Tridiagonal matrix decomposition for Hamiltonian simulation on a quantum computer

Boris Arseniev, Dmitry Guskov, Richik Sengupta, Jacob Biamonte,* and Igor Zacharov
Skolkovo Institute of Science and Technology, Moscow 121205, Russian Federation

(Received 23 October 2023; revised 26 March 2024; accepted 25 April 2024; published 29 May 2024)

The construction of quantum circuits to simulate Hamiltonian evolution is central to many quantum algorithms. State-of-the-art circuits are based on oracles whose implementation is often omitted, and the complexity of the algorithm is estimated by counting oracle queries. However, in practical applications, an oracle implementation contributes a large constant factor to the overall complexity of the algorithm. The key finding of this work is the efficient procedure for representation of a tridiagonal matrix in the Pauli basis, which allows one to construct a Hamiltonian evolution circuit without the use of oracles. The procedure represents a general tridiagonal matrix $2^n \times 2^n$ by systematically determining all Pauli strings present in the decomposition, dividing them into commuting subsets. The efficiency is in the number of commuting subsets $O(n)$. The method is demonstrated using the one-dimensional wave equation, verifying numerically that the gate complexity as a function of the number of qubits is lower than the oracle-based approach for $n < 15$ and requires half the number of qubits. This method is applicable to other Hamiltonians based on the tridiagonal matrices.

DOI: 10.1103/PhysRevA.109.052629

I. INTRODUCTION

Simulation of quantum Hamiltonians is one of the first directions that can potentially demonstrate quantum advantage [1]. Over the years this field has evolved, and great progress has been made since the first description [2]. The latest developed techniques, such as the quantum walk algorithm [3], simulation by qubitization [4], and others [5–8], are often described in terms of calls to an oracle, but the construction of the oracle is usually not specified. In a notable exception from the above, the oracle is constructed in Ref. [9] and the constant factor in gate scalability is calculated.

The standard approach to implement a circuit for Hamiltonian simulation on a quantum computer is to decompose the Hamiltonian H into a sum of Pauli strings (tensor product of Pauli matrices) and approximate the operator e^{-iHt} with product formulas [10,11]. With a naive approach, the number of Pauli strings to consider is 4^n if the size of H is $2^n \times 2^n$. It was conjectured that modeling e^{-iHt} using this approach is more efficient if one can group the resulting Pauli strings into commuting sets [12–14]. It was shown that the set of all Pauli operators (without identity) of size $4^n - 1$ can be divided into $2^n + 1$ different subsets, each consisting of $2^n - 1$ internally commuting elements [15].

The problem of partitioning a Hamiltonian decomposition in the Pauli basis featuring sets of commuting operators was studied in the framework of simultaneous measurements [16]. Typically, this problem may be solved by building a graph with Pauli strings as nodes, connected if they commute, i.e., the clique problem. Further, it can be reduced to the graph-coloring problem which is NP-complete, but heuristics exist [17].

In this work we consider Hamiltonians of a special kind which are constructed using tridiagonal matrices. Tridiagonal matrices come to light in many different areas of mathematical and applied sciences, commonly in the discretization of differential equations [18,19], and are used to represent discretized versions of differential operators in quantum computing.

The proposed procedure decomposes tridiagonal matrices into $O(n)$ internally commuting subsets of Pauli strings, each subset having size $O(2^n)$. It also provides the coefficients (weights) for each Pauli string in the decomposition. It automatically leverages the structure of the tridiagonal matrix to remove the majority of the redundant Pauli strings with zero coefficients and provides an upper bound for the number of Pauli strings with nonzero weights. Moreover, it contains a formula for calculation of the weights separate from the symbolic generation of Pauli strings.

We illustrate our method using the Hamiltonian of the one-dimensional wave equation as an example and numerically show the dependence of the number of gates on the number of qubits. We also show that our method for $n < 15$ qubits has fewer gates for practical applications than one with the oracle implementation, despite worse theoretical scaling.

This work is organized as follows. In Sec. II we introduce the notation and useful mathematical constructs. The decomposition algorithm for an arbitrary tridiagonal matrix is described in Sec. III, followed by specific variants for real and real symmetric tridiagonal matrices. Section IV is a special case of a symmetrized matrix H constructed from a real matrix B such that both B and $B^\top$ are on the antidiagonal. In Sec. V we focus on Hamiltonian simulation, while in Sec. VI the method is illustrated with an example of the one-dimensional wave equation. We defer longer proofs to the Appendix.

*No longer affiliated with Skolkovo Institute of Science and Technology.

TABLE I. Notational convention. Here, $\mathbf{x} = (x_1, \dots, x_n) \in \mathbb{B}^n$, $\mathbf{y} = (y_1, \dots, y_n) \in \mathbb{B}^n$, and X and Z are Pauli matrices.

Notation	Definition
$\overline{\mathbf{x}}$	Negation, $\overline{\mathbf{x}} = (\overline{x}_1, \dots, \overline{x}_n)$
$\mathbf{x}^{\mathbf{y}}$	Exponentiation, $\mathbf{x}^{\mathbf{y}} = (x_1^{y_1}, \dots, x_n^{y_n})$
$\mathbf{x} \cdot \mathbf{y}$	Inner product, $\mathbf{x} \cdot \mathbf{y} = \sum_{l=1}^{n} x_l y_l$
$\delta(\mathbf{x}, \mathbf{y})$	Kronecker delta, $\delta(\mathbf{x}, \mathbf{y}) = \prod_{j=1}^{n} \delta_{x_j, y_j}$
$Z^{\mathbf{y}}$	$Z^{\mathbf{y}} \equiv \bigotimes_{l=1}^{n} Z^{y_l} = Z^{y_1} \otimes \dots \otimes Z^{y_n}$

II. NOTATION AND DEFINITIONS

The Pauli matrices constitute a basis for the complex vector space of 2×2 matrices and comprise operators $\mathcal{P} = \{I, X, Y, Z\}$, where

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

A tensor product of several Pauli matrices is called a Pauli string. The length of a Pauli string is the number of Pauli operators in the string, and it is exactly n when decomposing a $2^n \times 2^n$ matrix. We denote the set of Pauli strings of length n as $\mathcal{P}_n$.

The Pauli basis decomposition of an arbitrary matrix B is given by

$$B = \frac{1}{2^n} \sum_{j=1}^{M} \alpha_j P_j, \quad P_j \in \mathcal{P}_n, \tag{1}$$

where M is the number of terms in the decomposition and $\alpha_j \in \mathbb{C}$ in general. Hereinafter we omit the tensor sign when writing Pauli strings (e.g., $X \otimes Y \otimes Z \otimes Y$ is abbreviated as $XYZY$).

Manipulation of Pauli strings is possible with bit arithmetic. For a single bit $x, p \in \mathbb{B} = \{0, 1\}$ we use the following notation for powers:

$$x^p = x \oplus \overline{p} = x \oplus p \oplus 1,$$

where $\oplus$ is XOR (addition modulo 2). Definitions for strings of bits $\mathbf{x} \equiv (x_1, \dots, x_n) \in \mathbb{B}^n$, where $x_j \in \mathbb{B}$, $j = 1, \dots, n$, are summarized in Table I.

For $\mathbf{x} \in \mathbb{B}^n$ we define the vector $|\mathbf{x}\rangle \in (\mathbb{C}^2)^{\otimes n}$ as

$$|\mathbf{x}\rangle = \bigotimes_{l=1}^{n} |x_l\rangle = |x_1, \dots x_n\rangle. \tag{2}$$

Further, we define the function that converts non-negative integers to binary

$$\mathrm{BIN} : \mathbb{N} \cup \{0\} \to \mathbb{B}^n. \tag{3}$$

Note that BIN as well as the bit string $\mathbf{x}$ are encodings of an integer, where the leftmost bit encodes the lower register. For example, the string 1101 is the encoding of $1 \times 1 + 1 \times 2 + 0 \times 4 + 1 \times 8 = 11$.

To express Pauli strings with X and Z matrices we use bit strings $\mathbf{z}, \mathbf{x}, \mathbf{p} \in \mathbb{B}^n$ and definitions in Table I,

$$Z^{\mathbf{z}}|\mathbf{p}\rangle = \bigotimes_{l=1}^{n} Z^{z_l}|p_l\rangle = (-1)^{\mathbf{z}\cdot\mathbf{p}}|\mathbf{p}\rangle, \tag{4}$$

$$X^{\mathbf{x}}|\mathbf{p}\rangle = \bigotimes_{l=1}^{n} X^{x_l}|p_l\rangle = |\overline{\mathbf{p}}^{\mathbf{x}}\rangle. \tag{5}$$

An arbitrary Pauli string can be defined as the image of the extended Pauli string operator (Walsh function) $\hat{W} : \mathbb{B}^n \times \mathbb{B}^n \to \mathcal{P}_n$ as follows:

$$\hat{W}(\mathbf{x}, \mathbf{z}) = i^{\mathbf{x}\cdot\mathbf{z}} X^{\mathbf{x}} Z^{\mathbf{z}} = \bigotimes_{j=1}^{n} i^{x_j z_j} X^{x_j} Z^{z_j}, \tag{6}$$

with the ordinary matrix product between $X^{\mathbf{x}}$ and $Z^{\mathbf{z}}$. It can be seen that the Walsh function is bijective. Thus, each Pauli string can be encoded with a unique pair $(\mathbf{x}, \mathbf{z})$ and (1) can be rewritten as

$$B = \frac{1}{2^n} \sum_{\mathbf{x},\mathbf{z} \in \mathbb{B}^n} \beta_{\mathbf{x},\mathbf{z}} \hat{W}(\mathbf{x}, \mathbf{z}), \tag{7}$$

where $\beta_{\mathbf{x},\mathbf{z}} \in \mathbb{C}$. There is one-to-one correspondence between α_j from (1) and $\beta_{\mathbf{x},\mathbf{z}}$ in (7).

By $\{P_1, P_2\}^{\otimes n}$ we denote the nth Cartesian product of the set $\{P_1, P_2\}$ with elements interpreted as Pauli strings. For example, $\{P_1, P_2\}^{\otimes 2} \equiv \{P_1, P_2\} \otimes \{P_1, P_2\} \equiv \{P_1 \otimes P_1, P_1 \otimes P_2, P_2 \otimes P_1, P_2 \otimes P_2\}$. As before, the product $P_k \otimes P_j$ is abbreviated $P_k P_j$.

III. TRIDIAGONAL MATRIX DECOMPOSITIONS

We consider an arbitrary tridiagonal matrix $B \in \mathbb{C}^{N \times N}$, where $N = 2^n$ of the following form:

$$B = \begin{pmatrix} c_1 & a_1 & 0 & \dots & 0 & 0 & 0 \\ b_1 & c_2 & a_2 & \dots & 0 & 0 & 0 \\ 0 & b_2 & c_3 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & c_{N-2} & a_{N-2} & 0 \\ 0 & 0 & 0 & \dots & b_{N-2} & c_{N-1} & a_{N-1} \\ 0 & 0 & 0 & \dots & 0 & b_{N-1} & c_N \end{pmatrix}. \tag{8}$$

Proposition 1 provides the maximal possible set of Pauli strings with nonzero coefficients in the decomposition of an arbitrary tridiagonal matrix with complex entries. We limit our consideration to tridiagonal matrices with only real entries (proposition 3), and further, we consider only tridiagonal symmetric real matrices ($a_i = b_i$) in corollary 2 and provide the maximal possible set of Pauli strings with nonzero coefficients in each case, as well as provide a possible partitioning of these strings into sets of internally commuting operators.

A. Pauli strings present in the decomposition

We formulate the following proposition regarding the decomposition of an arbitrary tridiagonal matrix B with complex entries shown in (8) into Pauli strings:

Proposition 1 (Decomposition of an arbitrary tridiagonal matrix). An arbitrary tridiagonal matrix $B \in \mathbb{C}^{N \times N}$, where $N \equiv 2^n$, can have Pauli strings in its decomposition with nonzero coefficients only from the union of the following disjoint sets with total cardinality of $(n+1)2^n$:

$$\begin{array}{lccccccccccc}
0. & \{I,Z\} & \otimes & \{I,Z\} & \otimes & \cdots & \otimes & \{I,Z\} & \otimes & \{I,Z\} & \otimes & \{I,Z\} \\
1. & \{I,Z\} & \otimes & \{I,Z\} & \otimes & \cdots & \otimes & \{I,Z\} & \otimes & \{I,Z\} & \otimes & \{X,Y\} \\
2. & \{I,Z\} & \otimes & \{I,Z\} & \otimes & \cdots & \otimes & \{I,Z\} & \otimes & \{X,Y\} & \otimes & \{X,Y\} \\
3. & \{I,Z\} & \otimes & \{I,Z\} & \otimes & \cdots & \otimes & \{X,Y\} & \otimes & \{X,Y\} & \otimes & \{X,Y\} \\
\vdots & & & & & & & & & & & \\
n-1. & \{I,Z\} & \otimes & \{X,Y\} & \otimes & \cdots & \otimes & \{X,Y\} & \otimes & \{X,Y\} & \otimes & \{X,Y\} \\
n. & \multicolumn{11}{c}{\underbrace{\{X,Y\} \otimes \{X,Y\} \otimes \cdots \otimes \{X,Y\} \otimes \{X,Y\} \otimes \{X,Y\}}_{n}}
\end{array}$$

The first step in the procedure for decomposition of a tridiagonal matrix in the Pauli basis (1) consists of symbolic generation of Pauli strings starting from all diagonal $\{I, Z\}$ operators to all antidiagonal operators $\{X, Y\}$ by replacing one diagonal operator on the right with an antidiagonal operator at each step m. The decomposition weights can be calculated later, based on the selected Pauli strings, see Sec. III B. Note that the cardinality of the union in proposition 1 is based on the structure of an arbitrary tridiagonal matrix; weight calculation may result in fewer Pauli strings in the decomposition.

We denote the sets in proposition 1 as

$$S^n_{m,\pm} = \{I,\ Z\}^{\otimes(n-m)} \otimes \{X,Y\}^{\otimes m}, \quad m = 0,\dots,n. \tag{9}$$

When $m = n$ ($m = 0$) the first (second) tensor product is omitted. Each step m generates 2^n Pauli strings, and we divide these strings into two sets named $S_{m,+}$ and $S_{m,-}$, where $+$ indicates that the number of Y operators in every Pauli string in the set is even and $-$ indicates that this number is odd. When $m = 0$, there are no Y operators, so we will denote it as S_0. Note that each m corresponds to a row labeled by m in proposition 1.

The bit string notation from Sec. II provides a concise description of sets $S^n_{m,\pm}$ where the correspondence to m is given by bit string $\mathbf{x}$ as follows:

$$\begin{aligned}
S^n_{m,+} &= \{\hat{W}(\mathbf{x},\mathbf{z}) : \mathbf{x}\cdot\mathbf{z} = 0 \pmod 2,\ \mathbf{x} = V^n_m,\ \mathbf{z}\in\mathbb{B}^n\},\\
S^n_{m,-} &= \{\hat{W}(\mathbf{x},\mathbf{z}) : \mathbf{x}\cdot\mathbf{z} = 1 \pmod 2,\ \mathbf{x} = V^n_m,\ \mathbf{z}\in\mathbb{B}^n\},
\end{aligned} \tag{10}$$

where V^n_m is a selector with m bits set to one and the following $n-m$ bits set to zero, like $V^n_m = (\underbrace{1,\dots,1}_{m}, \underbrace{0,\dots,0}_{n-m})$, formally:

$$V^n_m = (v^m_1,\dots,v^m_n), \quad v^m_j = \begin{cases} 1, & m \geqslant j \\ 0, & m < j \end{cases}. \tag{11}$$

The bit string $\mathbf{x}$ corresponds to m such that the first m positions of the bit string set to one, and the remaining $(n-m)$ positions are all zeros. To generate each subset $S^n_{m,\pm}$, $\mathbf{z}$ traverses all numbers $0,\dots,n$ and the generated Pauli strings are sorted according to the outcome of $\mathbf{x}\cdot\mathbf{z}$.

Each of the subsets $S^n_{m,\pm}$ contains commuting Pauli strings due to the following proposition:

Proposition 2 (Commutativity criterion). Let $P = \hat{W}(\mathbf{x},\mathbf{z})$ and $Q = \hat{W}(\mathbf{a},\mathbf{b})$ be two Pauli strings of length n, where $\mathbf{x},\mathbf{z},\mathbf{a},\mathbf{b}\in\mathbb{B}^n$. Then, P and Q commute if

$$\mathbf{x}\cdot\mathbf{b} = \mathbf{a}\cdot\mathbf{z} \pmod 2. \tag{12}$$

Corollary 1. Let $P = \hat{W}(\mathbf{x},\mathbf{z})$ and $Q = \hat{W}(\mathbf{x},\mathbf{b})$ be two Pauli strings of length n, where $\mathbf{x},\mathbf{z},\mathbf{b}\in\mathbb{B}^n$. Let the parity of Y operators in both strings be equal, then P and Q commute.

Proposition 2 and corollary 1 are proven in Appendix A 2. We reach a conclusion that $S^n_{m,+}$ and $S^n_{m,-}$ each are internally commuting subsets. Therefore, the general matrix decomposition will have $2n+1$ internally commuting subsets.

Analogous to the general case, for a real tridiagonal matrix B the following proposition will hold:

Proposition 3 (Real tridiagonal matrix). A real tridiagonal matrix $B\in\mathbb{R}^{N\times N}$, where $N = 2^n$, can have Pauli strings in its decomposition with nonzero coefficients only from the union of $2n+1$ disjoint internally commuting sets $S_{m,\pm}$, $2n$ of which have a cardinality of 2^{n-1} each and one of which is given by $S_{0,\pm}$ and has a cardinality of 2^n. The cardinality of the union is $(n+1)2^n$.

For the case of a real symmetric tridiagonal matrix B the symmetry will enable additional cancellations in the decomposition, such that only $S_{m,+}$ may be present. This is reflected in the following corollary:

Corollary 2 (Real symmetric tridiagonal matrix). A real symmetric tridiagonal matrix $B\in\mathbb{R}^{N\times N}$, where $N = 2^n$, can have Pauli strings in its decomposition with nonzero coefficients only from the union of $n+1$ disjoint internally commuting sets, n of which are given by $S_{m,+}$ and have a cardinality of 2^{n-1} each and one of which is given by S_0 and has a cardinality of 2^n. The cardinality of the union is $(n+2)2^{n-1}$.

For convenience we omit the superscript n in $S^n_{m,+}$ and omit the $+$, when the length n and parity is clear from context.

B. Decomposition weights

In order to calculate the coefficients (weights) of the proposed decomposition, it is easier to separate the calculation of the weights for the "diagonal" and "off-diagonal (antidiagonal)" subsets. The matrix B can be written as a sum of the diagonal D and off-diagonal F matrices: $B = D + F$. The diagonal matrix D after decomposition consists of Pauli strings of length n in the subset S_0 containing only Z and I operators. The coefficients $\beta_{\mathbf{x},\mathbf{z}}$ of corresponding $\hat{W}(\mathbf{x},\mathbf{z})$ within this first internally commuting set can be calculated as

$$\beta_{\mathbf{0},\mathbf{z}} = \sum_{p=0}^{2^n-1} (-1)^{\mathbf{z}\cdot\mathrm{BIN}(p)} c_p, \tag{13}$$

where c_p is a diagonal element of the matrix B as in (8) and $\mathbf{0}$ is $(0,\dots,0)$, $\mathbf{z}\in\mathbb{B}^n$.

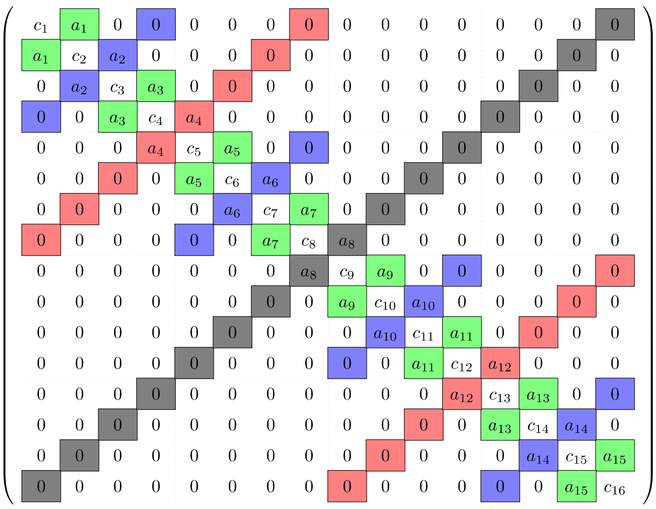

FIG. 1. Illustrating the contribution of commuting subsets $S_{m,+}$ to decomposition of the matrix B. Same colors contribute to the same commuting subset. Colors: S_1 – green, S_2 – blue, S_3 – red, S_4 – black.

For the off-diagonal matrix F, the weights $\beta_{\mathbf{x},\mathbf{z}}$ of the decomposition [see (7)] may be calculated for each $\hat{W}(\mathbf{x},\mathbf{z})$ in a subset $S_{m,\pm}$ given in (9) as follows:

$$\beta_{\mathbf{x},\mathbf{z}} = \sum_{p=0}^{2^n-2} i^{\mathbf{x}\cdot\mathbf{z}}(-1)^{\mathbf{z}\cdot\mathrm{BIN}(p)}\delta(\mathrm{BIN}(p+1), \overline{\mathrm{BIN}(p)}^{\mathbf{x}})a_p + \sum_{p=1}^{2^n-1} i^{\mathbf{x}\cdot\mathbf{z}}(-1)^{\mathbf{z}\cdot\mathrm{BIN}(p)}\delta(\mathrm{BIN}(p-1), \overline{\mathrm{BIN}(p)}^{\mathbf{x}})b_{p-1} \tag{14}$$

where $(\mathbf{x},\mathbf{z})$ corresponds to (10), and a_p and b_p are the off-diagonal elements of matrix B as in (8). The expression $\overline{\mathrm{BIN}(p)}^{\mathbf{x}}$ is calculated as the bit-wise XOR of bit string $\mathbf{x}$ with the inverted bit string $\mathbf{p}$.

Note that (14) holds only for the decomposition of off-diagonal elements, while (13) holds only for the diagonal elements of matrix B. These formulas are derived in Appendix A 1.

C. Visualization of commuting subsets

The elements of matrix B can be used to calculate the coefficients of Pauli strings in its decomposition using (14). In Fig. 1 we show the correspondence between elements of B and subsets S_m which contain the corresponding Pauli strings. The diagonal consists of Pauli strings containing only Z and I operators and corresponds to S_0. For the off-diagonal elements we are left with n subsets $S_{m,\pm}$ specified in (9). To show the correspondence, consider the off-diagonal component F of a real symmetric matrix B for $n = 4$. For a symmetric matrix (see corollary 2) we have the following subsets with size $2^{n-1} = 2^3$:

$$S_{1,+} = \{I, Z\}^{\otimes 3} \otimes \{X\},$$
$$S_{2,+} = \{I, Z\}^{\otimes 2} \otimes \{XX, YY\},$$
$$S_{3,+} = \{I, Z\} \otimes \{XXX, XYY, YXY, YYX\},$$
$$S_{4,+} = \{XXXX, XXYY, XYXY, XYYX, YXYX, YXXY, YYXX, YYYY\}.$$

Figure 1 illustrates how these subsets correspond to the elements of a $2^4 \times 2^4$ matrix B. Each $S_{m,+}$, $m = 1, \dots, 4$ is given a color and the structure is apparent. The length of the antidiagonal segment is equal to 2^m, where m is the number of the $\{X, Y\}$ Pauli operators on the right in the S_m expression.

The said structure not only appears for real symmetric matrices. It follows from the associativity of the tensor product, where we take the $n - m$ matrices from the set $\{I, Z\}$ on the left with m matrices from the set $\{X, Y\}$ on the right. For each Pauli string P from the subset $S_{m,\pm}$ we have

$$P = P_D \otimes P_A, \tag{15}$$

where P_D is a diagonal matrix of size $2^{n-m} \times 2^{n-m}$ and P_A is an antidiagonal matrix of size $2^m \times 2^m$.

IV. DECOMPOSITION FOR A SYMMETRIZED MATRIX

A Hermitian matrix H can be constructed from any matrix B as

$$H = \begin{pmatrix} 0 & B \\ B^\dagger & 0 \end{pmatrix}. \tag{16}$$

As before, consider the case where the matrix B is tridiagonal of size $2^n \times 2^n$, therefore the size of H is $2^{n+1} \times 2^{n+1}$. Here, we assume B to be real and therefore H is symmetric. The following statement holds:

Corollary 3. The number of terms in the decomposition of a symmetric matrix H (16) is equal to the number of terms in the decomposition of the real matrix B and is bounded above by $(n+1)2^n$. The Pauli strings in the decomposition of H can be partitioned into the following subsets of commuting strings:

$$S_0 = \{X\} \bigotimes_1^n \{I, Z\}, \quad (m = 0),$$
$$S_m = \{\widehat{X, Y}\} \bigotimes_1^{n-m} \{I, Z\} \bigotimes_1^m \{X, Y\}, \quad m = 1, \dots, n-1,$$
$$S_n = \{\widehat{X, Y}\} \bigotimes_1^m \{X, Y\}, \quad (m = n),$$

where the expression $\{\widehat{X, Y}\}$ selects X or Y such that the number of Y operators in each Pauli string is even.

The weights of the decomposition of H may be calculated using formulas (13) and (14). For any square matrix B with its Pauli decomposition given in (7) the following equality holds:

$$H = \frac{1}{2^n} \sum_{\mathbf{x},\mathbf{z} \in \mathbb{B}^n} (\mathrm{Re}\beta_{\mathbf{x},\mathbf{z}} X - i\mathrm{Im}\beta_{\mathbf{x},\mathbf{z}} XZ) \otimes \hat{W}(\mathbf{x},\mathbf{z}); \tag{17}$$

for details see Appendix A 3.

Consider a symmetric matrix H (16) consisting of B and $B^\top$ with $n = 4$ as an example. According to corollary 3, Pauli strings in the decomposition of H can be arranged into the

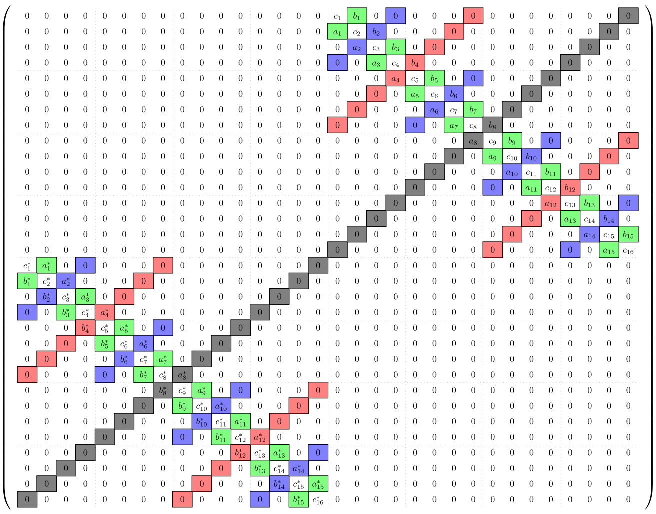

FIG. 2. Illustrating the contribution of commuting subsets S_m to decomposition of the matrix H of size $2^5 \times 2^5$ with B and $B^\dagger$ of size $2^4 \times 2^4$. Same colors contribute to the same commuting subset. Colors: S_1 – green, S_2 – blue, S_3 – red, S_4 – black. S_0 corresponds to the diagonal of each B matrix.

following sets:

$$S_0 = X \otimes \{I, Z\} \otimes \{I, Z\} \otimes \{I, Z\} \otimes \{I, Z\},$$
$$S_1 = \widehat{\{X, Y\}} \otimes \{I, Z\} \otimes \{I, Z\} \otimes \{I, Z\} \otimes \{X, Y\},$$
$$S_2 = \widehat{\{X, Y\}} \otimes \{I, Z\} \otimes \{I, Z\} \otimes \{X, Y\} \otimes \{X, Y\},$$
$$S_3 = \widehat{\{X, Y\}} \otimes \{I, Z\} \otimes \{X, Y\} \otimes \{X, Y\} \otimes \{X, Y\},$$
$$S_4 = \widehat{\{X, Y\}} \otimes \{X, Y\} \otimes \{X, Y\} \otimes \{X, Y\} \otimes \{X, Y\}.$$

These are similar to subsets which arise in the decomposition of a real symmetric matrix B, but now the size of each subset is 2^n. This follows from the fact that the terms with an odd number of Y operators, which were zero for the real symmetric case, now appear in the decomposition because the Y operator can be selected as the leading term to make the number of Y operators even (see corollary 2). Figure 2 shows which elements correspond to which set for the matrix H of size $2^5 \times 2^5$.

V. CIRCUIT FOR HAMILTONIAN SIMULATION

We have shown the two steps in matrix decomposition, namely, the generation of Pauli strings based on matrix structure and the calculation of decomposition weights. Importantly, Pauli strings are assembled into commuting sets. The information about commuting sets may serve to reduce the circuit complexity of Hamiltonian simulation [13] and to accelerate simulation of quantum dynamics on a classical computer [12]. In this section we use this approach to construct a circuit for Hamiltonian simulation.

The parameters that should be taken into account when implementing the evolution of the Hamiltonian [4] include the number of system qubits n, evolution time t, target error ϵ, and how information on the Hamiltonian H is accessed by the quantum computer. Our circuit does not require additional qubits, and it does not use oracles to access information. As we organized a large number of Pauli strings into an exponentially smaller number of commuting subsets, we could expect improvement in the accuracy according to the Trotterization formula.

TABLE II. Diagonalization operators D_k for a real symmetric tridiagonal matrix of size 2^4 consisting of the Hadamard operators H, the controlled-NOT $\mathrm{CX}(c, t)$, and controlled-Z $\mathrm{CZ}(c, t)$ operators, where c and t are the control and the target qubits, respectively. Qubits are labeled from 1 to 4.

D_k	Simultaneous diagonalization operators for subset S_k
D_1	$HHHI$ CX(1, 4) CX(2, 4) CX(3, 4) $HHHH$
D_2	$HHHI$ CX(1, 3) CX(1, 4) CX(2, 3) CX(2, 4) CX(3, 4) CZ(1, 4) CZ(2, 4) CZ(3, 4) $HHHH$
D_3	$HHHI$ CX(1, 3) CX(1, 4) CX(2, 4) CX(3, 4) CZ(1, 4) CZ(2, 4) CZ(3, 4) $HHHH$
D_4	$HHHI$ CX(1, 2) CX(1, 4) CX(2, 3) CZ(1, 3) CZ(3, 4) $HHHH$

The task of Hamiltonian simulation is to implement the operator e^{-iHt} on a quantum device. For an operator $\hat{H}$ represented by a tridiagonal or symmetrized (as discussed in Sec. IV) matrix H, the quantum circuit is constructed as follows.

First, we generate internally commuting sets of Pauli strings $\{S^n_{m,\pm}\}$ needed for the decomposition of H. These strings can be simultaneously diagonalized [12,13]. As an example, Table II contains the diagonalization operators D_k for the real symmetrical tridiagonal matrix of size 2^4 discussed in Sec. III C. Note that D_0 is an identity since the S_0 set is already diagonal. These procedures, i.e., the generation of the commuting sets of Pauli strings and optional simultaneous diagonalization, are determined by the tridiagonal matrix structure. These steps do not need to be repeated when the matrix changes values, making this approach suitable for use in variational algorithms [20,21].

When a quantum state evolves in time under the action of e^{-iHt} and the matrix H is represented in Pauli basis as in formula (1) [or, equivalently, as in (7)], the Lie-Trotter product formula [10] and its higher-order variants [11] should be implemented, taking the internally commuting sets into account. Given the time evolution governed by the Hermitian H and the number of Trotter repetitions r, using our decomposition we have

$$e^{-itH} = \exp\left[\frac{-it}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n}\beta_{\mathbf{x},\mathbf{z}}\hat{W}(\mathbf{x},\mathbf{z})\right] = \left[\prod_{k=0}^{2n}\exp\left(\frac{-it}{2^n r}\tilde{S}_k\right)\right]^r + \epsilon, \quad (18)$$

where $\tilde{S}_k = \sum_{\mathbf{z}}\beta_{\mathbf{x}_k,\mathbf{z}}\hat{W}(\mathbf{x}_k,\mathbf{z})$ are renumbered Pauli strings from subsets $S_{m,\pm}$ defined in (10), which contain up to 2^n terms each. The approximation is constructed with a Lie-Trotter product formula for $2n+1$ internally commuting sets

$S_{m,\pm}$ of Pauli strings, and the accuracy can be estimated as

$$\epsilon = \left\| e^{-itH} - \left[\prod_{k=0}^{2n} \exp\left(\frac{-it}{2^n r}\tilde{S}_k\right)\right]^r \right\| = O\left(\frac{t^2}{r}\right), \quad (19)$$

where $\|\cdot\|$ denotes the spectral norm. The expression (18) can be diagonalized as

$$\begin{aligned}\prod_{k=0}^{2n} &\exp\left(\frac{-it}{2^n r}\tilde{S}_k\right)\\ &= \prod_{k=0}^{2n} \exp\left[\frac{-it}{2^n r}\sum_{\mathbf{z}} \beta_{\mathbf{x}_k,\mathbf{z}}\hat{W}(\mathbf{x}_k,\mathbf{z})\right]\\ &= \prod_{k=0}^{2n} D_k^\dagger \exp\left(\frac{-it}{2^n r}\sum_{\mathbf{z}} \beta_{\mathbf{x}_k,\mathbf{z}}\Lambda_{k,\mathbf{z}}\right) D_k, \end{aligned} \quad (20)$$

where D_k is the diagonalization operator for each subset $\tilde{S}_k$. The number k fixes the string $\mathbf{x}_k$, and the inner summation over $\mathbf{z}$ covers the commuting subset as in (10). The operator $\Lambda_{k,\mathbf{z}}$ is a diagonal operator corresponding to $\hat{W}(\mathbf{x}_k, \mathbf{z})$. It is important to note that the coefficients β remain the same after diagonalization, or in other words, diagonalization depends only on the structure of the matrix (i.e., depends only on the Pauli subset) and does not depend on the values of the matrix elements. The computation of D_k is done by using Clifford algebras [12,13]. The D_k operators consist of the combinations of the single-qubit Hadamard operators and the two-qubit CX and CZ operators; the number of gates in D_k scales as $O(n^2)$, where n is the number of qubits.

It can be seen that in order to evaluate the propagator e^{-iHt} as in (18) and (20), one has to compute the coefficients $\beta_{\mathbf{x},\mathbf{z}}$ and implement $2n+1$ diagonal exponents $\Lambda_{k,\mathbf{z}}$. The coefficients can be computed by using formulas (13) and (14), and in order to implement $2n+1$ diagonal exponents, one can use the results from Ref. [22], where it is shown that the gate count for a circuit which implements the diagonal exponent can be reduced to $O(N')$ gates with $N' < 2^n$, without ancillary qubits.

Combining all the results together, we obtain gate complexity estimated as $O[r(2n+1)(2n^2+2^n)] = O(rn2^n)$, since for one Trotter step for each of $2n+1$ commuting sets one needs to implement D, $D^\dagger$, and diagonal exponents. When considering the Trotter formula of an arbitrary order p, each Trotter step will contain $O(5^{\lfloor p/2 \rfloor})$ times more gates, but the accuracy ϵ will be achieved in fewer steps r. In Ref. [11] the Trotterization error considered by leveraging information about commutation and the following scaling is given: $\epsilon = O(\frac{\alpha_{\text{comm}} t^{p+1}}{r^p})$ with $\alpha_{\text{comm}} = \sum_{j_1,\ldots,j_{p+1}}^{M} \|[H_{j_{p+1}},\ldots,[H_{j_2},H_{j_1}],\ldots]\|$ and $H = \sum_{j=1}^{M} H_j$, where H_j are anti-Hermitian (which is the case since simulation of e^{tH} is considered in Ref. [11]). It is an interesting question whether it is possible to obtain some tight upper bound for α_{comm}. For now, we will use scaling provided by the authors in Ref. [23], i.e., $\epsilon = O(\frac{(2M5^{\lfloor p/2 \rfloor -1}\|H\|t)^{p+1}}{r^p})$, where $H = \sum_{j=1}^{M} H_j$. Note that in this formula H can be considered as the sum of $2n+1$ matrices formed from commuting sets. Thus, $M = 2n+1$ and the resulting number of gates g is given by

$$g = O\left[tn^2 2^n 5^p \|H\| \left(\frac{tn\|H\|}{\epsilon}\right)^{1/p}\right]. \quad (21)$$

This scaling can be made more accurate by taking into account information about the commuting sets. This can be seen for the case where all Pauli strings commute, making the Trotter error equal to zero. It is possible to find some order of Pauli strings that will reduce the error, but since the number of all possible combinations grows exponentially, this is a difficult task [24].

VI. QUANTUM SIMULATION EXAMPLE: SOLVING THE WAVE EQUATION

Tridiagonal matrices arise when discretizing derivatives. For example, the solution of the heat equation $u_t(x,t) = [\kappa(x)u_x(x,t)]_x$ may be written as [19]

$$u(x,t) = e^{Bt}u(x,0),$$

where B is a real symmetric tridiagonal matrix as in Sec. III.

Another example is the wave equation, considered here in more detail. The wave equation in one dimension with amplitude $u(x,t)$ and speed $c(x)$ defined in the interval $x \in [0,1]$ is given by

$$u_{tt}(x,t) = [c^2(x)u_x(x,t)]_x, \quad u(x,t) \in \mathbb{R}. \quad (22)$$

We consider the case of the Dirichlet boundary conditions and set the initial conditions for u and u_x as follows:

$$\begin{aligned} u(0,t) = u(1,t) = 0, &\quad t \in \mathbb{R}_+,\\ u(x,0) = g(x), &\quad u_x(x,0) = 0. \end{aligned} \quad (23)$$

Following Costa *et al.* [25], we reduce (22) to the Schrödinger equation. Thus, consider the Hamiltonian in the following form (h is space discretization step):

$$H = \frac{1}{h}\begin{pmatrix} 0 & B \\ B^\dagger & 0 \end{pmatrix}, \quad (24)$$

which leads to the Schrödinger equation (we use natural units such that $\hbar = 1$) with a two-component quantum state $\psi = (\phi_V, \phi_E)^\top$,

$$\frac{d}{dt}\begin{pmatrix} \phi_V \\ \phi_E \end{pmatrix} = \frac{-i}{h}\begin{pmatrix} 0 & B \\ B^\dagger & 0 \end{pmatrix}\begin{pmatrix} \phi_V \\ \phi_E \end{pmatrix}.$$

This recovers the original (discretized) wave equation (22) if $-BB^\dagger = \mathcal{L}$, where $\mathcal{L}$ is the Laplacian giving an approximation of the second-order space derivative. To apply the method proposed in this work, we need the matrix B to be square and have the size $2^n \times 2^n$, so we slightly change the matrix B proposed by Costa *et al.* [25] by writing the Dirichlet boundary conditions explicitly, and we have also incorporated $c(x)$

into this matrix:

$$B_{N\times N} = \begin{pmatrix} 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & -c_2 & c_3 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & -c_3 & c_4 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -c_{N-2} & c_{N-1} & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & -c_{N-1} & c_N \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \end{pmatrix}.$$

The wave speed values c_k, $k = 1, \dots, N$ with $N = 2^n$ result from the discretization of the speed profile $c(x)$. The resulting Hamiltonian H has the form described in Sec. IV. Based on (17) the coefficients of the decomposition are given by

$$\beta_{\mathbf{0},\mathbf{z}} = -\sum_{k=1}^{2^n-2} (-1)^{\mathbf{z}\cdot \mathrm{BIN}(k)} c_{k+1},$$
$$\beta_{\mathbf{x},\mathbf{z}} = \sum_{k=1}^{2^n-2} i^{\mathbf{x}\cdot\mathbf{z}} (-1)^{\mathbf{z}\cdot \mathrm{BIN}(k)} \delta(\mathrm{BIN}(k+1), \overline{\mathrm{BIN}(k)}^{\mathbf{x}}) c_{k+2}, \tag{25}$$

where $\mathbf{x} = \mathbf{x}(m, n) = V_m^n$ as defined in (11).

Note that with our decomposition the expansion weights contain wave speeds explicitly. Therefore, this provides a method to solve partial differential equations with variable coefficients (piecewise constant over the discretization step dx), since decomposition can be done only once and weights recalculated.

We have implemented the solution for wave equation and use it with a constant speed $c = 1$ to determine the number of Trotter steps and the corresponding total number of gates needed to reach the set accuracy $\epsilon < 10^{-5}$ (evolution time is $t = 1$). This is shown in Fig. 3 as a function of the number of qubits n supporting discretization $N = 2^n$.

The gate complexity of our algorithm for solving the wave equation is calculated as

$$g_{1d} = O\left[5^p t n^2 N^2 \left(\frac{tnN}{\epsilon}\right)^{1/p}\right] \tag{26}$$

using $\|H\| \leqslant |\frac{4c_{\max}}{h}| = O(N)$, and for $p = 2$ is shown with a solid line in Fig. 3. The dashed lines in Fig. 3 represent manual approximation to experimental points with

$$g_a = \gamma N^\nu \log(N)^\mu = \gamma N^\nu n^\mu, \quad \text{with}$$
$$\nu = 1.5 + 1/p, \quad \mu = 2 + 1/p, \tag{27}$$

and the constant factor

$$\gamma = 25^{p/2-1} 10^{5/p}, \quad p = 2, 4,$$
$$\gamma = 5^{p/2-1} 10^{5/p}, \quad p = 6.$$

The actual number of gates g_a scales better than the theoretical one g_{1d} in formula (21). This is due to the factor

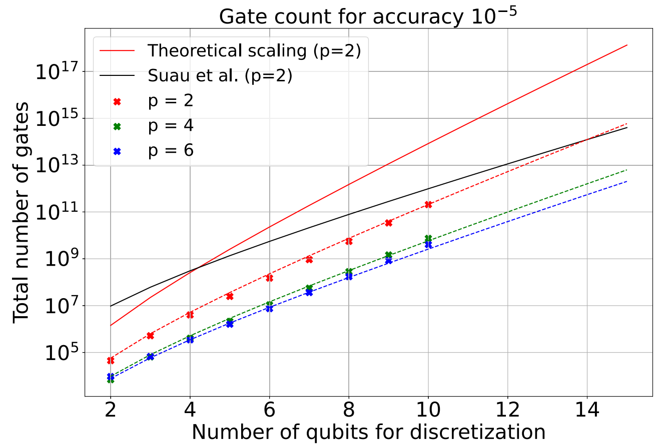

FIG. 3. Number of gates to approximate e^{-iHt} with accuracy $\epsilon = 10^{-5}$ for one-dimensional wave equation Hamiltonian (24). Points represent number of gates obtained in the simulation. Red solid line shows the theoretical gate scaling given by (26) for Trotter order $p = 2$. The black solid line shows number of gates from Ref. [9]. Dashed lines are fit to the simulation data with (27).

$\nu = 1.5 + 1/p < 2$ in the exponent of $2^n = N$, the number of the discretization points. The constant scaling factor is about ∼100, and the implementation is economical. Note that due to the two-component Hamiltonian used in the solution (24) of the Schrödinger equation, the actual number of qubits is $n + 1$.

Theoretical gate complexity for the approximation error ϵ using an oracle is calculated in Ref. [9] and is shown in Fig. 3 with a solid black line. In this reference the authors show oracle implementation of the algorithm presented in Refs. [3,25] with the gate complexity given by $O[5^p t n^2 N (\frac{tN}{\epsilon})^{1/p}]$. Here, the scaling factor is $\sim O(N^{1+1/p})$, which is better than ours [$\nu = 1.5 + 1/p$ in (27)] but with a constant factor of ∼300 000.

Therefore, when comparing our implementation with the oracle based for Trotter order $p = 2$, we obtain a smaller gate count for number of qubits $n < 15$. The oracle-based algorithm is also using $\geqslant 2n$ qubits to implement oracles and thus is less economical than our approach.

VII. CONCLUSION

We have presented an effective procedure for decomposition of a $N \times N$ tridiagonal matrix where $N = 2^n$ into $2n + 1$ subsets of commuting Pauli strings. Each of these subsets has 2^n Pauli strings of length n in a general case. Significant for applications is the decomposition of a Hermitian matrix consisting of two real tridiagonal matrices of the given type on the antidiagonal. For such a matrix there are $n + 1$ internally commuting subsets with 2^n Pauli strings of length n each, as shown in corollary 3. The suggested decomposition procedure considers only nonzero Pauli strings candidates, therefore improving on the brute-force method which examines all 4^{n+1} possible Pauli strings.

This advantage shows up in the calculation of the decomposition coefficients (weights, Sec. III B), because only the potentially nonzero Pauli strings participate in the evaluation. For the Hermitian matrix mentioned above there are $O(N \log N)$ binary multiplications in the evaluation of one expansion coefficient: we need to compute $O(1)$ products of bit strings with length $O(\log N)$ for each term [see (14)]. This compares favorably with the brute force approach using trace and matrix multiplications with complexity $O(N^3)$. The Pauli matrices are sparse, which reduces the complexity of the brute force method to $O(N^2)$ (e.g., Ref. [26]). Still, the presented decomposition procedure has exponentially fewer multiplications.

An additional advantage of the presented decomposition is the automatic availability of the commuting subsets. Each of these subsets can be simultaneously diagonalized. This presents an opportunity for complexity reduction when evaluating the Hamiltonian.

For a practical demonstration of the proposed decomposition procedure we constructed a circuit for Hamiltonian simulation using an example of the one-dimensional wave equation (Sec. V). It is a case where a tridiagonal matrix naturally arises in the discretization of the differential equation, while the Hamiltonian is of the type considered above. There have been numerous studies of this example in the literature; in particular, there is an implementation using an oracle for the Hamiltonian evolution [9].

Our main result is captured in Fig. 3, where we show that the computational complexity, specifically the number of gates needed to reach the accuracy of 10^{-5}, scales better than a theoretical estimate. It is also better than the oracle implementation [9] for small circuits $n < 15$, with an additional advantage that the number of qubits needed to implement the Hamiltonian evolution with the presented method is by a factor of two smaller than the oracle approach for any size matrix.

We believe that our method can be applied to the five-, seven-, and more-diagonal matrices which arise in the discretization of differential in two and three dimensions. Therefore, this will be a natural extension of our study.

We have formulated our results using Walsh operators which lift to a map on boolean strings. Similar results can be obtained by using Pauli strings directly. However, we believe, using this approach, the propositions on commuting sets and the other relevant results can be expressed algebraically in a concise and simple form.

Finally, we believe that the found commuting subsets are minimal, however a rigorous prove is deferred to future work.

The PYTHON code for the numerical experiment presented in Fig. 3 is available on GitHub [27].

ACKNOWLEDGMENT

The authors acknowledge the use of Skoltech's Zhores supercomputer [28] for obtaining the numerical results presented in this paper.

All authors conceived and developed the theory and design of this study and verified the method.

The authors declare no competing interests.

APPENDIX: PROOFS

1. Formulae for decomposition coefficients

We call matrix B an upper l-diagonal matrix if it has the following form:

$$B = \begin{pmatrix} 0 & \dots & a_0 & 0 & \dots & 0 \\ \vdots & 0 & \dots & a_1 & \vdots & \vdots \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & a_{2^n-1-l} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & \dots & 0 \end{pmatrix} \tag{A1}$$

and can be written as

$$\begin{aligned} B &= \sum_{k=0}^{2^n-1-l} a_k |\mathrm{BIN}(k)\rangle\langle \mathrm{BIN}(k+l)| \\ &= \sum_{p,q=0}^{2^n-1-l} a_p \delta_{\mathrm{BIN}(p+l),\mathrm{BIN}(q)} |\mathrm{BIN}(p)\rangle\langle \mathrm{BIN}(q)|. \end{aligned} \tag{A2}$$

Similarly, the lower l-diagonal matrix is introduced as the transposed upper l-diagonal matrix, thus we will not consider this case separately and limit ourselves to the upper l-diagonal matrix. We also note that Pauli strings present in the decomposition of some matrix B will be the same for $B^\top$ because all Pauli matrices except Y are symmetric and $Y^\top = -Y$, so after transposition Pauli strings in decomposition will be the same, but coefficients may change a bit.

Proposition 4. Let B be an upper l-diagonal matrix. If a Pauli string P enters the Pauli string decomposition of matrix $B \in \mathbb{C}^{2^n \times 2^n}$ nontrivially, then $\exists p \in \{0, \dots, 2^n - 1 - l\}$:

$$\mathrm{BIN}(p + l) = \mathrm{BIN}(p) \oplus \mathbf{x}, \tag{A3}$$

where $\mathbf{x}$ and $\mathbf{z}$ is such that $P = \hat{W}(\mathbf{x}, \mathbf{z})$.

Proof. Let $B \in \mathbb{C}^{2^n \times 2^n}$, then decomposition into standard basis may be written as

$$B = \sum_{\mathbf{p},\mathbf{q} \in \mathbb{B}^n} b_{\mathbf{p},\mathbf{q}} |\mathbf{p}\rangle\langle \mathbf{q}|. \tag{A4}$$

On the other hand, decomposition into Pauli basis in $\mathbb{C}^{2^n \times 2^n}$ takes the form

$$B = \frac{1}{2^n} \sum_{\mathbf{x},\mathbf{z} \in \mathbb{B}^n} \beta_{\mathbf{x},\mathbf{z}} \hat{W}(\mathbf{x}, \mathbf{z}), \tag{A5}$$

where coefficients $\beta_{\mathbf{x},\mathbf{z}} \in \mathbb{C}$. These coefficients can be found by taking the inner product of matrix B and $\hat{W}$ and using formulas (4) and (5) from the main text.

$$\begin{aligned}\beta_{\mathbf{x},\mathbf{z}} &= \mathrm{Tr}[B\hat{W}(\mathbf{x},\mathbf{z})] = \mathrm{Tr}\left(\sum_{\mathbf{p},\mathbf{q}\in\mathbb{B}^n} b_{\mathbf{p},\mathbf{q}}|\mathbf{p}\rangle\langle\mathbf{q}|i^{\mathbf{x}\cdot\mathbf{z}}X^{\mathbf{x}}Z^{\mathbf{z}}\right) = \sum_{\mathbf{s}\in\mathbb{B}^n}\langle\mathbf{s}|\left(\sum_{\mathbf{p},\mathbf{q}\in\mathbb{B}^n} b_{\mathbf{p},\mathbf{q}}|\mathbf{p}\rangle\langle\mathbf{q}|i^{\mathbf{x}\cdot\mathbf{z}}X^{\mathbf{x}}Z^{\mathbf{z}}\right)|\mathbf{s}\rangle\\
&= \sum_{\mathbf{s}\in\mathbb{B}^n}\sum_{\mathbf{p},\mathbf{q}\in\mathbb{B}^n}\langle\mathbf{s}|b_{\mathbf{p},\mathbf{q}}|\mathbf{p}\rangle\langle\mathbf{q}|i^{\mathbf{x}\cdot\mathbf{z}}(-1)^{\mathbf{z}\cdot\mathbf{s}}|\overline{\mathbf{s}}^{\mathbf{x}}\rangle = \sum_{\mathbf{s}\in\mathbb{B}^n}\sum_{\mathbf{p},\mathbf{q}\in\mathbb{B}^n} b_{\mathbf{p},\mathbf{q}}\langle\mathbf{s}\rangle\mathbf{p}\langle\mathbf{q}\rangle\overline{\mathbf{s}}^{\mathbf{x}}i^{\mathbf{x}\cdot\mathbf{z}}(-1)^{\mathbf{z}\cdot\mathbf{s}}\\
&= \sum_{\mathbf{s}\in\mathbb{B}^n}\sum_{\mathbf{p},\mathbf{q}\in\mathbb{B}^n} b_{\mathbf{p},\mathbf{q}}\delta_{\mathbf{s},\mathbf{p}}\delta_{\mathbf{q},\overline{\mathbf{s}}^{\mathbf{x}}}i^{\mathbf{x}\cdot\mathbf{z}}(-1)^{\mathbf{z}\cdot\mathbf{s}} = \sum_{\mathbf{p},\mathbf{q}\in\mathbb{B}^n} b_{\mathbf{p},\mathbf{q}}\delta_{\mathbf{q},\overline{\mathbf{p}}^{\mathbf{x}}}i^{\mathbf{x}\cdot\mathbf{z}}(-1)^{\mathbf{z}\cdot\mathbf{p}} = \sum_{\mathbf{p}\in\mathbb{B}} i^{\mathbf{x}\cdot\mathbf{z}}(-1)^{\mathbf{z}\cdot\mathbf{p}}b_{\mathbf{p},\overline{\mathbf{p}}^{\mathbf{x}}}.\end{aligned} \tag{A6}$$

Since from (A2) we have

$$B = \sum_{p,q=0}^{2^n-1-l} a_p\delta_{\mathrm{BIN}(p+l),\mathrm{BIN}(q)}|\mathrm{BIN}(p)\rangle\langle\mathrm{BIN}(q)|, \tag{A7}$$

we can obtain $b_{\mathbf{p},\mathbf{q}}$ by equating the coefficients for $|\mathbf{p}\rangle\langle\mathbf{q}|$ and $|\mathrm{BIN}(p)\rangle\langle\mathrm{BIN}(q)|$:

$$b_{\mathbf{p},\mathbf{q}} = b_{\mathrm{BIN}(p),\mathrm{BIN}(q)} = a_p\delta_{\mathrm{BIN}(p+l),\mathrm{BIN}(q)}. \tag{A8}$$

Substituting this expression into (A6), we obtain final formula for the coefficients

$$\beta_{\mathbf{x},\mathbf{z}} = \sum_{p=0}^{2^n-1-l} i^{\mathbf{x}\cdot\mathbf{z}}(-1)^{\mathbf{z}\cdot\mathrm{BIN}(p)}\delta_{\mathrm{BIN}(p+l),\overline{\mathrm{BIN}(p)}^{\mathbf{x}}}a_p. \tag{A9}$$

Hence, for $\beta_{\mathbf{x},\mathbf{z}} \neq 0$ there should exist a solution $p \in \{0,\dots,2^n-1-l\}$ to the following equation:

$$\mathrm{BIN}(p+l) = \overline{\mathrm{BIN}(p)}^{\mathbf{x}} = \mathrm{BIN}(p)\oplus\mathrm{BIN}(x). \tag{A10}$$

■

2. Proof of proposition 1 for decomposition of general tridiagonal matrix

Proof of Proposition 2.

$$\begin{aligned}PQ &= \hat{W}(\mathbf{x},\mathbf{z})\hat{W}(\mathbf{a},\mathbf{b})\\
&= \bigotimes_{l=1}^{n} i^{x_l\cdot z_l}X^{x_l}Z^{z_l}i^{a_l\cdot b_l}X^{a_l}Z^{b_l}\\
&= \bigotimes_{l=1}^{n} i^{x_l\cdot z_l}i^{a_l\cdot b_l}(-1)^{z_l\cdot a_l}X^{x_l}X^{a_l}Z^{z_l}Z^{b_l}\\
&= \bigotimes_{l=1}^{n} i^{x_l\cdot z_l}i^{a_l\cdot b_l}(-1)^{z_l\cdot a_l}X^{a_l}X^{x_l}Z^{b_l}Z^{z_l}\\
&= \bigotimes_{l=1}^{n} i^{x_l\cdot z_l}i^{a_l\cdot b_l}(-1)^{z_l\cdot a_l}(-1)^{x_l\cdot b_l}X^{a_l}Z^{b_l}X^{x_l}Z^{z_l}\\
&= (-1)^{\mathbf{z}\cdot\mathbf{a}}(-1)^{\mathbf{x}\cdot\mathbf{b}}\bigotimes_{l=1}^{n} i^{x_l\cdot z_l}i^{a_l\cdot b_l}X^{a_l}Z^{b_l}X^{x_l}Z^{z_l}\\
&= (-1)^{\mathbf{z}\cdot\mathbf{a}}(-1)^{\mathbf{x}\cdot\mathbf{b}}\hat{W}(\mathbf{a},\mathbf{b})\hat{W}(\mathbf{x},\mathbf{z})\\
&= (-1)^{\mathbf{z}\cdot\mathbf{a}}(-1)^{\mathbf{x}\cdot\mathbf{b}}QP.\end{aligned} \tag{A11}$$

Thus, we see that two Pauli strings P and Q commute if $\mathbf{z}\cdot\mathbf{a}+\mathbf{x}\cdot\mathbf{b} = 0 \pmod 2$. ■

Proof of Corollary 1. We denote the number of Y operators in Pauli string P as $N_Y(P)$. Let P correspond to $\hat{W}(\mathbf{x},\mathbf{z})$,

$$\hat{W}(\mathbf{x},\mathbf{z}) = \bigotimes_{l} i^{x_lz_l}X^{x_l}Z^{z_l}. \tag{A12}$$

Note that if and only if for some l, $x_l = z_l = 1$, then

$$i^{x_lz_l}X^{x_l}Z^{z_l} = iXZ = Y. \tag{A13}$$

Therefore, the number of all bit positions where $x_l = z_l = 1$ corresponds to the number of Y in the Pauli string, i.e.,

$$N_Y(P) = \mathbf{x}\cdot\mathbf{z}. \tag{A14}$$

Since we have $\mathbf{x}\cdot\mathbf{z} = \mathbf{x}\cdot\mathbf{b} \pmod 2$ from our assumption, then from proposition 2 (here $\mathbf{a} = \mathbf{x}$) a commutation of P and Q follows. Which means (12) holds, and therefore P and Q commute. ■

Proof of Proposition 1. A tridiagonal matrix decomposition comprises terms for the diagonal ($l = 0$), upper one-diagonal and lower one-diagonal (that is transposed upper one-diagonal). To obtain upper bounds on the number of Pauli strings in the decomposition of B it suffices to check the number of pairs $(\mathbf{x},\mathbf{z})$ that satisfy the necessary conditions from proposition 4.

(1) For $l = 0$ in (A3) we get

$$\text{Left-hand side} = \mathrm{BIN}(p+0) = (p_1,\dots,p_n), \tag{A15}$$

$$\text{Right-hand side} = \overline{\mathrm{BIN}(p)}^{\mathbf{x}} = (\overline{p}_1^{x_1},\dots,\overline{p}_n^{x_n}). \tag{A16}$$

Thus, for a Pauli string to satisfy the necessary conditions, $\mathbf{x} = (0,\dots,0)$ and $\mathbf{z}$ can be arbitrary. Therefore, the maximum number of Pauli strings in the decomposition of diagonal matrix is bounded by 2^n.

(2) For $l = 1$ in (A3), using binary summation rules (recall that leftmost bit encodes lowest register), we get

$$\mathrm{BIN}(p+1) = \left(\overline{p_1}, p_2^{\overline{p_1}}, p_3^{\overline{p_1p_2}},\dots\right). \tag{A17}$$

Since $x^p = x\oplus\overline{p}$, we have

$$[\mathrm{BIN}(p+1)]_1 = \overline{p}_1,$$

$$[\mathrm{BIN}(p+1)]_j = p_j^{\overline{\prod_{k=1}^{j-1}p_k}} = p_j\oplus\prod_{k=1}^{j-1}p_k,\quad j>1.$$

Now we can write (A3) in the following form:

$$p_j\oplus x_j = p_j\oplus\prod_{k=1}^{j-1}p_k,\quad j>1, \tag{A18}$$

which means

$$x_j = \prod_{k=1}^{j-1} p_k \tag{A19}$$

for $j > 1$ and $x_1 = 1$. It follows that for any $p \in \mathbb{B}^n$, if $p_j = 0$, then $x_{j+1}, \ldots, x_n$ are equal to 0. Therefore, x can only be one of the following strings:

$$\begin{gathered}(1, 0, \ldots, 0),\\ (1, 1, \ldots, 0),\\ \vdots\\ (1, 1, \ldots, 1).\end{gathered} \tag{A20}$$

Therefore, the maximum number of Pauli strings in the decomposition of upper one-diagonal matrix is bounded by $n2^n$.

In conclusion, adding the upper bounds for the diagonal, upper, and lower one-diagonal cases, we obtain that the number of terms in the decomposition of B is upper bounded by $(n+1)2^n$ because Pauli strings for upper and lower one-diagonal matrices will be the same as was mentioned before. Each of the above strings corresponds to a particular set in the decomposition of B provided in the proposition. ■

3. Proof of proposition 3 for decomposition of real tridiagonal matrix

Proof of Proposition 3. The condition that the matrix B is real can be expressed as

$$B = B^*, \tag{A21}$$

where $*$ means complex conjugation. From definitions (A5) and (A21) it follows that

$$\begin{aligned}\text{Left-hand side} &= \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n} \beta_{\mathbf{x},\mathbf{z}} \hat{W}(\mathbf{x},\mathbf{z})\\ &= \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n} \beta_{\mathbf{x},\mathbf{z}} i^{\mathbf{x}\cdot\mathbf{z}} X^{\mathbf{x}} Z^{\mathbf{z}},\\ \text{Right-hand side} &= \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n} \beta^*_{\mathbf{x},\mathbf{z}} \hat{W}^*(\mathbf{x},\mathbf{z})\\ &= \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n} \beta^*_{\mathbf{x},\mathbf{z}} (-i)^{\mathbf{x}\cdot\mathbf{z}} X^{\mathbf{x}} Z^{\mathbf{z}}\\ &= \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n} \beta^*_{\mathbf{x},\mathbf{z}} (-1)^{\mathbf{x}\cdot\mathbf{z}} i^{\mathbf{x}\cdot\mathbf{z}} X^{\mathbf{x}} Z^{\mathbf{z}}\\ &= \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n} \beta^*_{\mathbf{x},\mathbf{z}} (-1)^{\mathbf{x}\cdot\mathbf{z}} \hat{W}(\mathbf{x},\mathbf{z}).\end{aligned} \tag{A22}$$

By comparing the left- and right-hand side, one can obtain conditions on coefficients $\beta_{\mathbf{x},\mathbf{z}}$:

$$\beta_{\mathbf{x},\mathbf{z}} = \beta^*_{\mathbf{x},\mathbf{z}} (-1)^{\mathbf{x}\cdot\mathbf{z}}. \tag{A23}$$

Thus, $\beta_{\mathbf{x},\mathbf{z}}$ has only the real part if $\mathbf{x}\cdot\mathbf{z} = 0 \pmod 2$ or has only the imaginary part if $\mathbf{x}\cdot\mathbf{z} = 1 \pmod 2$.

From corollary 1, Pauli strings commute if the parity of the number of Y operators $N_Y = \mathbf{x}\cdot\mathbf{z}$ in each string is the same. Hence, we can partition the set of Pauli strings in terms of $\mathbf{x}$ in proposition 1 into commuting subsets due to the fact that for every string $\mathbf{x}$, there are two sets of $\mathbf{z}$ with equal cardinality 2^{n-1} that result in N_Y being zero or one modulo 2. However, for $\mathbf{x} = (0, \ldots, 0)$, any $\mathbf{z}$ will not change N_Y. Consequently, we will end up with $2n$ subsets, each with cardinality 2^{n-1}, and one subset with 2^n elements. ■

Proof of Corollary 2. Let us recall the decomposition of B:

$$B = \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n} \beta_{\mathbf{x},\mathbf{z}} \hat{W}(\mathbf{x},\mathbf{z}). \tag{A24}$$

Since $X^\top = X$, $Z^\top = Z$, and $XZ = -ZX$ for an arbitrary Pauli operator $\hat{W}(\mathbf{x},\mathbf{z})$, we can write

$$\hat{W}^\top(\mathbf{x},\mathbf{z}) = i^{\mathbf{x}\cdot\mathbf{z}} Z^{\mathbf{z}} X^{\mathbf{x}} = i^{\mathbf{x}\cdot\mathbf{z}} (-1)^{\mathbf{x}\cdot\mathbf{z}} X^{\mathbf{x}} Z^{\mathbf{z}} = (-1)^{\mathbf{x}\cdot\mathbf{z}} \hat{W}(\mathbf{x},\mathbf{z}). \tag{A25}$$

Since matrix B is symmetric,

$$\begin{aligned}B = B^\top &= \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n} \beta_{\mathbf{x},\mathbf{z}} \hat{W}^\top(\mathbf{x},\mathbf{z})\\ &= \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n} \beta_{\mathbf{x},\mathbf{z}} (-1)^{\mathbf{x}\cdot\mathbf{z}} \hat{W}(\mathbf{x},\mathbf{z}).\end{aligned} \tag{A26}$$

Comparing the right- and left-hand side, we obtain a condition for coefficients $\beta_{\mathbf{x},\mathbf{z}}$:

$$\beta_{\mathbf{x},\mathbf{z}} = \beta_{\mathbf{x},\mathbf{z}} (-1)^{\mathbf{x}\cdot\mathbf{z}}, \tag{A27}$$

which can be satisfied for nontrivial $\beta_{\mathbf{x},\mathbf{z}}$ only when

$$\mathbf{x}\cdot\mathbf{z} = 0 \pmod 2. \tag{A28}$$

If $\mathbf{x} = 0$, condition (A28) is automatically satisfied; for each of the n remaining possible strings $\mathbf{x}$ only half of the possible $\mathbf{z}$ strings satisfy (A28) and thus form subsets of size 2^{n-1}.

Also, it follows from (A28) that the number of Y operators N_Y must be even. Therefore, the decomposition of the real symmetric tridiagonal matrix consists only of subsets $S_{m,+}$. Thus, we have n subsets $S_{m,+}$ of size 2^{n-1} and one subset S_0 of size 2^n. ■

Proof of Corollary 3. The Hermitian matrix H is expressed in terms of matrix B:

$$H = \begin{pmatrix} 0 & B \\ B^\dagger & 0 \end{pmatrix}. \tag{A29}$$

Let us introduce the following matrices:

$$\begin{aligned}E_{12} &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \frac{1}{2}(X - XZ),\\ E_{21} &= \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} = \frac{1}{2}(X + XZ).\end{aligned} \tag{A30}$$

Then, H can be constructed as

$$\begin{aligned}H &= E_{12}\otimes B + E_{21}\otimes B^{\dagger}\\ &= \frac{1}{2}(X - XZ)\otimes B + \frac{1}{2}(X + XZ)\otimes B^{\dagger}\\ &= \frac{1}{2}[X\otimes(B+B^{\dagger}) - XZ\otimes(B-B^{\dagger})]\\ &= \frac{1}{2^n}X\otimes\left[\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n}\frac{\beta_{\mathbf{x},\mathbf{z}}+\beta^*_{\mathbf{x},\mathbf{z}}}{2}\hat{W}(\mathbf{x},\mathbf{z})\right]\\ &\quad - \frac{1}{2^n}XZ\otimes\left[\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n}\frac{\beta_{\mathbf{x},\mathbf{z}}-\beta^*_{\mathbf{x},\mathbf{z}}}{2}\hat{W}(\mathbf{x},\mathbf{z})\right]\\ &= \frac{1}{2^n}X\otimes\left[\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n}\mathrm{Re}\beta_{\mathbf{x},\mathbf{z}}\hat{W}(\mathbf{x},\mathbf{z})\right]\\ &\quad - \frac{1}{2^n}iXZ\otimes\left[\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n}\mathrm{Im}\beta_{\mathbf{x},\mathbf{z}}\hat{W}(\mathbf{x},\mathbf{z})\right]\\ &= \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n}(\mathrm{Re}\beta_{\mathbf{x},\mathbf{z}}X - i\mathrm{Im}\beta_{\mathbf{x},\mathbf{z}}XZ)\otimes\hat{W}(\mathbf{x},\mathbf{z})\\ &= \frac{1}{2^n}\sum_{\mathbf{x},\mathbf{z}\in\mathbb{B}^n}(\mathrm{Re}\beta_{\mathbf{x},\mathbf{z}}X - \mathrm{Im}\beta_{\mathbf{x},\mathbf{z}}Y)\otimes\hat{W}(\mathbf{x},\mathbf{z}). \qquad (A31)\end{aligned}$$

Since matrix B is real valued, then according to (A23),

$$\begin{aligned}&\mathrm{Im}\beta_{\mathbf{x},\mathbf{z}} = 0, \quad \text{if} \quad \mathbf{x}\cdot\mathbf{z} = 0\ (\mathrm{mod}\ 2),\\ &\mathrm{Re}\beta_{\mathbf{x},\mathbf{z}} = 0, \quad \text{if} \quad \mathbf{x}\cdot\mathbf{z} = 1\ (\mathrm{mod}\ 2). \qquad (A32)\end{aligned}$$

Now, in Eq. (A31), since either $\mathrm{Re}\beta_{\mathbf{x},\mathbf{z}} = 0$ or $\mathrm{Im}\beta_{\mathbf{x},\mathbf{z}} = 0$, for $\mathbf{x},\mathbf{z}\in\mathbb{B}^n$ the number of terms in the decomposition of H is the same as that for B. From (A31) and the decomposition in proposition 1 it follows that

$$\begin{aligned}&S_0 = \{X\}\bigotimes_1^n\{I,Z\} && (m=0)\\ &S_m = \{\widehat{X,Y}\}\bigotimes_1^{n-m}\{I,Z\}\bigotimes_1^m\{X,Y\}, && m = 1,\dots,n-1\\ &S_n = \{\widehat{X,Y}\}\bigotimes_1^m\{X,Y\}, && (m=n)\end{aligned}$$

i.e., the commuting subsets are constructed by appending X or Y at the beginning of the Pauli strings in the decomposition to ensure the required parity.

[1] R. P. Feynman *et al.*, Int. J. Theor. Phys. **21**, 467 (1982).
[2] S. Lloyd, Science **273**, 1073 (1996).
[3] D. W. Berry, A. M. Childs, and R. Kothari, in *Proceedings of the 2015 IEEE 56th Annual Symposium on Foundations of Computer Science, Berkeley, CA, USA* (IEEE, 2015).
[4] G. H. Low and I. L. Chuang, Quantum **3**, 163 (2019).
[5] A. M. Childs, R. Cleve, E. Deotto, E. Farhi, S. Gutmann, and D. A. Spielman, in *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing* (ACM, New York, NY, 2003).
[6] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Phys. Rev. Lett. **114**, 090502 (2015).
[7] D. W. Berry and L. Novo, Quantum Inf. Comput. **16**, 1295 (2016).
[8] G. H. Low and I. L. Chuang, Phys. Rev. Lett. **118**, 010501 (2017).
[9] A. Suau, G. Staffelbach, and H. Calandra, ACM Trans. Quantum Comput. **2**, 1 (2021).
[10] H. F. Trotter, Proc. Am. Math. Soc. **10**, 545 (1959).
[11] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe, and S. Zhu, Phys. Rev. X **11**, 011020 (2021).
[12] Y. Kawase and K. Fujii, Comput. Phys. Commun. **288**, 108720 (2023).
[13] E. van den Berg and K. Temme, Quantum **4**, 322 (2020).
[14] P. Mukhopadhyay, N. Wiebe, and H. T. Zhang, npj Quantum Inf. **9**, 31 (2023).
[15] J. Lawrence, Č. Brukner, and A. Zeilinger, Phys. Rev. A **65**, 032320 (2002).
[16] T.-C. Yen, A. Ganeshram, and A. F. Izmaylov, npj Quantum Inf. **9**, 14 (2023).
[17] O. Crawford, B. van Straaten, D. Wang, T. Parks, E. Campbell, and S. Brierley, Quantum **5**, 385 (2021).
[18] C. Pozrikidis, *Try the new Google Books* (Oxford University Press, USA, 2014).
[19] R. J. LeVeque, Draft version for use in AMath 585, 112 (1998).
[20] M. Cerezo *et al.*, Nat. Rev. Phys. **3**, 625 (2021).
[21] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, Nat. Commun. **5**, 4213 (2014).
[22] J. Welch, D. Greenbaum, S. Mostame, and A. Aspuru-Guzik, New J. Phys. **16**, 033040 (2014).
[23] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Commun. Math. Phys. **270**, 359 (2007).
[24] A. Tranter, P. J. Love, F. Mintert, N. Wiebe, and P. V. Coveney, Entropy **21**, 1218 (2019).
[25] P. C. S. Costa, S. Jordan, and A. Ostrander, Phys. Rev. A **99**, 012323 (2019).
[26] R. Yuster and U. Zwick, ACM Trans. Algorithms (TALG) **1**, 2 (2005).
[27] GitHub page with numerical experiments, https://github.com/barseniev/tridiagonal-matrix-decomposition-quantum-simulation
[28] I. Zacharov, R. Arslanov, M. Gunin, D. Stefonishin, A. Bykov, S. Pavlov, O. Panarin, A. Maliutin, S. Rykovanov, and M. Fedorov, Open Eng. **9**, 512 (2019).