix} |b_1|e^{i\theta_1} \\ |b_2|e^{i\theta_2} \end{bmatrix}. \tag{94}$$

The corresponding quantum state for the normalized vector $\bar{\boldsymbol{b}}$ is mathematically represented as follows:

$$|\bar{b}\rangle \triangleq \begin{bmatrix} |\bar{b}_1|e^{i\theta_1} \\ |\bar{b}_2|e^{i\theta_2} \end{bmatrix}. \tag{95}$$

The quantum circuit for preparing the one-qubit state $|\bar{b}\rangle$ includes an R_y gate followed by an R_z gate, as shown in Fig. 12. In Fig. 12, we can find the values of Ψ_1 and $\bar{\Psi}_1$ according to (50)–(54) as follows:

$$\Psi_1 = \arctan\frac{|b_2|}{|b_1|} \tag{96}$$

$$\bar{\Psi}_1 = \frac{1}{2}(\theta_2 - \theta_1). \tag{97}$$

Let the state after applying the operator $R_y(2\Psi_1)$ to the ground state $|0\rangle$ be denoted by $|\varphi_1\rangle$. Substituting (96) into (10) allows us to derive the state $|\varphi_1\rangle$ as follows:

$$\begin{aligned} |\varphi_1\rangle &\triangleq \begin{bmatrix} \cos\Psi_1 & -\sin\Psi_1 \\ \sin\Psi_1 & \cos\Psi_1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \cos\Psi_1 \\ \sin\Psi_1 \end{bmatrix} = \begin{bmatrix} |\bar{b}_1| \\ |\bar{b}_2| \end{bmatrix} \end{aligned} \tag{98}$$

which is the state $|\bar{b}\rangle$ with correct magnitude and zero phase.

Now, let the state after applying the operator $R_z(2\bar{\Psi}_1)$ to the state $|\varphi_1\rangle$ be denoted by $|\varphi_2\rangle$. Substituting (97) into (11) allows us to derive the state $|\varphi_2\rangle$ as follows:

$$\begin{aligned} |\varphi_2\rangle &\triangleq \begin{bmatrix} e^{-i\frac{\theta_2-\theta_1}{2}} & 0 \\ 0 & e^{i\frac{\theta_2-\theta_1}{2}} \end{bmatrix}\begin{bmatrix} |\bar{b}_1| \\ |\bar{b}_2| \end{bmatrix} \\ &= \begin{bmatrix} |\bar{b}_1|e^{i\frac{\theta_1-\theta_2}{2}} \\ |\bar{b}_2|e^{i\frac{\theta_2-\theta_1}{2}} \end{bmatrix}. \end{aligned} \tag{99}$$

In quantum computation, it is the relative difference between the phases for different coefficients that is important. A common phase $e^{i\theta_0}$ applied to the entire state will be ignored by the quantum computer. This common phase is called the *global phase* in the quantum computation literature [2]. Based on this, we define a global phase, θ_0, as follows:

$$\theta_0 = \frac{\theta_1 + \theta_2}{2}. \tag{100}$$

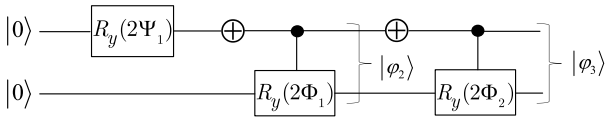


Fig. 13. Quantum circuit for preparing the magnitude of $|\bar{b}\rangle$: two-qubit case.

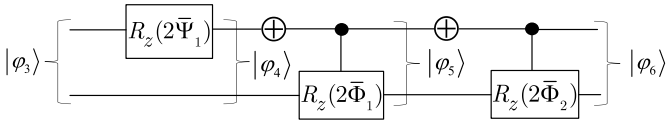


Fig. 14. Quantum circuit for preparing the phase of $|\bar{b}\rangle$: two-qubit case.

Recall that a global phase change has no impact on quantum measurements, and the state $|\varphi_2\rangle$ in (99) is equal to the desired state $|\bar{b}\rangle$ up to θ_0, i.e.,

$$|\varphi_2\rangle \sim \begin{bmatrix} |\bar{b}_1|e^{i\left(\frac{\theta_1-\theta_2}{2}+\theta_0\right)} \\ |\bar{b}_2|e^{i\left(\frac{\theta_2-\theta_1}{2}+\theta_0\right)} \end{bmatrix} = \begin{bmatrix} |\bar{b}_1|e^{i\theta_1} \\ |\bar{b}_2|e^{i\theta_2} \end{bmatrix}. \tag{101}$$

Appendix C
Illustration of the Quantum Computation Procedure for Preparing $|\bar{b}\rangle$: Two-Qubit Case

In this section, we illustrate the quantum computation procedure of for preparing the state $|\bar{b}\rangle$ in the two-qubit case. In this case, we have

$$\boldsymbol{b} = \begin{bmatrix} |b_1|e^{i\theta_1} \\ |b_2|e^{i\theta_2} \\ |b_3|e^{i\theta_3} \\ |b_4|e^{i\theta_4} \end{bmatrix}. \tag{102}$$

The corresponding quantum state for the normalized vector $\bar{\boldsymbol{b}}$ is mathematically represented as follows:

$$|\bar{b}\rangle \triangleq \begin{bmatrix} |\bar{b}_1|e^{i\theta_1} \\ |\bar{b}_2|e^{i\theta_2} \\ |\bar{b}_3|e^{i\theta_3} \\ |\bar{b}_4|e^{i\theta_4} \end{bmatrix}. \tag{103}$$

We first focus on the magnitude preparation circuit, as shown in Fig. 13. According to (50)–(54), we can find the values of the phase parameters, Ψ_1, Φ_1, and Φ_2, as follows:

$$\Psi_1 = \arctan\sqrt{\frac{|b_3|^2 + |b_4|^2}{|b_1|^2 + |b_2|^2}} \tag{104}$$

$$\Phi_1 = \arctan\frac{|b_2|}{|b_1|} \tag{105}$$

$$\Phi_2 = \arctan\frac{|b_4|}{|b_3|}. \tag{106}$$

Taking Ψ_1, Φ_1, and Φ_2 as the phase parameter θ and substituting (104)–(106) into (10) allow us to derive the states

$|\varphi_2\rangle$ and $|\varphi_3\rangle$ as follows:

$$|\varphi_2\rangle \triangleq \begin{bmatrix} \cos\Psi_1 \cos\Phi_1 \\ \cos\Psi_1 \sin\Phi_1 \\ \sin\Psi_1 \\ 0 \end{bmatrix} \tag{107}$$

$$|\varphi_3\rangle \triangleq \begin{bmatrix} \cos\Psi_1 \cos\Phi_1 \\ \cos\Psi_1 \sin\Phi_1 \\ \sin\Psi_1 \cos\Phi_2 \\ \sin\Psi_1 \sin\Phi_2 \end{bmatrix}. \tag{108}$$

Applying the sine and cosine laws based on (104)–(106) and substituting the results into (108), we can find that the state $|\varphi_3\rangle$ is equivalent to the state $|\bar{b}\rangle$ with correct magnitude and zero phase, i.e.,

$$|\varphi_3\rangle \triangleq \begin{bmatrix} |\bar{b}_1| \\ |\bar{b}_2| \\ |\bar{b}_3| \\ |\bar{b}_4| \end{bmatrix}. \tag{109}$$

Now, let us move on to the phase preparation circuit, as shown in Fig. 14, where the phase parameters, $\bar{\Psi}_1$, $\bar{\Phi}_1$, and $\bar{\Phi}_2$, are given by

$$\bar{\Psi}_1 = \frac{1}{4}[(\theta_3 + \theta_4) - (\theta_1 + \theta_2)] \tag{110}$$

$$\bar{\Phi}_1 = \frac{1}{2}(\theta_2 - \theta_1) \tag{111}$$

$$\bar{\Phi}_2 = \frac{1}{2}(\theta_4 - \theta_3). \tag{112}$$

Taking $\bar{\Psi}_1$, $\bar{\Phi}_1$, and $\bar{\Phi}_2$ as the phase parameter θ and substituting (110)–(112) into (11) allow us to derive the states $|\varphi_4\rangle$, $|\varphi_5\rangle$, and $|\varphi_6\rangle$ as follows:

$$|\varphi_4\rangle \triangleq \begin{bmatrix} |\bar{b}_1| e^{-i\bar{\Psi}_1} \\ |\bar{b}_2| e^{-i\bar{\Psi}_1} \\ |\bar{b}_3| e^{i\bar{\Psi}_1} \\ |\bar{b}_4| e^{i\bar{\Psi}_1} \end{bmatrix} \tag{113}$$

$$|\varphi_5\rangle \triangleq \begin{bmatrix} |\bar{b}_1| e^{-i(\bar{\Psi}_1+\bar{\Phi}_1)} \\ |\bar{b}_2| e^{-i(\bar{\Psi}_1-\bar{\Phi}_1)} \\ |\bar{b}_3| e^{i\bar{\Psi}_1} \\ |\bar{b}_4| e^{i\bar{\Psi}_1} \end{bmatrix} \tag{114}$$

$$|\varphi_6\rangle \triangleq \begin{bmatrix} |\bar{b}_1| e^{-i(\bar{\Psi}_1+\bar{\Phi}_1)} \\ |\bar{b}_2| e^{-i(\bar{\Psi}_1-\bar{\Phi}_1)} \\ |\bar{b}_3| e^{i(\bar{\Psi}_1-\bar{\Phi}_2)} \\ |\bar{b}_4| e^{i(\bar{\Psi}_1+\bar{\Phi}_2)} \end{bmatrix}. \tag{115}$$

We define a global phase θ_0 as follows:

$$\theta_0 = \frac{1}{4}(\theta_1 + \theta_2 + \theta_3 + \theta_4). \tag{116}$$

Given this definition, the state $|\varphi_6\rangle$ in (115) is equivalent to

$$|\varphi_6\rangle \sim \begin{bmatrix} |\bar{b}_1| e^{i(\theta_0-\bar{\Psi}_1-\bar{\Phi}_1)} \\ |\bar{b}_2| e^{i(\theta_0-\bar{\Psi}_1+\bar{\Phi}_1)} \\ |\bar{b}_3| e^{i(\theta_0+\bar{\Psi}_1-\bar{\Phi}_2)} \\ |\bar{b}_4| e^{i(\theta_0+\bar{\Psi}_1+\bar{\Phi}_2)} \end{bmatrix}. \tag{117}$$

Substituting (110)–(112) and (116) into (117), we can find that $|\varphi_6\rangle$ is equal to our desired state $|\bar{b}\rangle$ up to θ_0

$$|\varphi_6\rangle \sim \begin{bmatrix} |\bar{b}_1| e^{i\theta_1} \\ |\bar{b}_2| e^{i\theta_2} \\ |\bar{b}_3| e^{i\theta_3} \\ |\bar{b}_4| e^{i\theta_4} \end{bmatrix}. \tag{118}$$

REFERENCES

[1] J. Zhang, F. Feng, and Q. J. Zhang, "Quantum method for finite element simulation of electromagnetic problems," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Atlanta, GA, Jun. 2021, pp. 120–123.

[2] J. D. Hidary, *Quantum Computing: An Applied Approach*. Berlin, Germany: Springer, 2019.

[3] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[4] S. Sinha and P. Russer, "Quantum computing algorithm for electromagnetic field simulation," *Quantum Inf. Process.*, vol. 9, no. 3, pp. 385–404, Jun. 2010.

[5] J. A. Russer, M. Haider, C. Jirauschek, and P. Russer, "On the possibility of quantum simulation of electromagnetic structures," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Boston, MA, USA, Jun. 2019, pp. 267–270.

[6] J. M. Jin, *The Finite Element Method in Electromagnetics*. New York, NY, USA: Wiley, 2002.

[7] C. Phillips and V. I. Okhmatovski, "Quantum algorithms for the solution of matrix equations in electromagnetics," in *Proc. Int. Appl. Comput. Electromagn. Soc. Symp. (ACES)*, Aug. 2021, pp. 1–3.

[8] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum algorithm for linear systems of equations," *Phys. Rev. Lett.*, vol. 103, no. 15, Oct. 2009, Art. no. 150502.

[9] D. Dervovic, M. Herbster, P. Mountney, S. Severini, N. Usher, and L. Wossnig, "Quantum linear systems algorithms: A primer," 2018, *arXiv:1802.08227*.

[10] B. Duan, J. Yuan, C. H. Yu, J. Huang, and C. Y. Hsieh, "A survey on HHL algorithm: From theory to application in quantum machine learning," *Phys. Lett. A*, vol. 384, no. 24, May 2020, Art. no. 126595.

[11] D. W. Berry, "High-order quantum algorithm for solving linear differential equations," *J. Phys. A, Math. Gen.*, vol. 47, no. 10, Feb. 2014, Art. no. 105301.

[12] N. Wiebe, D. Braun, and S. Lloyd, "Quantum algorithm for data fitting," *Phys. Rev. Lett.*, vol. 109, no. 5, Aug. 2012, Art. no. 050505.

[13] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Phys. Rev. Lett.*, vol. 113, no. 13, Sep. 2014, Art. no. 130503.

[14] I. Kerenidis and A. Prakash, "Quantum recommendation systems," 2016, *arXiv:1603.08675*.

[15] B. D. Clader, B. C. Jacobs, and C. R. Sprouse, "Preconditioned quantum linear system algorithm," *Phys. Rev. Lett.*, vol. 110, no. 25, Jun. 2013, Art. no. 250504.

[16] M. Mottonen, J. J. Vartiainen, V. Bergholm, and M. M. Salomaa, "Transformation of quantum states using uniformly controlled rotations," *Quantum Inf. Comput.*, vol. 5, no. 6, pp. 467–473, Sep. 2005.

[17] V. V. Shende, S. S. Bullock, and I. L. Markov, "Synthesis of quantum-logic circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 6, pp. 1000–1010, Jun. 2006.

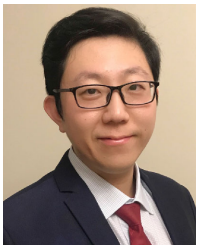
[18] G. W. Stewart, "A Krylov–Schur algorithm for large eigenproblems," *SIAM J. Matrix Anal. Appl.*, vol. 23, no. 3, pp. 601–614, Jan. 2002.

[19] J. Trommler, S. Koch, and T. Weiland, "A finite-element approach in order to avoid ill-conditioning in thin-sheet problems in frequency domain—Application to magneto-quasistatics," *J. Comput. Appl. Math.*, vol. 236, no. 18, pp. 4671–4680, Dec. 2012.

[20] S. Xinqing and P. Zhen, "Novel high-performance element in the electromagnetic finite-element method—Node-edge element," *J. Syst. Eng. Electron.*, vol. 19, no. 5, pp. 878–881, Oct. 2008.
[21] M. Benzi, "Preconditioning techniques for large linear systems: A survey," *J. Comput. Phys.*, vol. 182, no. 2, pp. 418–477, Nov. 2002.
[22] J. L. Volakis, A. Chatterjee, and L. C. Kempel, *Finite Element Method for Electromagnetics: Antennas, Microwave Circuits, and Scattering Applications*. New York, NY, USA: IEEE Press, 1998.
[23] M. J. Grote and T. Huckle, "Parallel preconditioning with sparse approximate inverses," *SIAM J. Sci. Comput.*, vol. 18, no. 3, pp. 838–853, May 1997.
[24] E. Chow and Y. Saad, "Approximate inverse preconditioners via sparse-sparse iterations," *SIAM J. Sci. Comput.*, vol. 19, no. 3, pp. 995–1023, May 1998.
[25] E. Chow, "A priori sparsity patterns for parallel sparse approximate inverse preconditioners," *SIAM J. Sci. Comput.*, vol. 21, no. 5, pp. 1804–1822, Jan. 2000.
[26] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, "Efficient quantum algorithms for simulating sparse Hamiltonians," *Commun. Math. Phys.*, vol. 270, no. 2, pp. 359–371, Mar. 2007.
[27] K. J. Sung et al., "Using models to improve optimizers for variational quantum algorithms," 2020, *arXiv:2005.11011*.
[28] Cirq Developers. (2018). *Cirq Documentation*. [Online]. Available: https://cirq.readthedocs.io/en/latest/tutorial.html
[29] S. Endo, Z. Cai, S. C. Benjamin, and X. Yuan, "Hybrid quantum-classical algorithms and quantum error mitigation," *J. Phys. Soc. Jpn.*, vol. 90, no. 3, Mar. 2021, Art. no. 032001.
[30] A. Mari, N. Shammah, and W. J. Zeng, "Extending quantum probabilistic error cancellation by noise scaling," *Phys. Rev. A, Gen. Phys.*, vol. 104, no. 5, Nov. 2021, Art. no. 052607.
[31] R. Takagi, "Optimal resource cost for error mitigation," *Phys. Rev. Res.*, vol. 3, no. 3, Aug. 2021, Art. no. 033178.
[32] Z. Cai et al., "Quantum error mitigation," 2022, *arXiv:2210.00921*.
[33] R. Takagi, H. Tajima, and M. Gu, "Universal sampling lower bounds for quantum error mitigation," 2022, *arXiv:2208.09178*.
[34] S. Aaronson, "Read the fine print," *Nature Phys.*, vol. 11, no. 4, pp. 291–293, Apr. 2015.
[35] K. Bharti et al., "Noisy intermediate-scale quantum algorithms," *Rev. Mod. Phys.*, vol. 94, Feb. 2022, Art. no. 015004.
[36] M. Cerezo et al., "Variational quantum algorithms," *Nature Rev. Phys.*, vol. 3, no. 9, pp. 625–644, 2021.
[37] W.-B. Ewe, D. E. Koh, S. T. Goh, H.-S. Chu, and C. E. Png, "Variational quantum-based simulation of waveguide modes," *IEEE Trans. Microw. Theory Techn.*, vol. 70, no. 5, pp. 2517–2525, May 2022.
[38] L. Zhang and Q. J. Zhang, "Quantum method for scaling the finite element-based quantum solutions of electromagnetic problems," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Denver, CO, Jun. 2022, pp. 321–324.
[39] J. S. Savage and A. F. Peterson, "Higher-order vector finite elements for tetrahedral cells," *IEEE Trans. Microw. Theory Techn.*, vol. 44, no. 6, pp. 874–879, Jun. 1996.
[40] A. J. Otto, N. Marais, E. Lezar, and D. B. Davidson, "Using the FEniCS package for FEM solutions in electromagnetics," *IEEE Antennas Propag. Mag.*, vol. 54, no. 4, pp. 206–223, Aug. 2012.

Jianan Zhang (Member, IEEE) received the B.Eng. degree from Tianjin University, Tianjin, China, in 2013, and the Ph.D. degree from the School of Microelectronics, Tianjin University, and the Department of Electronics, Carleton University, Ottawa, ON, Canada, in 2020.

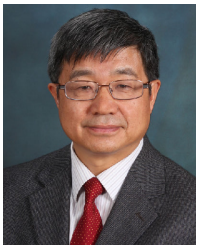
From 2020 to 2022, he was a Post-Doctoral Research Associate with the Department of Electronics, Carleton University. He is currently an Associate Professor with the State Key Laboratory of Millimeter Waves, Southeast University, Nanjing, China. His research interests include machine-learning approaches to metasurface design, surrogate modeling and surrogate-assisted optimization, finite element analysis in electromagnetic (EM), and quantum computing with applications to EM problems.



Feng Feng (Senior Member, IEEE) received the B.Eng. degree from Tianjin University, Tianjin, China, in 2012, and the Ph.D. degree from the School of Microelectronics, Tianjin University, and the Department of Electronics, Carleton University, Ottawa, ON, Canada, in 2017.

From 2017 to 2020, he was a Post-Doctoral Fellow with the Department of Electronics, Carleton University. In 2020, he joined the School of Microelectronics, Tianjin University, where he is currently a Full Professor. He has authored or coauthored over 100 publications. His research interests include artificial intelligence (AI) and machine learning-based electromagnetic parametric modeling and optimization methods for high-speed/high-frequency circuit design.

Dr. Feng is a member of the Technical Committee on Design Automation (TC-2) of the IEEE Microwave Theory and Techniques (MTT) Society.



Qi-Jun Zhang (Fellow, IEEE) received the B.Eng. degree from the Nanjing University of Science and Technology, Nanjing, China, in 1982, and the Ph.D. degree in electrical engineering from McMaster University, Hamilton, ON, Canada, in 1987.

From 1988 to 1990, he was a Research Engineer with Optimization Systems Associates Inc., Dundas, ON, Canada, developing advanced optimization software for microwave modeling and design. In 1990, he joined the Department of Electronics, Carleton University, Ottawa, ON, Canada, where he is currently a Chancellor's Professor. He has authored the book *Neural Networks for RF and Microwave Design* (Boston, MA, USA: Artech House, 2000) and is a Coeditor of the book *Modeling and Simulation of High-Speed VLSI Interconnects* (Boston: Kluwer, 1994) and *Simulation-Driven Design Optimization and Modeling for Microwave Engineering* (London, U.K.: Imperial College Press, 2013). He has more than 360 publications in his research area. His research interests include modeling, optimization, and machine learning for high-speed/high-frequency electronic design.

Dr. Zhang is a Fellow of the Canadian Academy of Engineering and the Engineering Institute of Canada. He is the Co-Chair of the Working Group on Artificial Intelligence (AI) and Machine Learning-Based Technologies for Microwaves in the Future Directions Committee of the IEEE Microwave Theory and Techniques (MTT) Society. He was twice a Guest Editor of the Special Issues on Applications of Artificial Neural Network (ANN) for RF/Microwave Design for the *International Journal of RF/Microwave Computer-Aided Engineering* in 1999 and 2002, and a Guest Coeditor of the Special Issue on Machine Learning in Microwave Engineering for the *IEEE Microwave Magazine* in 2021. He is a Topic Editor of the IEEE JOURNAL OF MICROWAVES.