

## ARTICLE OPEN



## Synthesizing efficient circuits for Hamiltonian simulation

Priyanka Mukhopadhyay<sup>1,2</sup>, Nathan Wiebe<sup>3,4</sup> and Hong Tao Zhang<sup>5</sup>

We provide an approach for compiling quantum simulation circuits that appear in Trotter, qDRIFT and multi-product formulas to Clifford and non-Clifford operations that can reduce the number of non-Clifford operations. The total number of gates, especially CNOT, reduce in many cases. We show that it is possible to implement an exponentiated sum of commuting Paulis with at most  $m$  (controlled)-rotation gates, where  $m$  is the number of distinct non-zero eigenvalues (ignoring sign). Thus we can collect mutually commuting Hamiltonian terms into groups satisfying one of several symmetries identified in this work. This allows an inexpensive simulation of the entire group of terms. We further show that the cost can in some cases be reduced by partially allocating Hamiltonian terms to several groups and provide a polynomial time classical algorithm that can greedily allocate the terms to appropriate groupings.

npj Quantum Information (2023)9:31; <https://doi.org/10.1038/s41534-023-00697-6>

## INTRODUCTION

One main reason that led Feynman<sup>1</sup> and others to propose the idea of quantum computers was the fact that problems like simulating the dynamics of quantum systems are intractable on a classical computer. Starting from the seminal work of Lloyd<sup>2</sup>, much research<sup>3</sup> has been done to develop algorithms for simulating Hamiltonians, culminating in various techniques like product formulas<sup>4,5</sup>, quantum walks<sup>6</sup>, linear combination of unitaries<sup>7</sup>, truncated Taylor series<sup>8</sup>, and quantum signal processing<sup>9</sup>. Special techniques have been developed for simulating particular physical systems<sup>10–17</sup>, which might find applications in developing new pharmaceuticals, catalysts and materials. Phase estimation can be combined with quantum simulation to find the ground state energy<sup>18</sup> and excited state energies<sup>19–21</sup> of the Hamiltonian. This is called the electronic structure problem<sup>14</sup>, which is important in chemistry and material science. Research in quantum simulation has also inspired the development of quantum algorithms for various other problems<sup>22–26</sup>.

One main challenge for digital quantum simulation is the implementation of efficient circuits that can produce reliable results. Without it, a theoretical exponential speedup may not lead to a useful algorithm if a typical practical application requires an amount of time and memory that is beyond the reach of even a quantum computer. There are a number of factors that can affect the efficiency of a quantum circuit i.e. its running time and error, for example, the number of qubits, depth, gate count, etc. So depending upon the applications or other hardware constraints, one can design algorithms that optimize or reduce the count/depth of one particular type of quantum gate or other resources. For example, there are algorithms that do T-count and T-depth-optimal synthesis<sup>27–29</sup> given a unitary or does re-synthesis of a given circuit with reduced T-count, T-depth<sup>30–32</sup> or CNOT-count<sup>33–35</sup>. The non-Clifford T gate has known constructions in most of the error correction schemes and the cost of fault-tolerantly implementing it exceeds the cost of the Clifford group gates by as much as a factor of hundred or more<sup>36–38</sup>. Quantum error correction and fault tolerance are especially significant for large quantum circuits, else the accumulation of errors will make

any output highly unreliable and hence useless. The minimum number of T-gates required to implement certain unitaries is a quantifier of difficulty in many algorithms<sup>39,40</sup> that try to classically simulate quantum computation. So, even though alternative fault-tolerance methods such as completely transversal Clifford+T scheme<sup>41</sup> and anyonic quantum computing<sup>42</sup> are also being explored, minimization of the number of T gates in quantum circuits remain an important and widely studied goal. Multi-qubit gates like CNOT introduce more error than single qubit gates, so reducing CNOT gate is important and especially relevant for the noisy intermediate scale quantum (NISQ) computers.

## Our contributions

(I) One main result in this paper is Lemma 2.4, which shows that it is possible to implement an exponentiated sum of commuting Paulis with at most  $m$  (controlled)-rotation gates, where  $m$  is the number of distinct non-zero eigenvalues (ignoring sign). For illustration, we consider the Hamiltonian for the Heisenberg model and we show that it is possible to achieve about 50% reduction in the rotation gate cost and for certain underlying graphs this reduction can be about 75%. However, the cost of Toffolis may increase. We have given explicit circuits for 4-qubit and 6-qubit chain (or cycle), where we attempt to reduce both the rotation and Toffoli gate cost.

(II) In most previous works, circuits for individual exponentiated Paulis are synthesized and combined. We show that it is possible to reduce the gate count (not only non-Clifford gates) if we instead consider groups of commuting Paulis. To give some practical demonstration, we consider the qDRIFT Hamiltonian simulation algorithm<sup>43</sup>. We call the error introduced due to the algorithm as ‘simulation error’. We take the 1-D 4 qubit and 6 qubit Heisenberg Hamiltonians (Fig. 7) and also 4-qubit Hamiltonians for  $H_2$  and  $LiH$  (with freezing in the STO-3G basis) (Fig. 8), and compare the case where a single Pauli term is selected with the case where a set of commuting Pauli terms is selected for implementation at each iteration of qDRIFT. We observe that the error accumulation is less for multiple terms and also the rotation gate cost is less in this error regime. The number of Toffoli pairs is

<sup>1</sup>Institute for Quantum Computing, University of Waterloo, Waterloo, ON, Canada. <sup>2</sup>Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ON, Canada.

<sup>3</sup>Department of Computer Science, University of Toronto, Toronto, ON, Canada. <sup>4</sup>Pacific Northwest National Laboratory, Richland, WA, USA. <sup>5</sup>Department of Mathematics, University of Toronto, Toronto, ON, Canada. ✉email: mukhopadhyay.priyanka@gmail.com

**Table 1.** Comparison of gate counts required to simulate  $e^{-iHt}$  (Eqs. (19), (20)) using the circuit synthesized by us with the circuit in ref. <sup>50</sup>.

	Case I	Case II	Case III	[50]
CNOT	6	10	8	38
H	2	2	2	2
X	2	6	6	0
$(c)R_z$	1	1	3	8
#Toff. pair	2	2	4	0
Total	15	23	27	42

The gate counts have been given for the three different cases considered in this work.

roughly equal to the number of  $R_z/cR_z$  used, in case of multiple terms. So overall, we have less T-count when implementing multiple commuting Paulis per iteration of qDRIFT. This adds to the motivation of building efficient circuits for such Hamiltonians.

(III) In subsection ‘Optimized Circuits for Quantum Chemistry’ we derive explicit quantum circuits for the two-body excitation terms appearing in the Coulomb Hamiltonian in quantum chemistry. We mainly use the Clifford+T universal fault-tolerant gate set to implement unitaries. We design efficient circuits for a different grouping of commuting Pauli operators. It is evident (Table 2) that the rotation gate cost depends on the coefficients of the Pauli summands. For some combination of coefficients the circuits derived here are optimal, in the sense, that they have the minimum (i.e. 1) number of  $cR_z$  gates. Though our focus is on reducing the non-Clifford gate count, but most of the quantum circuits derived here have an overall reduced gate count, including a reduction in the 2-qubit gates like CNOT. In Table 1 we have compared the number of gates required to implement one of the Hamiltonians considered in this paper with a previous construction. For the remaining Hamiltonians we did not find any compact previous construction to compare with. In short, our approach can be useful not only in the fault-tolerant regime, but also in the NISQ era.

(IV) In Algorithm 1, we describe a greedy method of grouping into commuting Paulis, but the objective is to optimize the number of non-Clifford gates. There have been a host of work that tackles the question of how to group the commuting Paulis and to the best of our knowledge most (if not all) of them has the objective to reduce the number of measurements required to make an estimation<sup>44</sup>. The latter problem is especially important for variational quantum eigensolvers. The grouping that optimizes the non-Clifford gates may not optimize the number of measurements. In most cases, finding the optimal grouping is difficult. But we can always ask the question that given a grouping (for whatever objective), is it possible to compile efficient circuits. In this case, we can use our techniques (Lemma 2.4) to reduce the gate count. Thus our methods can also be used to design circuits for the measurement problem.

In this paper, we use the Jordan–Wigner (JW) transformation<sup>45</sup> to map from the fermionic to the qubit space. And then we group into commuting Paulis. Other transformations like Bravyi–Kitaev and parity transformations<sup>46</sup> can also be used and may be beneficial in circumstances where Clifford operations are costly or inherent quantum error correction is desirable. We focus on Jordan–Wigner for two reasons. First, in this paper, we focus on the synthesis of efficient quantum circuits for exponentiated commuting Paulis and the techniques hold no matter whichever mapping is considered. Second, previous work has not shown obvious advantages for Bravyi–Kitaev transformations within the domain of fault-tolerant quantum computing.

## How we compare the cost of non-Clifford resources

In all the constructions discussed in this paper, two approximately implementable gates are used— $R_z$  and controlled  $R_z$  ( $cR_z$ ), whose T-count varies inversely with precision or synthesis error. From the results given in<sup>29</sup> and from the implementations performed here until the error  $10^{-6}$ , we believe that T-count of  $cR_z$  can be less than that of  $R_z$  for most modestly small rotation angles. However, for convenience, we assume these have equal cost and with some abuse of terms, we refer to the T-count of  $R_z/cR_z$  as the ‘(non-Clifford) rotation gate cost’. The only exactly implementable non-Clifford unitary/gate considered in the constructions is Toffoli with T-count  $7^{27}$  or  $4^{47}$ . For low error regime, the T-count of approximately implementable  $R_z/cR_z$  will dominate, while in a high error regime the T-count of Toffoli may matter, if we use a lot of them. To reduce the T-count of compute-uncompute Toffoli pairs, we can use the temporary logical AND gadget, proposed by Gidney<sup>48</sup>. In fact, in our circuits, we use  $R_z$  gates controlled on  $n$  qubits ( $n > 1$ ), each of which can be decomposed into (compute–uncompute) pairs of NOT gates controlled on  $n$  qubits and a  $cR_z$  gate. Each such multi-controlled NOT can be implemented with  $n-1$  Toffoli or  $4n-4$  T-gates<sup>48</sup>. If we combine compute-uncompute pairs then the overall T-count of the circuit can reduce further, by using logical AND gadget. We must keep in mind that the implementations in<sup>47,48</sup> use classical resources and measurements, and it is not straightforward to argue that it will give advantage, in spite of using less number of T-gates. We can also use the construction in<sup>49</sup> that implements an  $n$ -controlled NOT gate using  $4n-4$  T,  $4n-3$  CNOT and  $n-1$  ancillae qubits. In our paper, we have expressed the non-Clifford T-gate cost in terms of the rotation gate cost and the number of Toffoli pairs used.

## Related work

In ref. <sup>50</sup> the authors studied the non-Clifford resource cost required to simulate the chemical process of biological nitrogen fixation by nitrogenase. In ref. <sup>51</sup> the authors developed algorithms to synthesize circuits for the Clifford operators that diagonalize a group of commuting Paulis. The goal was to reduce the two-qubit CNOT gate count because of its low fidelity and limited qubit connectivity of near-term quantum computer architectures. A similar diagonalization algorithm has been used in<sup>52</sup> for efficient simulation of Hamiltonian dynamics. Much work has been done for the construction of quantum circuits for the evolution of molecular systems<sup>16,53–59</sup> and the Heisenberg model<sup>60</sup>.

## RESULTS

### Notation

In many places we write  $G_{(q)}$  to denote that the gate or operator  $G$  acts on qubit  $q$ . For multi-qubit gates we write  $CNOT_{(c,t)}$  to denote a CNOT with control at qubit  $c$  and target at qubit  $t$ . For convenience, we have removed the parenthesis in the subscript whenever there is less ambiguity. We write  $[K] = \{1, 2, \dots, K\}$ . We denote the  $n \times n$  identity matrix by  $\mathbb{I}_n$  or  $\mathbb{I}$  if dimension is clear from the context. We denote the set of  $n$ -qubit unitaries by  $\mathcal{U}_n$ . The size of an  $n$ -qubit unitary is  $N \times N$  where  $N = 2^n$ . We have given detailed description about the  $n$ -qubit Pauli operators ( $\mathcal{P}_n$ ), Clifford group ( $\mathcal{C}_n$ ) and the group ( $\mathcal{J}_n$ ) generated by the Clifford and T gates in Supplementary Note 1.

### Optimizing Trotter-decompositions

The time evolution of a quantum system, described by a Hamiltonian  $H$  is  $e^{-iHt}$ . Most often the Hamiltonian  $H$  can be decomposed as the sum  $H = \sum_{j=1}^m a_j H_j$ , where each  $H_j$  is Hermitian. There can be more than one decomposition of  $H$  and we select the one such that for each  $H_j$  the unitary  $e^{-i\tau H_j}$  is efficiently implementable on a quantum computer, for any  $\tau$ . The goal of

the Hamiltonian simulation problem is to find an approximation of  $e^{-itH}$  into a sequence of  $e^{-itH_j}$ , up to some desired precision. For example, using the Lie-Trotter formula<sup>5</sup> we have that

$$e^{-itH} = \lim_{k \rightarrow \infty} \left( \prod_j e^{-i(t/k)H_j} \right)^k.$$

In the non-asymptotic regime, the Trotter scheme provides a first-order approximation, with the norm of the difference between the exact and approximate time evolution scaling as  $O(t^2/k)$ . More advanced higher order schemes<sup>3,4</sup> are also available. Alternatively, a randomized approach called qDRIFT can be used in place of a Trotter formula wherein the quantum state is evolved according to the probabilistic channel

$$\rho \mapsto \sum_j \frac{\alpha_j}{|\alpha_1|} e^{-i|\alpha_1|H_j t} \rho e^{i|\alpha_1|H_j t}. \quad (1)$$

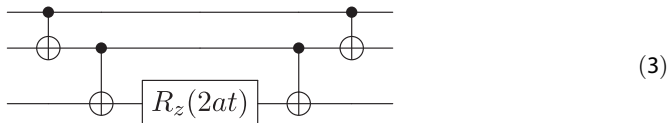
Note the error here is also  $O(t^2)$ ; however, in this case, a single exponential is performed rather than  $O(m)$  as would be needed for the comparable Trotter formula. The cost of such an approach scales as  $O(|\alpha_1|^2 t^2 / \epsilon)$  for error  $\epsilon$  and does not directly depend on  $m$ .

The approximation errors arising in the use of product formulas are caused by non-commuting terms in the Hamiltonian. For example, see ref. <sup>61</sup> for a detailed exposition on Trotter errors. Given any set of mutually commuting operators  $P_1, \dots, P_m$  we have the following:

$$e^{-it \sum_{j=1}^m P_j} = \prod_{j=1}^m e^{-itP_j} \quad (2)$$

Thus, the operators are partitioned into mutually commuting subsets. Time evolution for the sum of mutually commuting operators in each such subset is trivial, and the product formulas can be applied to the sum of Hamiltonians formed as the sum of each subset. This approach becomes especially applicable in scenarios where the Hamiltonian can be expressed as a sum of Pauli operators, for which the commutation relations can easily be evaluated.

As a specific example, consider the case where  $H = aZ \otimes Z \otimes Z$ . Since the Hamiltonian is diagonal,  $e^{-iaZ \otimes Z \otimes Z t}$  has computational basis vector  $|b_1, b_2, b_3\rangle$  and eigenvalues  $e^{-i(-1)^{b_1 \oplus b_2 \oplus b_3} at}$ . Thus the eigenvalues are determined by the parity of the bit strings, which can be computed using CNOT gates. From this reasoning the following quantum circuit will perform the simulation of this Pauli operator exactly.



As every Pauli operator of weight 3 can be diagonalized by Clifford conjugation, this circuit up to an elementary basis transformation, will simulate any weight 3 Pauli Hamiltonian. The exact same strategy of diagonalizing and simulating the Pauli operator in the eigenbasis shows that each exponential of a weight  $v$  Pauli operator Hamiltonian requires  $2(v-1)$  CNOT operators and one rotation gate. This strategy is at the heart of most elementary networks for simulating chemistry and spin models<sup>53,62</sup>.

The work of<sup>50</sup> provided another way of thinking about these decompositions by showing an explicit method that can diagonalize sums of commuting operators that appear in chemistry simulations by transforming into a simultaneous eigenbasis of such terms. In full generality, such transformations reduce the circuit depth but need not reduce the circuit size. However, we will see here that for some Hamiltonians these transformations can reduce the circuit size as well.

As a motivating example, consider the Hamiltonian  $H = XX + YY + ZZ$ . This Hamiltonian can be simulated, up to a global phase, by



This can be implemented using two Toffoli gates and a single qubit rotation. In contrast, the standard approach from<sup>53,62</sup> would use three single qubit rotations and no Toffoli gates. As rotation synthesis often is 10 times more expensive than Toffoli gates<sup>27,29,50</sup>, this will almost always be a favorable way of performing the simulation. In contrast, if this symmetry is broken then the Hamiltonian term will be more expensive to simulate. Thus it can be favorable to introduce such symmetries as needed artificially. For example, consider

$$H = XX + YY + (3/2)ZZ = (XX + YY + ZZ) + ZZ/2. \quad (5)$$

Such a simulation can be performed using two rotation gates rather than the 3 naïvely needed and so it makes sense to compile the Hamiltonian terms this way to reduce the overall complexity.

As another example, not all rotations are equally expensive and so we should also combine terms in such a way as to minimize the cost. For example consider the time-evolution operator

$$U(t) = e^{-i(\pi/4\sqrt{2}-\epsilon)Z-i(\pi/4\sqrt{2}+\epsilon)X} \approx e^{-i\pi/4(X+Z)/\sqrt{2}} e^{-i(X-Z)\epsilon}. \quad (6)$$

While the first operation in this Trotterization is not a Clifford operation, it is a simulation of a Hadamard gate for time  $\pi/4$ . As this corresponds to a special angle and since the Hadamard gate can be diagonalized using a constant size  $H$  and  $T$  circuit, the cost of implementing this first term is  $O(1)$  and thus the dominant cost is the remaining rotation. In contrast, if this property were not used then we would have two arbitrary rotations in the Trotterization which would be nearly twice the cost of this simplified approach. These ideas can further be used in concert: remainder terms that arise from inexact rounding a Hamiltonian evolution to a known cheap simulation can be absorbed into other terms or even other Trotter steps.

### Algorithm 1. Hamiltonian compilation using Greedy 1-norm minimization

**Data:**  $H = \sum_{j=1}^m \alpha_j P_j$  for distinct Pauli  $P_j$  and  $\alpha_j \geq 0$ , cost function  $\text{COST} : \mathbb{R}^m \mapsto \mathbb{Z}^+$  such that  $\text{COST}(\vec{x})$  is the cost of simulating  $H = \sum_j x_j P_j$ , where  $\vec{x} = (x_1, \dots, x_m)$  and  $\text{COST}(\vec{x}) = 1$  if  $\vec{x}$  contains at most 1 non-zero entry.  
**Output:**  $\{H_j = h_j \sum_i P_i : j = 1, \dots, m'\}$  such that  $H = \sum_{j=1}^{m'} H_j$ .

- 1  $\vec{\alpha}' \leftarrow \vec{\alpha}$ ;
- 2 **while**  $\sum_j |\alpha'_j| > 0$  **do**
- 3      $S \leftarrow \{P_j : \alpha'_j \neq 0\}$ ;
- 4      $i \leftarrow 0, \Gamma_{\max} \leftarrow 0$ ;
- 5     **while**  $S \neq \emptyset$  **do**
- 6          $i \leftarrow i + 1$ ;
- 7          $k_i \leftarrow \text{argmax}(\{|\alpha'_i| | P_i \in S\})$ ;
- 8          $\Gamma_{\max} \leftarrow \max(\Gamma_{\max}, \max_{\vec{\beta}}(\sum_j |\alpha'_j| - \sum_j |\alpha'_j - \beta_j|) / \text{COST}(\vec{\beta}))$  such that  $\beta_j \neq 0 \Rightarrow \exists p \leq i$  with  $k_p = j$ ;
- 9          $S \leftarrow \{P_i \in S | [P_i, P_{k_i}] = 0 \text{ and } \alpha'_i \neq 0\}$ ;
- 10     **end**
- 11     Store  $\sum_{j=k_i}^{k_i} \Gamma_{\max} P_j$  as a Hamiltonian term;
- 12      $\vec{\alpha}' \leftarrow \vec{\alpha}' - \Gamma_{\max} \vec{1}$ ;
- 13 **end**
- 14 **return** Set of Hamiltonian terms computed.

We propose an algorithm in Algorithm 1 that exploits this intuition through a greedy decomposition of the Hamiltonian into sums of commuting terms. These mutually commuting terms, or fragments, are chosen such that the ratio of the fraction of the Hamiltonian that is simulated by the term to the cost of the term is maximized. This choice is motivated in part by the fact that the query complexity of a quantum simulation is lower bounded by  $\Omega(|\alpha_1|t)$ <sup>63</sup> and thus designing circuits that simulate as large a fraction of this one-norm as possible per quantum gate operation

is a sensible optimization heuristic for our greedy algorithm. Unlike traditional approaches to partitioning the Hamiltonian, our approach allows partial allocation of Hamiltonian terms to multiple commuting sets. Further, the allocation can be negative in our approach. This negative allocation is important because we will see that in some cases the introduction of more Hamiltonian weights on some terms can be more than offset by the reduced costs of simulating the fragment.

The number of optimization steps required for our greedy algorithm is at most  $O(m^2)$ . To see this, assume that the optimal strategy involves  $\mu$  iterations of the outer loop for  $\mu \in \Omega(m)$  and assume that the inner loop optimization requires  $\nu$  iterations. Since  $\text{COST} \geq 1$  it holds that  $\Gamma_{\max} \leq \sum_j |\alpha'_j| - \sum_j |\alpha'_j - \beta_j|$ . Assume that  $|\sum_j |\alpha'_j| - \sum_j |\alpha'_j - \beta_j| < |\alpha'|_\infty$ . In this case, by assumption there exists a trivial solution that outperforms this where the largest term is simulated in isolation at cost 1. Therefore we must have that  $|\sum_j |\alpha'_j| - \sum_j |\alpha'_j - \beta_j| \geq |\alpha'|_\infty$ . Then from standard norm inequalities we have that  $|\alpha'|_\infty \geq |\alpha'|_1/m$ . Thus the one-norm of the vector is given by a first-order difference equation of the form  $|\alpha^{(j+1)}|_1 \leq (1 - 1/m)|\alpha^{(j)}|_1$ . The general solution to this is  $(1 - 1/m)^j |\alpha|_1$  which is  $\epsilon$  for  $j \in O(\log(1/\epsilon)/\log(1/(1 - 1/m))) \in O(m \log(1/\epsilon))$ . This implies that  $\mu \in O(m \log(1/\epsilon))$ . Next  $\nu$  is the maximum number of iterations for the inner loop. Since each iteration continues until the total number of terms remaining is reduced by one we have that  $\nu \in O(m)$ . Thus the total number of iteration steps is  $\mu\nu \in O(m^2 \log(1/\epsilon))$ . This shows that the algorithm scales polynomially with the number of terms if the optimization process is also efficient.

The cost of optimization can vary strongly depending on the continuity/convexity of the objective function and without making further assumptions we cannot assume that the optima over  $\vec{\beta}$  can be found in polynomial time. If we assume, however, that the optimizer works by considering one of a polynomial number of potential circuits for simulating the terms and then uses linear programming to find the optimal value of  $\vec{\beta}$ , we have that the optimization problem can be solved in polynomial time on a classical computer. Such a choice corresponds exactly to the discussion in the next sections, where we propose the use of a discrete set of optimization strategies for simulating chemistry that can then be used within Algorithm 1 to greedily find the best possible simulation circuit given these discrete set of optimizations for the value of  $\vec{\beta}$  chosen.

**Truncating Hamiltonian.** We can terminate Algorithm 1 before all terms are allocated i.e. we output  $\{H_j = h_j \sum_i P_i; j = 1, \dots, m'\}$  such that  $\sum_{j=1}^{m'} H_j = \tilde{H} \neq H$ . This leads to truncation errors in our simulation algorithm that will be present even if an algorithm such as qDRIFT is used for the simulation. We show here that if we truncate some terms of the Hamiltonian, then the error incurred is at most twice the error incurred from the complete Hamiltonian simulation by qDRIFT, given that the distance of the truncated and given Hamiltonian is at most square root of the qDRIFT simulation error. We do this because in some cases we may be able to simulate the truncated Hamiltonian with less number of gates.

Suppose we write the given Hamiltonian as follows.

$$H = \sum_{j=1}^M w_j H_j + \delta H = \tilde{H} + \delta H \quad (\|H\| \leq 1) \quad (7)$$

Here each  $H_j$  is a Hermitian matrix for which an efficient simulation circuit exists. The protocol working with the truncated Hamiltonian  $\tilde{H}$ , samples each  $H_j$  independently with probability  $p_j = \frac{w_j}{\lambda}$  (where  $\lambda = \sum_j w_j$ ), in each iteration.

The error per iteration of qDRIFT, i.e.  $\epsilon_N$ , is given by bounding the diamond distance between the channel  $\mathcal{U}_N(\rho)$  corresponding to the Hamiltonian  $H$  and the channel  $\tilde{\mathcal{E}}(\rho)$  implemented by the protocol.

**Lemma 2.1.** The error observed when there are  $N$  time-steps taken using a qDRIFT channel,  $\epsilon_N$ , as quantified by the diamond distance as a function of the truncation error in the Hamiltonian  $\delta$  is

$$\epsilon_N \leq \|\tilde{\mathcal{E}}(\rho) - \mathcal{U}_N(\rho)\| \leq \epsilon_{\text{qDRIFT}} + 2\delta\sqrt{\epsilon_{\text{qDRIFT}}}$$

where  $\epsilon_{\text{qDRIFT}} \lesssim \frac{2\lambda^2 t^2}{N^2}$  and  $\lambda = \sum_i |w_i|$ .

The proof has been given in Supplementary Method 4 (Lemma 8). Thus the total error after all repetitions is as follows.

$$\epsilon \leq N\epsilon_N \lesssim N \frac{2\lambda^2 t^2}{N^2} + 2\sqrt{2}\delta N \frac{\lambda t}{N} = \frac{2\lambda^2 t^2}{N} + 2\sqrt{2}\delta\lambda t \quad (8)$$

This shows that if  $\delta \in O(\sqrt{\epsilon_{\text{qDRIFT}}})$  then the asymptotic scaling is not impacted by the exclusion of the terms from the Hamiltonian.

**Expected cost.** Let the cost of implementing the unitary  $e^{itwL_j/N}$  be  $c_j$ . Cost can be defined in many ways, like total number of gates, number of non-Clifford gates like T or Toffoli gate, number of multi-qubit gates like CNOT, etc. In our paper we focus mainly on the number of non-Clifford gates. Let  $C_N$  be the variable denoting the cost per repetition of our protocol. Then the expected cost and the variance per repetition is as follows.

$$\mathbb{E}[C_N] = \sum_{j=1}^M p_j c_j = \frac{1}{\lambda} \sum_{j=1}^M w_j c_j = \mu_N \quad \text{and} \quad \text{Var}[C_N] = \frac{1}{\lambda^2} \left( \lambda \sum_{j=1}^M w_j c_j^2 - \left( \sum_{j=1}^M w_j c_j \right)^2 \right) = \sigma_N^2 \quad (9)$$

By Chebyshev's inequality (Supplementary Note 1) we have the following for some real number  $k > 0$ .

$$\Pr[|C_N - \mu_N| \geq k\sigma_N] \leq \frac{1}{k^2} \quad (10)$$

The cost per repetition of our protocol is a bounded variable i.e.  $a \leq C_N \leq b$ , for some real numbers  $a, b$ . If  $C$  is the variable denoting the cost of all repetitions of our protocol, then

$$\mathbb{E}[C] = N\mu_N \quad (11)$$

and since each repetition is independent, making the corresponding cost variables per repetition distributed identically and independently, so we apply Hoeffding's inequality (Supplementary Note 1) and obtain the following.

$$\Pr[|C - N\mu_N| \geq cN\mu_N] \leq 2 \exp\left(-\frac{2c^2 N^2 \mu_N^2}{N(b-a)^2}\right) = 2 \exp\left(-\frac{2c^2 N \mu_N^2}{(b-a)^2}\right) = \epsilon_c \quad [c > 0] \quad (12)$$

Thus with probability at least  $1 - \epsilon_c$ , the cost of all repetitions of the protocol is at most  $\frac{(c+1)N}{\lambda} \sum_{j=1}^M w_j c_j$ , where  $c = \frac{b-a}{\mu_N \sqrt{2N}} \log\left(\frac{2}{\epsilon_c}\right)$ .

**Error in simulation while sampling multiple Paulis.** We consider the qDRIFT protocol<sup>43</sup> for simulating Hamiltonians. If  $H = \sum_j h_j H_j$ , then in each iteration we sample  $H_j$  with probability proportional to  $h_j$  and then simulate it for a short time period. Now  $H_j$  can be a single Pauli operator or a sum of commuting Paulis, as is achieved in Algorithm 1, to optimize the cost of simulation. Here we derive a bound on the difference in simulation error for these two cases.

Let  $H_j = \sum_{i=1}^{L_j} P_i$ —sum over commuting Paulis and  $M$  be the total number of Pauli operators. So the Hamiltonian can be written as  $H = \sum_{j=1}^L h_j H_j = \sum_{j=1}^L \sum_{i=1}^{L_j} h_j P_i$ . We assume the most general case where a single Pauli can be shared between multiple commuting groups i.e.  $H_j$ .

In the first case, a group of commuting Paulis i.e. one of the  $H_j$  is selected independently with probability  $q_j = \frac{h_j}{\sum_j h_j}$ . In the second case, one single Pauli operator  $P_k$  is sampled independently with probability  $p'_k = \frac{\sum_j h_j}{\sum_{h_i, L_i} h_i}$ , where in the numerator the sum is over all the commuting Pauli groups in which  $P_k$  appears. Let  $\lambda = \sum_j h_j$  and  $\lambda' = \sum_j h_j L_j$ . We define the Liouvillian that generates unitaries

under Hamiltonian  $H_j$  and  $P_{ij}$  so that

$$\mathcal{L}_j = i(H_j\rho - \rho H_j) \quad \text{and} \quad \mathcal{L}_{ij} = i(P_{ij}\rho - \rho P_{ij}). \quad (13)$$

Thus if  $\mathcal{L} = i(H\rho - \rho H)$ , then  $\mathcal{L} = \sum_{j=1}^L h_j \mathcal{L}_j = \sum_{j=1}^L h_j \sum_{i=1}^{L_j} \mathcal{L}_{ij}$ . We define two channels  $\mathcal{E}_1 = \sum_{j=1}^L q_j e^{\tau \mathcal{L}_j}$  and  $\mathcal{E}_2 = \sum_{j=1}^L p_j \sum_{i=1}^{L_j} e^{\tau' \mathcal{L}_{ij}}$ , where  $p_j = \frac{h_j}{\lambda}$ , that evolves the super-operators  $\mathcal{L}_j$  and  $\mathcal{L}_{ij}$  for time interval  $\tau = \frac{h_j}{N}$  and  $\tau' = \frac{\lambda' t}{N}$  respectively. Here we note that for the second channel, for each Pauli  $P_{kl}$  we have expanded the sum  $p_{kl} = \sum_{j'} \frac{h_{j'}}{\lambda}$  to reflect the commuting groups in which it belongs. Thus  $\sum_{k=1}^M p_{kl} = \sum_{j=1}^L \sum_{i=1}^{L_j} p_{ij}$ . Then we can prove the following.

**Lemma 2.2.** The distance between the qDRIFT channel with single and grouped Hamiltonian terms for simulation time  $t$  using  $N$  time steps obeys

$$\|\mathcal{E}_2 - \mathcal{E}_1\| \leq \frac{4t^2\lambda'^2}{N^2}$$

The proof has been given in Supplementary Method 4 (Lemma 7).

### Optimized circuits for quantum chemistry

In this section we review quantum algorithms for quantum chemistry and design efficient circuits that are useful for quantum chemistry simulation within the Trotter–Suzuki formalism. The electronic structure problem has emerged as a central application of quantum computers in recent years, with quantum algorithms providing potential exponential speedups relative to the best known classical algorithms<sup>50,53</sup>. The electronic structure problem more specifically is, for a fixed set of positions of the nuclei, find the configuration of electrons that minimizes the total energy for a fixed number of electrons. The properties of materials, molecules and atoms at low temperatures emerge from these energies. In the non-relativistic case, the dynamics of these electrons are governed by the Coulomb Hamiltonian.

$$H = - \sum_i \frac{\nabla_i^2}{2} - \sum_{i,j} \frac{\zeta_j}{|R_j - r_i|} + \sum_{i<j} \frac{1}{|r_i - r_j|} + \sum_{i<j} \frac{\zeta_i \zeta_j}{|R_i - R_j|}$$

where we have used atomic units,  $r_i$  represent the positions of electrons,  $R_i$  represent the positions of nuclei, and  $\zeta_i$  are the charges of nuclei.

Following the strategy outlined in<sup>13</sup>, we select the second quantization and discretize the Hamiltonian by representing it within some canonical basis such as a Gaussian basis or a plane-wave basis. Under the above assumptions, the electronic Hamiltonian can be represented in terms of creation and annihilation operators as follows<sup>64,65</sup>. Each spin orbital is assigned a (distinct) qubit where the state  $|1\rangle$  corresponds to an occupied orbital and  $|0\rangle$  an unoccupied orbital. Specifically, let  $a_p^\dagger$  and  $a_p$  be the fermionic raising and lowering operators acting on spin-orbital  $p$  satisfying the anti-commutation relation  $\{a_p^\dagger, a_q\} = \delta_{pq}$  and  $\{a_p, a_q\} = \{a_p^\dagger, a_q^\dagger\} = 0$ ,

$$H = \sum_{p,q} h_{pq} a_p^\dagger a_q + \frac{1}{2} \sum_{p,q,r,s} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s \quad (14)$$

where the coefficients  $h_{pq}$ ,  $h_{pqrs}$  are determined by the discrete basis set chosen, and the sums run over the number of discretization elements or basis set for a single particle. From inspection, we can see that the number of terms in Equation (14) is  $O(N^4)$ , where  $N$  is the size of the discrete representation. The molecular orbitals are one widely used basis set. These, in turn, can be expressed as linear combinations of atomic basis functions<sup>66,67</sup>. The coefficients of this expansion are obtained by solving the set of Hartree–Fock equations that arise from the variational minimization

of the energy using a single determinant wave function. Thus in this representation the location of (indistinguishable) electrons are specified by the occupations of the discrete basis.

The Jordan–Wigner<sup>45</sup> or Bravyi–Kitaev<sup>46</sup> transformations are commonly used to convert the fermionic creation and annihilation operators into Pauli operators. For example, within the Jordan–Wigner encoding,  $a$  and  $a^\dagger$  can be written in terms of qubit operators as follows.

$$a_p = Q_{(p)} \prod_{j=0}^{p-1} Z_{(j)} = \frac{1}{2} (X_{(p)} + iY_{(p)}) \prod_{j=0}^{p-1} Z_{(j)} \quad \text{and} \quad a_p^\dagger = Q_{(p)}^\dagger \prod_{j=0}^{p-1} Z_{(j)} = \frac{1}{2} (X_{(p)} - iY_{(p)}) \prod_{j=0}^{p-1} Z_{(j)}$$

Here  $Q_{(p)}^\dagger = \frac{1}{2} (X_{(p)} - iY_{(p)})$  and  $Q_{(p)} = \frac{1}{2} (X_{(p)} + iY_{(p)})$  are the qubit creation and annihilation operators respectively.  $\prod_j Z_{(j)}$  acts as an exchange-phase factor, accounting for the anti-commutation relations of  $a$  and  $a^\dagger$ .

With these tools in place, the second-order Trotter–Suzuki approximation reads

$$e^{-iHt} = \prod_{p,q} e^{-it(h_{pq} a_p^\dagger a_q + h_{qp} a_q^\dagger a_p)} / 2 \prod_{p,q,r,s} e^{-it(h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s + h_{srqp} a_s^\dagger a_r^\dagger a_q a_p)} / 4 + O(t^2) \quad (15)$$

Such a simulation can then be performed by substituting in the Pauli representation yielded by the Jordan–Wigner transformation. Higher order versions of this are also known<sup>63</sup> that can achieve error scaling  $O(t^{2k+1})$ ; however, we do not focus on such cases here since the optimizations to the operator exponentials that we consider here will apply in all such cases.

*Optimizing two-body operator exponentials.* The two-body terms are the most common, and often the most significant, contribution to the complexity of a simulation of the Coulomb Hamiltonian in second quantization<sup>10</sup>. In this section, we consider the general two-body double excitation terms to reduce this dominant cost for simulation of chemistry, which when expressed using the Jordan–Wigner transformation, can be written as product of  $X, Y, Z$  operators as follows<sup>53</sup>. We have removed the parentheses in the subscripts, for convenience.

$$h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s + h_{srqp} a_s^\dagger a_r^\dagger a_q a_p = \left( \bigotimes_{k=s+1}^{r-1} Z_k \right) \left( \bigotimes_{k=q+1}^{p-1} Z_k \right) \left( \frac{\Re\{h_{pqrs}\}}{8} H_r + \frac{\Im\{h_{pqrs}\}}{8} H_i \right) \quad (16)$$

$$\text{where } H_r = X_s X_r X_q X_p - X_s X_r Y_q Y_p + X_s Y_r X_q Y_p + Y_s X_r X_q Y_p \\ + Y_s X_r Y_q X_p - Y_s Y_r X_q X_p + X_s Y_r Y_q X_p + Y_s Y_r Y_q Y_p$$

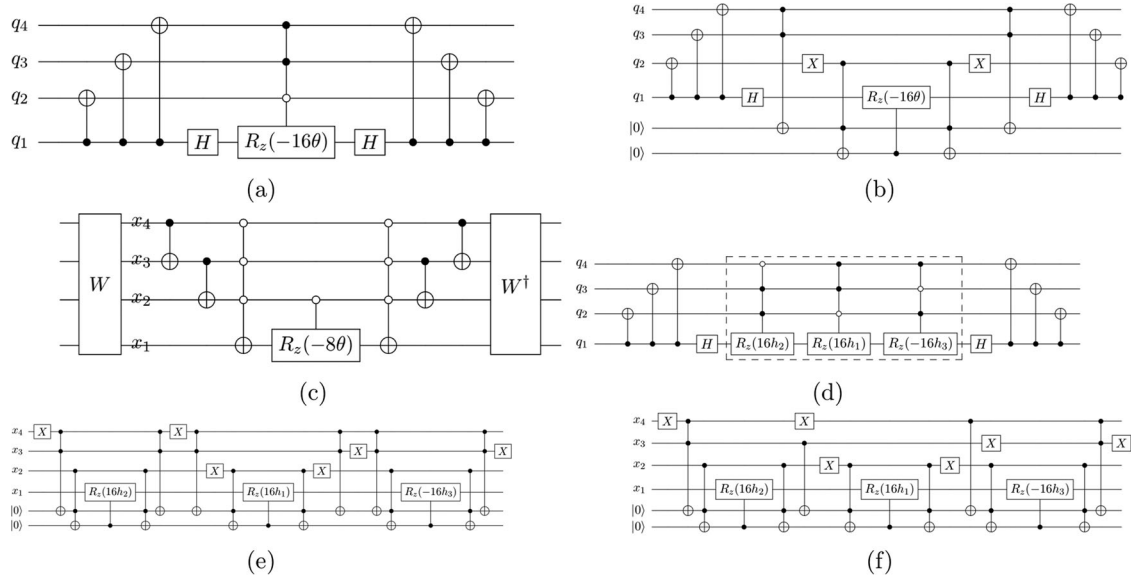
$$\text{and } H_i = Y_s X_r X_q X_p + X_s Y_r X_q X_p - X_s X_r Y_q Y_p - X_s Y_r Y_q Y_p \\ - Y_s X_r Y_q Y_p + Y_s Y_r X_q X_p + Y_s Y_r X_q Y_p + Y_s Y_r Y_q X_p \quad (17)$$

Note that if a Gaussian orbital basis is chosen then the values of  $h_{pqrs}$  are typically real, resulting in  $H_i = 0$ . We will assume in the remainder of the discussion that such terms are zero and focus our attention on only the real part of the Hamiltonian.

If we define  $h_1 = (h_{pqrs} \delta_{X_p X_s} \delta_{X_q X_r} - h_{qprs} \delta_{X_p X_r} \delta_{X_q X_s})$ ,  $h_2 = (h_{psqr} \delta_{X_p X_s} \delta_{X_q X_r} - h_{spqr} \delta_{X_p X_q} \delta_{X_r X_s})$  and  $h_3 = (h_{prsq} \delta_{X_p X_s} \delta_{X_r X_q} - h_{prqs} \delta_{X_p X_s} \delta_{X_q X_r})$ , for distinct  $p, q, r, s$  then we have the following<sup>53</sup>.

$$\frac{1}{2} \sum_{p,q,r,s} h_{pqrs} (a_p^\dagger a_q^\dagger a_r a_s + a_s^\dagger a_r^\dagger a_q a_p) \\ = \frac{1}{8} \left( \bigotimes_{k=p+1}^{q-1} \bigotimes_{l=r+1}^{s-1} Z_k \right) ((X_p X_q X_r X_s + Y_p Y_q Y_r Y_s)(-h_1 - h_2 + h_3) \\ + (X_p X_q Y_r Y_s + Y_p Y_q X_r X_s)(h_1 - h_2 + h_3) + (Y_p X_q Y_r X_s + X_p Y_q X_r Y_s)(-h_1 - h_2 - h_3) \\ + (Y_p X_q X_r Y_s + X_p Y_q Y_r X_s)(-h_1 + h_2 + h_3)) \quad (18)$$

Thus conventionally, the part of the Hamiltonian which can be expressed in the form of Equation (16), are broken down into groups of at most 8 commuting operators that act on the qubits in question. Each term is diagonalized by a Clifford circuit and the



**Fig. 1** Quantum circuit simulating  $e^{-iHt}$ . **a** Circuit when  $a_1t = a_6t = -\theta$  and  $a_0t = a_2t = \dots = a_5t = a_7t = \theta$ . **b** Circuit in **a** with the multi-controlled rotation implemented with Toffoli and controlled rotation. **c** Circuit when  $a_0t = \dots = a_7t = \theta$ . **d** Circuit when the coefficients are as in Equation 18. **e** Circuit in **d** with the multi-controlled rotations in the boxed section implemented with Toffolis and controlled rotations. **f** Circuit in **e** with some further reduction of the intermediate Toffoli gates.

evolution is performed based on this, with some  $R_z$  gates. In<sup>50</sup> the authors diagonalize all 8 terms in the simultaneous eigenbasis and parallelizes all 8  $R_z$  gates. This reduces the number of Clifford gates, depth, but comes at the cost of using extra 4 ancillae. Excluding the diagonalizing circuits on both sides they use 32 CNOTs and 8  $R_z$ . Each diagonalizing circuit uses 3 CNOT and 1 H gate. Our goal in this section is to design more efficient quantum circuits for the double excitation terms. In Table 1 we have compared the gate costs of the circuit in<sup>50</sup> with the circuits derived by us in each of the 3 cases considered by us. Fermionic SWAP gates<sup>55,68</sup> can be used to make the orbitals neighboring and hence get rid of the tensor product of Z terms. So from here on, we ignore these terms.

Let  $q_1, q_2, q_3, q_4$  be the qubits to which the fermions in the orbitals  $p, q, r, s$  are mapped respectively. We follow the technique used in<sup>50</sup>.  $W = CNOT_{(1,2)}CNOT_{(1,3)}CNOT_{(1,4)}H_{(1)}$  is the unitary diagonalizing the 8 terms in the simultaneous eigenbasis. We rewrite the Hamiltonian  $H$  with general coefficients  $a_0, \dots, a_7 \in \mathbb{R}$ . Unless mentioned, the leftmost operator acts on qubit  $q_1$ , next ones on  $q_2, q_3$  and the rightmost on qubit  $q_4$ .

$$H = a_0XXXX + a_1YYXX + a_2YXYX + a_3YXXY + a_4XYYX + a_5XYXY + a_6XXYY + a_7YYYY \quad (19)$$

Then following the arguments in<sup>50</sup> we have the following.

$$e^{-iHt} = W(e^{-ia_0ZIIIt}e^{ia_1ZZIIIt}e^{ia_2ZIZIIIt}e^{ia_3ZIZIIIt}e^{ia_4ZZIZIt}e^{ia_5ZZIZIt}e^{ia_6ZIZIZIt}e^{-a_7ZZZZIt})W^\dagger = We^{(-a_0ZIIII+a_1ZZIII+a_2ZIZII+a_3ZIZII+a_4ZZIZI+a_5ZZIZI+a_6ZIZIZ-a_7ZZZZ)It}W^\dagger \quad (20)$$

The terms in between  $W$  and  $W^\dagger$  add an overall phase  $\phi$ . We denote the state of the qubits  $q_1, \dots, q_4$  after application of  $W$  by variables  $x_1, \dots, x_4$  respectively. It is sufficient to analyse the phase when the state is in the standard basis. Consider  $e^{-ia_0ZIIIt}$  - this term contributes a phase of  $-a_0t$  if  $x_1 = 0$  and  $a_0t$  if  $x_1 = 1$ . Similarly  $e^{ia_1ZZIIIt}$  contributes a phase of  $a_1t$  if  $x_1 \oplus x_2 = 0$  and vice versa. Thus we can have the following expression for the overall phase.

$$\phi = (-1)^{x_1}a_0 + (-1)^{x_1 \oplus x_2}a_1 + (-1)^{x_1 \oplus x_3}a_2 + (-1)^{x_1 \oplus x_4}a_3 + (-1)^{x_1 \oplus x_2 \oplus x_3}a_4 + (-1)^{x_1 \oplus x_2 \oplus x_4}a_5 + (-1)^{x_1 \oplus x_3 \oplus x_4}a_6 - (-1)^{x_1 \oplus x_2 \oplus x_3 \oplus x_4}a_7 \quad (21)$$

For different values of  $a_0, \dots, a_7$  we get different value of overall phase and different circuits. We consider the following three

cases. It is easy to see that  $\phi_{x_1=1} = -\phi_{x_1=0}$ . So in all the cases below it is sufficient to calculate the phase while setting  $x_1 = 0$ .

**Case I.** Let  $a_1t = a_6t = -\theta$  and the remaining  $a_0t = a_2t = \dots = a_5t = a_7t = \theta$ . Then we can verify that  $\phi = 8\theta$  if  $x_1 = 1, x_2 = 0, x_3 = x_4 = 1$ ,  $\phi = -8\theta$  if  $x_1 = 0, x_2 = 0, x_3 = x_4 = 1$  and  $\phi = 0$  for the remaining values of  $x_1, \dots, x_4$ . Then the quantum circuit simulating  $e^{-iHt}$  is shown in Fig. 1a and b.

**Case II.** Let  $a_0t = \dots = a_7t = \theta$ . If  $x_2 = 1, x_3 \oplus x_4 = 0$  then  $\phi = 0$  and if  $x_2 = 0, x_3 \oplus x_4 = 0$  then  $\phi = (-1)^{x_3}4\theta$ . When  $x_3 \oplus x_4 = 1$  then  $\phi = -2\theta + (-1)^{x_2}2\theta$ . This is equal to  $-4\theta$  if  $x_2 = 1$ . The quantum circuit simulating  $e^{-iHt}$  is shown in Fig. 1c.

**Case III.** Let  $a_0t = a_7t = -h_1 - h_2 + h_3$ ,  $a_1t = a_6t = h_1 - h_2 + h_3$ ,  $a_2t = a_5t = -h_1 - h_2 - h_3$  and  $a_3t = a_4t = -h_1 + h_2 + h_3$ , as shown in Equation (18). It can be verified that  $\phi_{x_2=x_3=1, x_4=0} = 8h_2$ ,  $\phi_{x_2=0, x_3=x_4=1} = 8h_1$  and  $\phi_{x_2=x_4=1, x_3=0} = -8h_3$ . For every other values of  $x_2, x_3, x_4$ ,  $\phi = 0$ . The corresponding quantum circuit simulating  $e^{-iHt}$  has been shown in Fig. 1d, e, f.

We already remarked that we can ignore the product of Z terms in Equations (16) and (18) by using fermionic SWAP gates. Now if we take two Hamiltonians of the form (19) having some overlapping qubits, then we can get different Hamiltonians by rearranging the commuting Paulis. In the next few subsections we design circuits for the corresponding exponentials of these Hamiltonians. We must keep in mind that in the following subsections  $P_0 = X$  and  $P_1 = Y, i = i + 1 \text{ mod } 2$ . Table 2 summarizes the number of non-Clifford gates used to implement the various circuits. All rotation gates with  $n (>1)$  controls, can be decomposed into  $cR_z$  (single control) and NOT with  $n$  controls, each of which can be decomposed into  $n - 1$  Toffolis (as shown in Fig. 1b). We have discussed in 'Introduction' about special gadgets that can be used to further reduce the T-count of the circuits. In Fig. 1e, 1f we show how Toffolis can be reduced in segments of the circuits. Our circuits have less gates (even the Clifford gates), compared to<sup>50</sup> or the approaches where we synthesize circuit for each exponentiated Pauli and then concatenate them. In fact, we show the dependence of the circuit size or Clifford and non-Clifford gate cost on the coefficients of the commuting Paulis in the Hamiltonian expression.

**Table 2.** The first table shows the number of  $R_z$ ,  $cR_z$  and Toffoli (Toff.) pairs used to design the circuits implementing  $e^{-iHt}$ , where  $H$  are the Hamiltonians (Ham.) described in Section 'Results'.

# $q_{overlap}$	Ham.	Case I			Case II			Case III		
		# $cR_z$	# $R_z$	#Toff. pair	# $cR_z$	# $R_z$	#Toff. pair	# $cR_z$	# $R_z$	#Toff. pair
0	$H$	1	0	2	1	0	2	3	0	4
1	$H_{1y}$	2	0	2	0	2	8	6	0	24
	$H_{1x}$	2	0	2	0	2	8	6	0	6
2	$H_{21}$	-	-	-	2	0	2	4	0	4
	$H_{20}$	2	0	2	2	0	2	4	0	4
3	$H_{3y}$	2	0	2	1	0	2	6	0	6
	$H_{3x}$	2	0	2	0	2	2	6	0	6

# $q_{overlap}$	Ham.	
0	$H$	$a_0XXXX + a_1YYXX + a_2YXYX + a_3YXXY + a_4XYXX + a_5YXXY + a_6XXYY + a_7YYYY$
1	$H_{1y}$	$a_3YXXYIIII + a_5XYXYIIII + a_6XXYYIIII + a_7YYYYIIII + b_1IIIIYYXX + b_2IIIIYXYX + b_3IIIIYXXY + b_7IIIIYYYY$
	$H_{1x}$	$a_0XXXXIIII + a_1YYXXIIII + a_2YXYXIIII + a_4XYXXIIII + b_0IIIIXXXX + b_4IIIIYXYX + b_5IIIIYXXY + b_6IIIIXXYY$
2	$H_{21}$	$a_2YXYXII + a_3YXXYII + a_4XYXXII + a_5XYXYII + b_2IIYXYX + b_3IIYXYX + b_4IIYXXY + b_5IIYXXY$
	$H_{20}$	$a_0XXXXII + a_1YYXXII + a_6XXYYII + a_7YYYYII + b_0IIXXXX + b_6IIYYXX + b_1IIXXYY + b_7IIYYYY$
3	$H_{3y}$	$a_1YXXYI + a_2YXYXI + a_3YXXYI + a_7YYYYI + b_3IYXXY + b_5IXYXY + b_6IXXXY + b_7IYYYY$
	$H_{3x}$	$a_0XXXXI + a_4XYXXI + a_5YXXYI + a_6XXYYI + b_0IXXXX + b_1IYYXX + b_2IYXYX + b_4IXYXX$

Case	Coefficients
I	$a_1t = a_6t = -\theta_1; a_0t = a_2t = \dots = a_5t = a_7t = \theta_1; b_1t = b_6t = -\theta_2; a_0t = a_2t = \dots = a_5t = a_7t = \theta_2$
II	$a_0t = \dots = a_7t = \theta_1; b_0t = \dots = b_7t = \theta_2$
III	$a_0t = a_7t = -h_1 - h_2 + h_3; a_1t = a_6t = h_1 - h_2 + h_3; a_2t = a_5t = -h_1 - h_2 - h_3; a_3t = a_4t = -h_1 + h_2 + h_3; b_0t = b_7t = -g_1 - g_2 + g_3; b_1t = b_6t = g_1 - g_2 + g_3; b_2t = b_5t = -g_1 - g_2 - g_3; b_3t = b_4t = -g_1 + g_2 + g_3$

# $q_{overlap}$  is the number of overlapping qubits. Pairs of Toffoli gates are cited here, since [46] can be used in these cases to uncompute the action of the Toffoli gates in these cases. The second table summarizes the Hamiltonians and the third table summarizes the value of the coefficients for the different cases. When # $q_{overlap} = 0$  then  $\theta_1 = \theta$ , since there are no  $b$ -coefficients.

**Overlap on 1 qubit.** Previously, we provided an analysis of the circuits for cases where many of the Hamiltonian coefficients are chosen to follow regular patterns and see that the costs of the simulation can be reduced through the use of these techniques. Here we provide a more aggressive strategy wherein we combine multiple commuting terms together and find particular combinations of angles such that the simulation circuits are efficient. The results are summarized in Table 2. We consider the case when there is overlap on 1 qubit. We can have the following sets of commuting Paulis.

$$G_{1y} = \{P_i P_j P_k Y IIII, IIII Y P_a P_b P_c : i + j + k \equiv 1 \pmod 2, a + b + c \equiv 1 \pmod 2\} \tag{22}$$

$$G_{1x} = \{P_i P_j P_k X IIII, IIII X P_a P_b P_c : i + j + k \equiv 0 \pmod 2, a + b + c \equiv 0 \pmod 2\} \tag{23}$$

Without loss of generality, we assume that the leftmost operator acts on qubit  $q_1$ , next one on  $q_2$  and so on - rightmost one acts on qubit  $q_7$ . We denote a state vector as  $|Q_1 \nu Q_2\rangle$  where  $Q_1 = |q_1 q_2 q_3\rangle$ ,

$Q_2 = |q_4 q_5 q_6 q_7\rangle$  and  $\nu, q_1, \dots, q_7 \in \{0, 1\}$ . We can have the following Hamiltonian terms, expressed as sums of commuting Paulis from the above two sets.

$$H_{1y} = a_3 YXXYIIII + a_5 XYXYIIII + a_6 XXYYIIII + a_7 YYYYIIII + b_1 IIIIYYXX + b_2 IIIIYXYX + b_3 IIIIYXXY + b_7 IIIIYYYY \tag{24}$$

$$H_{1x} = a_0 XXXXIIII + a_1 YYXXIIII + a_2 YXYXIIII + a_4 XYXXIIII + b_0 IIIIXXXX + b_4 IIIIYXYX + b_5 IIIIYXXY + b_6 IIIIXXYY \tag{25}$$

**Circuit for simulating  $e^{-iH_{1y}t}$ .** Let  $W_{1y}$  be the unitary consisting of the following sequence of gates. The rightmost one is the first to be applied. With a slight abuse of notation we denote  $CNOT_{(4,1)} CNOT_{(4,2)} CNOT_{(4,3)}$  by  $CNOT_{(4,l)}$  and  $CNOT_{(4,l)} CNOT_{(4,5)} CNOT_{(4,6)} CNOT_{(4,7)}$  by  $CNOT_{(4,II)}$ .

$$W_{1y} = CNOT_{(4,I)} H_{(4)} Z_{(4)} CNOT_{(4,I)} CNOT_{(4,II)} H_{(4)} CNOT_{(4,I)}$$

In the following theorem we show that this is a diagonalizing circuit for the set of Paulis in  $G_{1y}$ .

**Theorem 2.1.** For each  $i, j, k, l, a, b, c \in \mathbb{Z}_2$ , such that  $P_i P_j P_k Y_{IIII}, IIII Y P_a P_b P_c \in G_{1y}$  we have the following.

$$\sqrt{-1}^{i+j+k+1} W_{1y} \left( Z_{(1)}^i Z_{(2)}^j Z_{(3)}^k Z_{(4)}^{IIII} \right) W_{1y}^\dagger = P_i P_j P_k Y_{IIII}$$

and  $\sqrt{-1}^{a+b+c+1} W_{1y} \left( IIII Z_{(4)} Z_{(5)}^a Z_{(6)}^b Z_{(7)}^c \right) W_{1y}^\dagger = IIII Y P_a P_b P_c$

We prove this theorem by showing that the operators on the LHS and RHS have equivalent actions on the eigenstates corresponding to an eigenbasis for the Paulis in  $G_{1y}$ . The proof of this theorem has been given in Theorem 1 of Supplementary Method 1. The eigenbasis has been shown in Lemma 1 of Supplementary Method 1.

Thus we have the following.

$$e^{-iH_{1y}t} = e^{-i(-a_3 W_{1y}(Z_{1111})W_{1y}^\dagger - a_5 W_{1y}(Z_{1211})W_{1y}^\dagger - a_6 W_{1y}(Z_{1121})W_{1y}^\dagger + a_7 W_{1y}(Z_{1112})W_{1y}^\dagger)t} \\ e^{-i(-b_1 W_{1y}(IIII Z_{11})W_{1y}^\dagger - b_2 W_{1y}(IIII Z_{12})W_{1y}^\dagger - b_3 W_{1y}(IIII Z_{13})W_{1y}^\dagger + b_4 W_{1y}(IIII Z_{14})W_{1y}^\dagger)t} \\ = W_{1y} e^{i(a_3 Z_{1111} + a_5 Z_{1211} + a_6 Z_{1121} + a_7 Z_{1112} + b_1 Z_{11} + b_2 Z_{12} + b_3 Z_{13} + b_4 Z_{14})t} W_{1y}^\dagger$$

The state of the qubits  $q_1, \dots, q_7$  after the application of  $W_{1y}$  is denoted by variables  $x_1, \dots, x_7$  respectively. We have the following expression for the overall phase incurred between  $W_{1y}$  and  $W_{1y}^\dagger$ .

$$\phi = (-1)^{x_4 \oplus x_1} a_3 t + (-1)^{x_4 \oplus x_2} a_5 t + (-1)^{x_4 \oplus x_3} a_6 t - (-1)^{x_4 \oplus x_1 \oplus x_2 \oplus x_3} a_7 t \\ + (-1)^{x_4 \oplus x_7} b_3 t + (-1)^{x_4 \oplus x_6} b_2 t + (-1)^{x_4 \oplus x_5} b_1 t - (-1)^{x_4 \oplus x_5 \oplus x_6 \oplus x_7} b_4 t$$

It is easy to check that  $\phi_{x_4} = -\phi_{\bar{x}_4}$ . We consider the following three cases and it is sufficient to check the phase values when  $x_4 = 0$ .

**Case I.** Let  $a_6 t = -\theta_1$ ,  $b_1 t = -\theta_2$ ,  $a_3 t = a_5 t = a_7 t = \theta_1$  and  $b_2 t = b_3 t = b_7 t = \theta_2$ . Following the conventions and explanations given for Case I we have the following overall phase after the application of  $W_{1y}$ . We can write  $\phi = f_1(\theta_1) + f_2(\theta_2)$ , for two functions  $f_1$  and  $f_2$ . The following can be verified.

1. If  $x_1 = x_2 = 0$  and  $x_3 = 1$  then  $\phi = 4\theta_1 + f_2(\theta_2)$ . Analogously, if  $\phi = f_1(\theta_1) + 4\theta_2$  when  $x_7 = x_6 = 0, x_5 = 1$ .
2. If  $x_1 = x_2 = 1$  and  $x_3 = 0$  then  $\phi = -4\theta_1 + f_2(\theta_2)$  and if  $x_7 = x_6 = 1, x_5 = 0$  then  $\phi = f_1(\theta_1) - 4\theta_2$ .
3. For any other values of  $x_1, x_2, x_3$ ,  $\phi = f_2(\theta_2)$  and analogously, for any other values of  $x_7, x_6, x_5$ ,  $\phi = f_1(\theta_1)$ .

A quantum circuit simulating  $e^{-iH_{1y}t}$  is shown in Fig. 2a.

**Case II.** Now we consider the case when  $a_6 t = a_3 t = a_5 t = a_7 t = \theta_1$  and  $b_1 t = b_2 t = b_3 t = b_7 t = \theta_2$ . Here also  $\phi$  can be written as

sum of two functions :  $\phi = f_1(\theta_1) + f_2(\theta_2)$ . We can make the following observations.

1. If only one of  $x_1, x_2, x_3$  is 1 then  $\phi = 2\theta_1 + f_2(\theta_2)$  and analogously, if any one of  $x_5, x_6, x_7$  is 1 then  $\phi = f_1(\theta_1) + 2\theta_2$ .
2. If any two of  $x_1, x_2, x_3$  is 1 then  $\phi = -2\theta_1 + f_2(\theta_2)$ . Similarly, if any two of  $x_5, x_6, x_7$  is 1 then  $\phi = f_1(\theta_1) - 2\theta_2$ .
3. If  $x_1 = x_2 = x_3 = 0$  then  $\phi = 2\theta_1 + f_2(\theta_2)$  and similarly, if  $x_5 = x_6 = x_7 = 0$  then  $\phi = f_1(\theta_1) + 2\theta_2$ .
4. If  $x_1 = x_2 = x_3 = 1$  then  $\phi = -2\theta_1 + f_2(\theta_2)$  and analogously, if  $x_5 = x_6 = x_7 = 1$  then  $\phi = f_1(\theta_1) - 2\theta_2$ .

A circuit simulating  $e^{-iH_{1y}t}$  in this case, has been shown in Fig. 2b.

**Case III.** Let  $a_3 t = -h_1 + h_2 + h_3, a_5 t = -h_1 - h_2 - h_3, a_6 t = h_1 - h_2 + h_3, a_7 t = -h_1 - h_2 + h_3$  and  $b_3 t = -g_1 + g_2 + g_3, b_2 t = -g_1 - g_2 - g_3, b_1 t = g_1 - g_2 + g_3, b_7 t = -g_1 - g_2 + g_3$  (Equation (18)). Let  $\mathbf{h} = (h_1, h_2, h_3)$  and  $\mathbf{g} = (g_1, g_2, g_3)$ . We can write  $\phi = f_1(\mathbf{h}) + f_2(\mathbf{g})$ . We can make the following observations.

1. If  $x_1 = x_2 = x_3$  then  $\phi = f_2(\mathbf{g})$  and analogously, if  $x_5 = x_6 = x_7$  then  $\phi = f_1(\mathbf{h})$ .
2. Suppose  $x_i = x_j$  and  $x_k \neq x_l$ , where  $i, j, k \in \{1, 2, 3\}$  and  $i \neq j \neq k$ . Then flipping the values changes the sign. For example, if  $\phi_{x_1=x_2=0, x_3=1} = f_1(\mathbf{h}) + f_2(\mathbf{g})$ , then  $\phi_{x_1=x_2=1, x_3=0} = -f_1(\mathbf{h}) + f_2(\mathbf{g})$ . Similar phenomenon occurs if  $i, j, k \in \{7, 6, 5\}$ , except this time sign of  $f_2(\mathbf{g})$  flips.. So it is sufficient to consider the case when two variables are 1.

$$\phi_{x_1=x_2=1, x_3=0} = 4h_1 + f_2(\mathbf{g}), \quad \phi_{x_7=x_6=1, x_5=0} = f_1(\mathbf{h}) + 4g_1 \\ \phi_{x_2=x_3=1, x_1=0} = 4h_2 + f_2(\mathbf{g}), \quad \phi_{x_6=x_5=1, x_7=0} = f_1(\mathbf{h}) + 4g_2 \\ \phi_{x_3=x_1=1, x_2=0} = -4h_3 + f_2(\mathbf{g}), \quad \phi_{x_5=x_7=1, x_6=0} = f_1(\mathbf{h}) - 4g_3$$

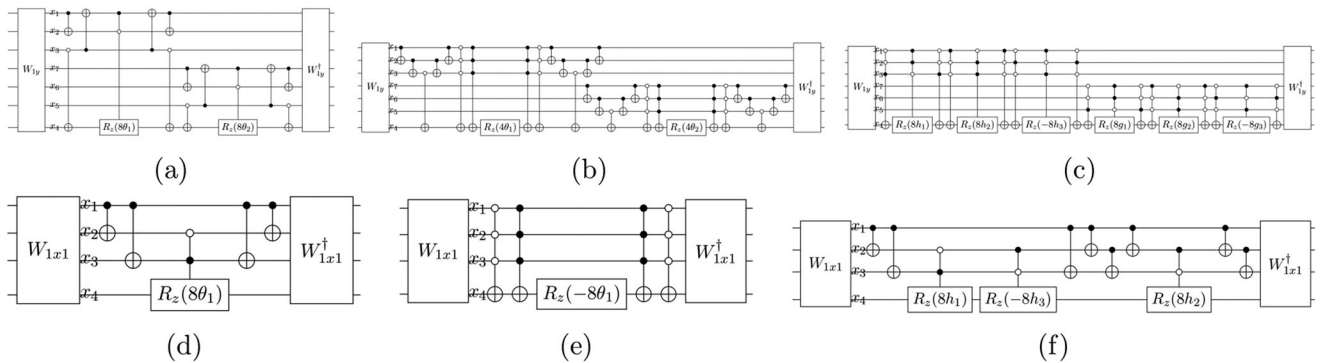
A circuit simulating  $e^{-iH_{1x}t}$  in this case, has been shown in Fig. 2c.

**Circuit for simulating  $e^{-iH_{1x}t}$ .** An eigenbasis for the Paulis in  $G_{1x}$  has been given in Lemma 2 of Supplementary Method 1. But we are unable to find out (by hand) a unitary (analogous to  $W_{1y}$ ) that diagonalizes the set of commuting Paulis in  $G_{1x}$ , as we did in the previous subsection for  $G_{1y}$ . So we divide the commuting Paulis into two groups of 4-qubit Paulis, i.e. we consider the following two sets.

$$G_{1x1} = \{P_i P_j P_k X_{IIII} : i + j + k \equiv 0 \pmod{2}\} \\ G_{1x2} = \{IIII Y P_a P_b P_c : a + b + c \equiv 0 \pmod{2}\}$$

and the following two Hamiltonians

$$H_{1x1} = a_0 XXXX_{IIII} + a_1 YYXX_{IIII} + a_2 YXYX_{IIII} + a_4 XYXX_{IIII} \\ H_{1x2} = b_0 IIIIXXXX + b_4 IIIIXYXX + b_5 IIIIXYXY + b_6 IIIIXXYX$$



**Fig. 2 Quantum circuit for  $e^{-iH_{1y}t}$  and  $e^{-iH_{1x}t}$ .** **a-c:** Circuit simulating  $e^{-iH_{1y}t}$  **a** when  $a_6 t = -\theta_1, b_1 t = -\theta_2, a_3 t = a_5 t = a_7 t = \theta_1$  and  $b_2 t = b_3 t = b_7 t = \theta_2$ ; **b** when  $a_0 t = \dots = a_7 t = \theta_1$  and  $b_0 t = \dots = b_7 t = \theta_2$ ; **c** when the coefficients are as in Equation 18. **d-f:** Circuit simulating  $e^{-iH_{1x}t}$  **d** when  $a_1 t = -\theta_1, b_6 t = -\theta_2, a_0 t = a_2 t = a_3 t = \theta_1$  and  $b_0 t = b_4 t = b_5 t = \theta_2$ ; **e** when  $a_0 t = \dots = a_7 t = \theta_1$  and  $b_0 t = \dots = b_7 t = \theta_2$ ; **f** when the coefficients are as in Equation 18.



We can use the diagonalizing circuit of<sup>50</sup> and have the following.

$$e^{-iH_{1x1}t} = W_{1x1} e^{-ia_0 \text{IIIZIIII}t} e^{ia_1 \text{ZZIZIIII}t} e^{ia_2 \text{ZIZIIII}t} e^{ia_4 \text{IZZZIIII}t} W_{1x1}^\dagger$$

$$e^{-iH_{1x2}t} = W_{1x2} e^{-ib_0 \text{IIIZIIII}t} e^{ib_4 \text{IIIZZZII}t} e^{ib_5 \text{IIIZZZII}t} e^{ib_6 \text{IIIZZZII}t} W_{1x2}^\dagger$$

where  $W_{1x1} = \text{CNOT}_{(4,1)} \text{CNOT}_{(4,2)} \text{CNOT}_{(4,3)} H_{(4)}$  and  $W_{1x2} = \text{CNOT}_{(4,5)} \text{CNOT}_{(4,6)} \text{CNOT}_{(4,7)} H_{(4)}$ , where the rightmost gate is the first one to be applied. We denote the state of the qubits  $q_1, \dots, q_4$  after the application of  $W_{1x1}$  by the variables  $x_1, \dots, x_4$  respectively. Also, the variables  $x'_4, \dots, x'_7$  denote the state of the qubits  $q_4, \dots, q_7$ , respectively, after the application of  $W_{1x2}$ . We have the following expression for the overall phase incurred between  $W_{1x1}$ ,  $W_{1x1}^\dagger$  and between  $W_{1x2}$ ,  $W_{1x2}^\dagger$ .

$$\phi_1 = -(-1)^{x_4} a_0 t + (-1)^{x_4 \oplus x_1 \oplus x_3} a_1 t + (-1)^{x_4 \oplus x_1 \oplus x_3} a_2 t + (-1)^{x_4 \oplus x_2 \oplus x_3} a_4 t$$

$$\phi_2 = -(-1)^{x'_4} b_0 t + (-1)^{x'_4 \oplus x'_5 \oplus x'_6} b_4 t + (-1)^{x'_4 \oplus x'_5 \oplus x'_7} b_5 t + (-1)^{x'_4 \oplus x'_6 \oplus x'_7} b_6 t$$

We consider the following three cases, in each of which  $\phi_{1,x_4} = -\phi_{1,x_4}$  and  $\phi_{2,x'_4} = \phi_{2,x'_4}$ .

**Case I.** Let  $a_1 t = -\theta_1$ ,  $b_6 t = -\theta_2$ ,  $a_0 t = a_2 t = a_4 t = \theta_1$  and  $b_0 t = b_4 t = b_5 t = \theta_2$ . It is easy to verify that a non-zero phase  $\phi_1 = -4\theta_1$  exists if and only if  $x_1 = x_2 \neq x_3$ . Analogously,  $\phi_2 = -4\theta_2$  if  $x'_7 = x'_6 \neq x'_5$ , else it is 0.

**Case II.** Now we consider the case when  $a_0 t = a_1 t = a_2 t = a_4 t = \theta_1$ ,  $b_0 t = b_4 t = b_5 t = b_6 t = \theta_2$ . If  $x_1 = x_2 = x_3$  then  $\phi_1 = 2\theta_1$ , else it is  $-2\theta_1$ . Similarly for  $\phi_2$ .

**Case III.** Next we consider the case where  $a_0 t = -h_1 - h_2 + h_3$ ,  $a_1 t = h_1 - h_2 + h_3$ ,  $a_2 t = -h_1 - h_2 - h_3$ ,  $a_4 t = -h_1 + h_2 + h_3$ , and  $b_0 t = -g_1 - g_2 + g_3$ ,  $b_4 t = -g_1 + g_2 + g_3$ ,  $b_5 t = -g_1 - g_2 - g_3$ ,  $b_6 t = g_1 - g_2 + g_3$ . Here, non-zero phase exists if any two of the variable have same value.

$$\phi_1(x_1 = x_2 \neq x_3) = 4h_1; \quad \phi_1(x_2 = x_3 \neq x_1) = 4h_2; \quad \phi_1(x_1 = x_3 \neq x_2) = -4h_3;$$

$$\phi_2(x'_5 = x'_6 \neq x'_7) = 4g_2; \quad \phi_2(x'_6 = x'_7 \neq x'_5) = 4g_1; \quad \phi_2(x'_5 = x'_7 \neq x'_6) = -4g_3;$$

Circuits simulating  $e^{-iH_{1x1}t}$  in Case I, II and III have been shown in Fig. 2d, e and f respectively. Circuits for  $e^{-iH_{1x2}t}$  are similar. Circuit for  $e^{-iH_{1x}t}$  in each case is obtained by concatenating the corresponding circuits.

**Overlap on 2 qubits.** In general, the more options that we have for grouping mutually commuting terms the more effective our compilation strategy will be. While the most natural case to examine is the case where all of the Hamiltonian terms act on disjoint sets of qubits, Hamiltonian terms can commute if they overlap on only two qubits as well. For example, we can have the following sets of commuting Pauli operations

$$G_{21} = \{P_k P_l P_j P_i \text{II}, \text{II} P_l P_j P_k P_i : i+j \equiv 1 \pmod{2}, k, l = i, j \text{ or } \bar{i}, \bar{j} \text{ respectively}\} \quad (26)$$

$$G_{20} = \{P_k P_l P_j P_i \text{II}, \text{II} P_l P_j P_k P_i : i+j \equiv 0 \pmod{2}, k, l = i, j \text{ or } \bar{i}, \bar{j} \text{ respectively}\} \quad (27)$$

Without loss of generality, we assume that the leftmost operator acts on qubit  $q_1$ , next one on  $q_2$  and so on - rightmost one acts on qubit  $q_6$ . We denote a state vector as  $|Q_1 Q_2 Q_3\rangle$  where  $Q_1 = |q_1 q_2\rangle$ ,  $Q_2 = |q_3 q_4\rangle$  and  $Q_3 = |q_5 q_6\rangle$  are the first, second and third pairs of qubits respectively. We can have the following Hamiltonian terms, expressed as sums of commuting Paulis from the above two sets.

$$H_{21} = a_2 \text{YXXYII} + a_3 \text{YXXYII} + a_4 \text{YXXYII} + a_5 \text{YXXYII} \\ + b_2 \text{IIYXXY} + b_3 \text{IIYXXY} + b_4 \text{IIYXXY} + b_5 \text{IIYXXY} \quad (28)$$

$$H_{20} = a_0 \text{XXXXII} + a_1 \text{YXXYII} + a_6 \text{XXYYII} + a_7 \text{YYYYII} \\ + b_0 \text{IIXXXX} + b_6 \text{IIYXXY} + b_1 \text{IIXXYY} + b_7 \text{IIYYYY} \quad (29)$$

**Circuit for simulating  $e^{-iH_{21}t}$ .** As before our simulation strategy involves diagonalizing the Hamiltonian using a Clifford circuit and then build Let  $W_1$  be the unitary consisting of the following sequence of gates. The rightmost one is the first to be applied.

$$W_1 = \text{CNOT}_{(3,1)} \text{CNOT}_{(3,4)} H_{(3)} Z_{(3)} \text{CNOT}_{(3,1)} \text{CNOT}_{(3,5)} H_{(3)} \text{CNOT}_{(3,1)}$$

The following theorem shows that this is a diagonalizing circuit for the set of Paulis in  $G_{21}$ .

**Theorem 2.2.** For each  $i, j, k, l \in \mathbb{Z}_2$ , such that  $P_k P_l P_j P_i \text{II}, \text{II} P_l P_j P_k P_i \in G_{21}$  we have the following.

$$\sqrt{-1}^{i+j+k+l} W_1 \left( Z_{(1)}^i Z_{(2)}^j Z_{(3)}^k Z_{(4)}^l \text{II} \right) W_1^\dagger = P_k P_l P_j P_i \text{II}$$

and

$$\sqrt{-1}^{i+j+k+l} W_1 \left( \text{II} Z_{(3)}^i Z_{(4)}^j Z_{(5)}^k Z_{(6)}^l \right) W_1^\dagger = \text{II} P_l P_j P_k P_i$$

The proof is similar to Theorem 2.1 and has been given in Supplementary Method 2. Theorem 2.2 then gives us the following.

$$e^{-iH_{21}t} = e^{-i(-a_2 W_1 (Z_{12111}) W_1^\dagger - a_3 W_1 (Z_{12211}) W_1^\dagger - a_4 W_1 (Z_{12111}) W_1^\dagger - a_5 W_1 (Z_{12211}) W_1^\dagger) t} \\ \cdot e^{-i(-b_2 W_1 (IIZ1211) W_1^\dagger - b_3 W_1 (IIZ2211) W_1^\dagger - b_4 W_1 (IIZ1211) W_1^\dagger - b_5 W_1 (IIZ2211) W_1^\dagger) t} \\ = W_1 e^{ia_2 Z_{12111}t} e^{ia_3 Z_{12211}t} e^{ia_4 Z_{12111}t} e^{ia_5 Z_{12211}t} e^{ib_2 IIZ1211t} e^{ib_3 IIZ2211t} e^{ib_4 IIZ1211t} e^{ib_5 IIZ2211t} W_1^\dagger$$

We denote the state of the qubits  $q_1, \dots, q_6$  after the application of  $W_1$  by the variables  $x_1, \dots, x_6$  respectively. We have the following expression for the overall phase incurred between  $W_1$  and  $W_1^\dagger$ .

$$\phi = (-1)^{x_3 \oplus x_1} a_2 t + (-1)^{x_3 \oplus x_2} a_4 t + (-1)^{x_3 \oplus x_4 \oplus x_1} a_3 t + (-1)^{x_3 \oplus x_4 \oplus x_2} a_5 t \\ + (-1)^{x_3 \oplus x_5} b_2 t + (-1)^{x_3 \oplus x_6} b_4 t + (-1)^{x_3 \oplus x_4 \oplus x_5} b_3 t + (-1)^{x_3 \oplus x_4 \oplus x_6} b_5 t$$

It is easy to check that  $\phi_{x_3} = -\phi_{x_3}$ . We consider the following cases and it is sufficient to check the phase values when  $x_3 = 0$ .

**Case I (II).** We consider the case when  $a_2 t = a_3 t = a_4 t = a_5 t = \theta_1$ ,  $b_2 t = b_3 t = b_4 t = b_5 t = \theta_2$ . There are no  $a_1, a_6, b_1, b_6$  in the expression of the Hamiltonian. So, for consistency with the previous and following subsection, we can consider this as either Case I or II.

We can write  $\phi = f_1(\theta_1) + f_2(\theta_2)$ . We can verify that when  $q_1 \oplus q_2 = q_4 = 0$  then  $\phi = (-1)^{q_1} 4\theta_1 + f_2(\theta_2)$  and analogously, when  $q_5 \oplus q_6 = q_4 = 0$  then  $\phi = f_1(\theta_1) + (-1)^{q_5} 4\theta_2$ . For all other values of  $q_1, q_2, q_4$ ,  $\phi = f_2(\theta_2)$  and for all other values of  $q_5, q_6, q_4$ ,  $\phi = f_1(\theta_1)$ . A quantum circuit for simulating  $e^{-iH_{21}t}$  in this case has been shown in Fig. 3a.

**Case III.** Let  $a_2 t = a_5 t = -h_1 - h_2 - h_3$  and  $a_3 t = a_4 t = -h_1 + h_2 + h_3$ ,  $b_2 t = b_5 t = -g_1 - g_2 - g_3$  and  $b_3 t = b_4 t = -g_1 + g_2 + g_3$ . We can write  $\phi = f_1(\mathbf{h}) + f_2(\mathbf{g})$ , where  $\mathbf{h} = (h_1, h_2, h_3)$  and  $\mathbf{g} = (g_1, g_2, g_3)$ . When  $q_1 \oplus q_2 = q_4 = 0$ , then  $\phi = -(-1)^{q_1} 4h_1 + f_2(\mathbf{g})$  and when  $q_1 \oplus q_2 = q_4 = 1$ , then  $\phi = -(-1)^{q_1} 4(h_1 + h_3) + f_2(\mathbf{g})$ . For every other values of  $q_1, q_2, q_4$ ,  $\phi = f_2(\mathbf{g})$ . Similarly, when  $q_5 \oplus q_6 = q_4 = 0$ , then  $\phi = f_1(\mathbf{h}) - (-1)^{q_5} 4g_1$  and when  $q_5 \oplus q_6 = q_4 = 1$ , then  $\phi = f_1(\mathbf{h}) - (-1)^{q_5} 4(g_1 + g_3)$ . For every other values of  $q_5, q_6, q_4$ ,  $\phi = f_1(\mathbf{h})$ . A quantum circuit simulating  $e^{-iH_{21}t}$  in this case has been shown in Fig. 3b.

**Circuit for simulating  $e^{-iH_{20}t}$ .** An eigenbasis for the Paulis in  $G_{20}$  has been shown in Lemma 4 of Supplementary Method 2. But since we have been unable to derive a diagonalizing circuit, so we divide the commuting Paulis into two groups of 4-qubit Paulis as follows.

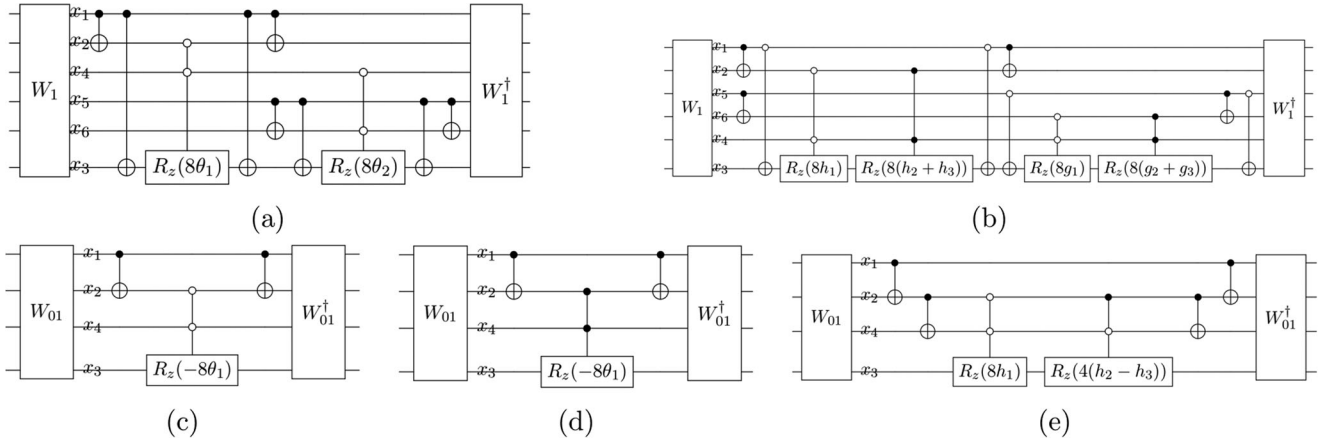
$$G_{201} = \{P_k P_l P_j P_i \text{II} : i+j, k+l \equiv 0 \pmod{2}\}$$

$$G_{202} = \{\text{II} P_l P_j P_k P_i : i+j, k+l \equiv 0 \pmod{2}\}$$

We get the following two Hamiltonians.

$$H_{201} = a_0 \text{XXXXII} + a_1 \text{YXXYII} + a_6 \text{XXYYII} + a_7 \text{YYYYII}$$

$$H_{202} = b_0 \text{IIXXXX} + b_6 \text{IIYXXY} + b_1 \text{IIXXYY} + b_7 \text{IIYYYY}$$



**Fig. 3** Quantum circuit for  $e^{-iH_{201}t}$  and  $e^{-iH_{202}t}$ . **(a, b)**: Circuit simulating  $e^{-iH_{201}t}$  **(a)** when  $a_2t = a_3t = a_4t = a_5t = \theta_1$  and  $b_2t = b_3t = b_4t = b_5t = \theta_2$ ; **(b)** when the coefficients are as in Equation 18. **(c–e)** Circuit simulating  $e^{-iH_{202}t}$  **(c)** when  $a_1t = a_6t = -\theta_1$ ,  $b_1t = b_6t = -\theta_2$ ,  $a_0t = a_7t = \theta_1$  and  $b_0t = b_7t = \theta_2$ ; **(d)** when  $a_0t = \dots = a_7t = \theta_1$  and  $b_0t = \dots = b_7t = \theta_2$ ; **(e)** when the coefficients are as in Equation 18.

Using the diagonalizing circuit of<sup>50</sup> and have the following.

$$e^{-iH_{201}t} = W_{01}e^{-ia_0\mathbb{I}ZZ\mathbb{I}\mathbb{I}\mathbb{I}t}e^{ia_1ZZZ\mathbb{I}\mathbb{I}t}e^{ia_6\mathbb{I}ZZ\mathbb{I}\mathbb{I}t}e^{-ia_7ZZZZ\mathbb{I}\mathbb{I}t}W_{01}^\dagger$$

$$e^{-iH_{202}t} = W_{02}e^{-ib_0\mathbb{I}ZZ\mathbb{I}\mathbb{I}\mathbb{I}t}e^{ib_6\mathbb{I}ZZ\mathbb{I}\mathbb{I}t}e^{ib_1\mathbb{I}ZZ\mathbb{I}\mathbb{I}t}e^{-ib_7\mathbb{I}ZZZZ\mathbb{I}\mathbb{I}t}W_{02}^\dagger$$

where  $W_{01} = CNOT_{(3,1)}CNOT_{(3,2)}CNOT_{(3,4)}H_{(3)}$  and  $W_{02} = CNOT_{(3,4)}CNOT_{(3,5)}CNOT_{(3,6)}H_{(3)}$ , where the rightmost gate is the first one to be applied. We denote the state of the qubits  $q_1, \dots, q_4$  and  $q_3, \dots, q_6$  after the application of  $W_{01}$  and  $W_{02}$  by the variables  $x_1, \dots, x_4$  and  $x'_3, \dots, x'_6$  respectively. We have the following expression for the overall phase incurred between  $W_{01}$ ,  $W_{01}^\dagger$  and between  $W_{02}$ ,  $W_{02}^\dagger$ .

$$\phi_1 = -(-1)^{x_3}a_0t + (-1)^{x_3 \oplus x_1 \oplus x_2}a_1t + (-1)^{x_3 \oplus x_4}a_6t - (-1)^{x_3 \oplus x_4 \oplus x_1 \oplus x_2}a_7t$$

$$\phi_2 = -(-1)^{x'_3}b_0t + (-1)^{x'_3 \oplus x'_4}b_6t + (-1)^{x'_3 \oplus x'_5 \oplus x'_6}b_1t - (-1)^{x'_3 \oplus x'_4 \oplus x'_5 \oplus x'_6}b_7t$$

In all the cases considered below it is easy to verify that  $\phi_{1,x_3} = -\phi_{1,x'_3}$  and  $\phi_{2,x'_3} = -\phi_{2,x_3}$ . So it is enough to consider  $x_3 = x'_3 = 0$ .

**Case I.** Assume  $a_1t = a_6t = -\theta_1$ ,  $b_1t = b_6t = -\theta_2$ ,  $a_0t = a_7t = \theta_1$  and  $b_0t = b_7t = \theta_2$ . If  $x_1 \oplus x_2 = x_4 = 0$  then  $\phi_1 = -4\theta_1$ , else it is 0. Similar conclusions follow for  $\phi_2$  if we replace  $x_1, x_2, x_3, x_4$  by  $x'_6, x'_5, x'_3, x'_4$  respectively.

**Case II.** Let  $a_0t = a_1t = a_6t = a_7t = \theta_1$ ,  $b_0t = b_1t = b_6t = b_7t = \theta_2$ . If  $x_1 \oplus x_2 = x_4 = 1$  then  $\phi_1 = -4\theta_1$ , else it is 0. Similar conclusions follow for  $\phi_2$  if we replace  $x_1, x_2, x_3, x_4$  by  $x'_6, x'_5, x'_3, x'_4$  respectively.

**Case III.** Let  $a_0t = a_7t = -h_1 - h_2 + h_3$ ,  $a_1t = a_6t = h_1 - h_2 + h_3$  and  $b_0t = b_7t = -g_1 - g_2 + g_3$ ,  $b_1t = b_6t = g_1 - g_2 + g_3$ . If  $x_1 \oplus x_2 = x_4 = 0$  then  $\phi_1 = 4h_1$  and if  $x_1 \oplus x_2 = x_4 = 1$  then  $\phi_2 = 2(h_2 - h_3)$ . Similar conclusions follow for  $\phi_2$  if we replace  $x_1, x_2, x_4$  by  $x'_6, x'_5, x'_4$  respectively.

Circuits simulating  $e^{-iH_{201}t}$  in Case I, II and III have been shown in Fig. 3c, d and e respectively. Circuits for  $e^{-iH_{202}t}$  are similar. Circuit for  $e^{-iH_{20}$  in each of these cases is obtained by concatenating the corresponding circuits.

**Overlap on 3 qubits.** Now we consider the case when there is overlap on 3 qubits. We can have the following sets of commuting Paulis.

$$G_{3y} = \{YP_iP_jP_k\mathbb{I}, \mathbb{I}P_iP_jP_kY : i + j + k \equiv 1 \pmod{2}\} \quad (30)$$

$$G_{3x} = \{XP_iP_jP_k\mathbb{I}, \mathbb{I}P_iP_jP_kX : i + j + k \equiv 0 \pmod{2}\} \quad (31)$$

Without loss of generality, we assume that the leftmost operator acts on qubit  $q_1$ , next one on  $q_2$  and so on - rightmost one acts on

qubit  $q_5$ . We denote a state vector as  $|Q_1Q_2Q_3Q_4Q_2\rangle$  where  $Q_1 = |q_1\rangle$ ,  $Q_2 = |q_5\rangle$  and  $q_1, \dots, q_5 \in \{0, 1\}$ . We can have the following Hamiltonian terms, expressed as sums of commuting Paulis from the above two sets.

$$H_{3y} = a_1YXX\mathbb{I} + a_2YXY\mathbb{I} + a_3YXXY\mathbb{I} + a_7YYYY\mathbb{I} + b_3\mathbb{I}YXXY + b_5\mathbb{I}XYXY + b_6\mathbb{I}XXYY + b_7\mathbb{I}YYYY \quad (32)$$

$$H_{3x} = a_0XXXX\mathbb{I} + a_4XYYX\mathbb{I} + a_5YXXY\mathbb{I} + a_6XXYY\mathbb{I} + b_0\mathbb{I}XXXX + b_1\mathbb{I}YYXX + b_2\mathbb{I}YXYX + b_4\mathbb{I}XYYX \quad (33)$$

**Circuit for simulating  $e^{-iH_{3y}t}$ .** Let  $W_{3y}$  be the unitary consisting of the following sequence of gates. The rightmost one is the first to be applied. With a slight abuse of notation we denote  $CNOT_{(c,t_1)}CNOT_{(c,t_2)}CNOT_{(c,t_3)}\dots$  by  $CNOT_{(c;t_1,t_2,t_3,\dots)}$  (multi-target CNOT).

$$W_{3y} = CNOT_{(2;1,3,4)}H_{(2)}Z_{(2)}CNOT_{(2;1,5)}H_{(2)}CNOT_{(2,1)}$$

**Theorem 2.3.** For each  $i, j, k \in \mathbb{Z}_2$ , such that  $YP_iP_jP_k\mathbb{I}, \mathbb{I}P_iP_jP_kY \in G_{3y}$  we have the following.

$$\sqrt{-1}^{i+j+k+1}W_{3y}\left(Z_{(1)}Z_{(2)}Z_{(3)}^jZ_{(4)}^k\mathbb{I}\right)W_{3y}^\dagger = YP_iP_jP_k\mathbb{I}$$

$$\text{and } \sqrt{-1}^{i+j+k+1}W_{3y}\left(\mathbb{I}Z_{(2)}Z_{(3)}^jZ_{(4)}^kZ_{(5)}\right)W_{3y}^\dagger = \mathbb{I}P_iP_jP_kY$$

The proof is similar to Theorem 2.1 and has been shown in Supplementary Method 3. Thus we have the following.

$$e^{-iH_{3y}t} = e^{-i(-a_1W_{3y}(ZZ\mathbb{I}\mathbb{I}\mathbb{I})W_{3y}^\dagger - a_2W_{3y}(ZZZ\mathbb{I}\mathbb{I})W_{3y}^\dagger - a_3W_{3y}(ZZIZ\mathbb{I})W_{3y}^\dagger + a_7W_{3y}(ZZZZ\mathbb{I})W_{3y}^\dagger)t}$$

$$\cdot e^{-i(-b_3W_{3y}(LZ\mathbb{I}Z)W_{3y}^\dagger - b_5W_{3y}(LZZIZ)W_{3y}^\dagger - b_6W_{3y}(LZIZZ)W_{3y}^\dagger + b_7W_{3y}(LZZZZ)W_{3y}^\dagger)t}$$

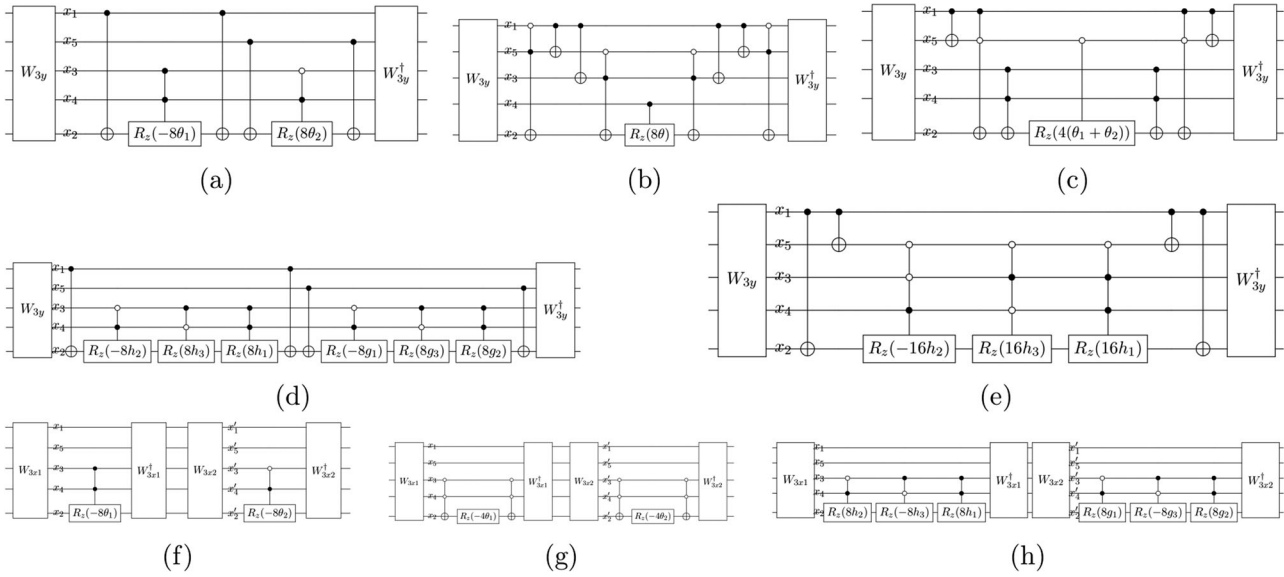
$$= W_{3y}e^{ia_1ZZ\mathbb{I}\mathbb{I}\mathbb{I}t}e^{ia_2ZZZ\mathbb{I}\mathbb{I}t}e^{ia_3ZZIZ\mathbb{I}t}e^{-ia_7ZZZZ\mathbb{I}t}e^{ib_3LZ\mathbb{I}Zt}e^{ib_5LZZIZt}e^{ib_6LZIZZt}e^{-ib_7LZZZZt}W_{3y}^\dagger$$

We denote the state of the qubits  $q_1, \dots, q_5$  after the application of  $W_{3y}$  by the variables  $x_1, \dots, x_5$  respectively. We have the following expression for the overall phase incurred between  $W_{3y}$  and  $W_{3y}^\dagger$ .

$$\phi = (-1)^{x_2 \oplus x_1}a_1 + (-1)^{x_2 \oplus x_1 \oplus x_3}a_2 + (-1)^{x_2 \oplus x_1 \oplus x_4}a_3 - (-1)^{x_2 \oplus x_1 \oplus x_3 \oplus x_4}a_7 + (-1)^{x_2 \oplus x_5}b_3 + (-1)^{x_2 \oplus x_5 \oplus x_3}b_5 + (-1)^{x_2 \oplus x_5 \oplus x_4}b_6 - (-1)^{x_2 \oplus x_5 \oplus x_3 \oplus x_4}b_7$$

It is easy to verify that  $\phi_{x_2}^- = -\phi_{x_2}$ . We consider the following cases and it is sufficient to check the phase values when  $x_2 = 0$ .

**Case I.** We consider the case when  $a_1t = -\theta_1$ ,  $b_6t = -\theta_2$ ,  $a_2t = a_3t = a_7t = \theta_1$  and  $b_3t = b_5t = b_7t = \theta_2$ . We can write



**Fig. 4** Quantum circuit for  $e^{-iH_{3y}t}$  and  $e^{-iH_{3x1}t}$ . **a–e**: Circuit simulating  $e^{-iH_{3y}t}$  (**a**) when  $a_1t = -\theta_1, b_6t = -\theta_2, a_2t = a_3t = a_7t = \theta_1$  and  $b_3t = b_5t = b_7t = \theta_2$ ; (**b**) with the coefficient values in (**a**), except here  $\theta_1 = \theta_2$ ; (**c**) when  $a_1t = a_2t = a_3t = a_7t = \theta_1$  and  $b_3t = b_5t = b_6t = b_7t = \theta_2$ ; (**d**) when the coefficients are as in Equation 18; (**e**) Circuit with the coefficients in (**d**), except in this case  $h_2 = g_1, h_3 = g_3, h_1 = g_2$ . **f–h**: Circuit simulating  $e^{-iH_{3x1}t}$  (**f**) when  $a_6t = -\theta_1, b_1t = -\theta_2, a_0t = a_5t = a_4t = \theta_1$  and  $b_0t = b_2t = b_4t = \theta_2$ ; (**g**) when  $a_0t = \dots = a_7t = \theta_1$  and  $b_0t = \dots = b_7t = \theta_2$ ; (**h**) when the coefficients are as in Equation 18.

$\phi = f_1(\theta_1) + f_2(\theta_2)$ . It can be verified that  $\phi_{\bar{x}_1, \bar{x}_5} = -f(\theta_1) + f(\theta_2)$  and  $\phi_{x_1, \bar{x}_5} = f(\theta_1) - f(\theta_2)$ . So we concentrate on  $x_1 = x_5 = 0$ . If  $x_3 = x_4 = 1$  then  $\phi = -4\theta_1$  and if  $x_3 = 0, x_4 = 1$  then  $\phi = 4\theta_2$ . A quantum circuit simulating  $e^{-iH_{3y}t}$  has been shown in Fig. 4a. If  $\theta_1 = \theta_2$  then we can have a further reduction of controlled rotation gates, as shown in Fig. 4b.

**Case II.** Next we consider the case when  $a_1t = a_2t = a_3t = a_7t = \theta_1, b_6t = b_3t = b_5t = b_7t = \theta_2$ . In this case  $\phi = 0$  whenever  $x_1 \oplus x_5 = 1$ . Else, as before  $\phi_{\bar{x}_1, \bar{x}_5} = -f(\theta_1) + f(\theta_2)$  and  $\phi_{x_1, \bar{x}_5} = f(\theta_1) - f(\theta_2)$ . So it is enough to consider  $x_1 = x_5 = 0$ . When  $x_3 = x_4 = 1$  then  $\phi = -2(\theta_1 + \theta_2)$ , else  $\phi = 2(\theta_1 + \theta_2)$ . Thus we can have a quantum circuit simulating  $e^{-iH_{3y}t}$ , as shown in Fig. 4c.

**Case III.** Now we consider the case when  $a_1t = h_1 - h_2 + h_3, a_2t = -h_1 - h_2 - h_3, a_3t = -h_1 + h_2 + h_3, a_7t = -h_1 - h_2 + h_3$  and  $b_3t = -g_1 + g_2 + g_3, b_5t = -g_1 - g_2 - g_3, b_6t = g_1 - g_2 + g_3, b_7t = -g_1 - g_2 + g_3$ . If we denote  $\mathbf{h} = (h_1, h_2, h_3)$  and  $\mathbf{g} = (g_1, g_2, g_3)$ , then we can write  $\phi = f(\mathbf{h}) + f(\mathbf{g})$ . Here too,  $\phi_{\bar{x}_1, \bar{x}_5} = -f(\mathbf{h}) + f(\mathbf{g})$  and  $\phi_{x_1, \bar{x}_5} = f(\mathbf{h}) - f(\mathbf{g})$ . So let us consider  $x_1 = x_5 = 0$ . Then we have the following phase values.

$$\phi_{x_3=x_4=0} = 0, \quad \phi_{x_3=0, x_4=1} = -4(h_2 + g_1), \quad \phi_{x_3=1, x_4=0} = 4(h_3 + g_3), \quad \phi_{x_3=x_4=1} = 4(h_1 + g_2)$$

A circuit simulating  $e^{-iH_{3y}t}$  in this case has been shown in Fig. 4d. If  $h_2 = g_1, h_3 = g_3, h_1 = g_2$  then we can have a simpler circuit, as shown in Fig. 4e.

**Circuit for simulating  $e^{-iH_{3x1}t}$ .** The diagonalizing transformation for the Pauli operators in  $G_{3x}$  is shown in Lemma 6 of Supplementary Method 3. Since we have been unable to find a diagonalizing circuit, so we divide the commuting Paulis into two groups of 4-qubit Paulis,

$$G_{3x1} = \{XP_iP_jP_k\mathbb{I} : i + j + k \equiv 0 \pmod{2}\}$$

$$G_{3x2} = \{\mathbb{I}P_iP_jP_kX : i + j + k \equiv 0 \pmod{2}\}$$

and have the following two Hamiltonians.

$$H_{3x1} = a_0XXXX\mathbb{I} + a_4XXYY\mathbb{I} + a_5XYXY\mathbb{I} + a_6XXYY\mathbb{I}$$

$$H_{3x2} = b_0\mathbb{I}XXXX + b_1\mathbb{I}YYXX + b_2\mathbb{I}YXYX + b_4\mathbb{I}XXYY$$

Using the diagonalizing circuit of<sup>50</sup> and have the following.

$$e^{-iH_{3x1}t} = W_{3x1}e^{-ia_0\mathbb{I}ZZ\mathbb{I}\mathbb{I}\mathbb{I}t}e^{ia_6\mathbb{I}ZZ\mathbb{I}\mathbb{I}t}e^{ia_5\mathbb{I}Z\mathbb{I}Z\mathbb{I}t}e^{ia_4\mathbb{I}ZZ\mathbb{I}\mathbb{I}t}W_{3x1}^\dagger$$

$$e^{-iH_{3x2}t} = W_{3x2}e^{-ib_0\mathbb{I}Z\mathbb{I}\mathbb{I}\mathbb{I}t}e^{ib_1\mathbb{I}ZZ\mathbb{I}\mathbb{I}t}e^{ib_2\mathbb{I}Z\mathbb{I}Z\mathbb{I}t}e^{ib_4\mathbb{I}ZZ\mathbb{I}\mathbb{I}t}W_{3x2}^\dagger$$

where  $W_{3x1} = CNOT_{(2; 1, 3, 4)}H_{(2)}$  and  $W_{3x2} = CNOT_{(2; 3, 4, 5)}H_{(2)}$ , where the rightmost gate is the first one to be applied. We denote the state of the qubits  $q_1, \dots, q_4$  and  $q_2, \dots, q_5$  after the application of  $W_{3x1}$  and  $W_{3x2}$  by the variables  $x_1, \dots, x_4$  and  $x'_2, \dots, x'_5$  respectively. We have the following expression for the overall phase incurred between  $W_{3x1}, W_{3x1}^\dagger$  and between  $W_{3x2}, W_{3x2}^\dagger$ .

$$\phi_1 = -(-1)^{x_2}a_0t + (-1)^{x_2 \oplus x_3 \oplus x_4}a_6t + (-1)^{x_2 \oplus x_4}a_5t + (-1)^{x_2 \oplus x_3}a_4t$$

$$\phi_2 = -(-1)^{x'_2}b_0t + (-1)^{x'_2 \oplus x'_3}b_1t + (-1)^{x'_2 \oplus x'_4}b_2t + (-1)^{x'_2 \oplus x'_3 \oplus x'_4}b_4t$$

In all the cases considered below it is easy to verify that  $\phi_{1, \bar{x}_2} = -\phi_{1, x_2}$  and  $\phi_{2, \bar{x}'_2} = -\phi_{2, x'_2}$ . So it is enough to consider  $x_2 = x'_2 = 0$ .

**Case I.** We consider the case when  $a_6t = -\theta_1, b_1t = -\theta_2, a_0t = a_5t = a_4t = \theta_1$  and  $b_0t = b_4t = b_2t = \theta_2$ .  $\phi_1 = -4\theta_1$  when  $x_3 = x_4 = 1$ , else it is 0. And  $\phi_2 = -4\theta_2$  when  $x'_3 = 0, x'_4 = 1$ , else it is 0.

**Case II.** Let  $a_0t = a_6t = a_5t = a_4t = \theta_1, b_0t = b_4t = b_2t = \theta_2$ .  $\phi_1 = 2\theta_1$  when  $x_3 = x_4 = 0$ , else  $\phi_1 = -2\theta_1$ . Similarly for  $\phi_2$ .

**Case III.** Assume  $a_0t = -h_1 - h_2 + h_3, a_6t = h_1 - h_2 + h_3, a_5t = -h_1 - h_2 - h_3, a_4t = -h_1 + h_2 + h_3$ , and  $b_0t = -g_1 - g_2 + g_3, b_4t = -g_1 + g_2 + g_3, b_2t = -g_1 - g_2 - g_3, b_1t = g_1 - g_2 + g_3$ . We have the following phases.

$$\phi_1(x_3 = 0, x_4 = 1) = 4h_2; \quad \phi_1(x_3 = 1, x_4 = 0) = -4h_3; \quad \phi_1(x_3 = x_4 = 1) = 4h_1;$$

$$\phi_2(x'_3 = 0, x'_4 = 1) = 4g_1, \quad \phi_2(x'_3 = 1, x'_4 = 0) = -4g_3; \quad \phi_2(x'_3 = x'_4 = 1) = 4g_2;$$

Circuits simulating  $e^{-iH_{3x1}t}$  in Case I, II and III have been shown in Fig. 4f, g and h respectively. Circuits for  $e^{-iH_{3x2}t}$  are similar. Circuit for  $e^{-iH_{3x}t}$  in each case is obtained by concatenating the corresponding circuits.

### Circuit for arbitrary exponentiated Hamiltonians

Our previous discussion focuses on the case of fermionic simulation within a Jordan–Wigner representation using Hamiltonian terms

that are fermionically swapped to be adjacent to each other. While these simulation circuits are among the most important for applications in chemistry, it does not necessarily represent all cases of physical interest let alone chemistry. Here we address this by discussing ways to synthesize circuits for arbitrary exponentiated Hamiltonians in  $\mathbb{C}^{2^n \times 2^n}$ , expressible as sum of Pauli operators, with an aim to reduce the number of non-Clifford resources. For reasons discussed previously, it is enough to consider a Hamiltonian  $H$  expressed as sum of commuting Pauli operators.

$$H = \sum_i \alpha_i P_i \quad P_i \in \mathcal{P}_n$$

In most cases one synthesizes circuit for each  $e^{-i\alpha_i P_i t}$  using a number of CNOT and one  $R_z$  gate. Thus the number of  $R_z$  gates required is equal to the number of summands. Here we describe procedure to synthesize circuit for  $e^{-iHt}$  i.e. considering multiple summands or Pauli operators.

We diagonalize  $H$ , for example, by using the algorithms in<sup>51</sup>. In the previous section we have constructed explicit eigenbases for the diagonalization of some specific Hamiltonians. Then we get the following.

$$H = W \left( \sum_i \alpha'_i Q_i \right) W^\dagger$$

Here  $Q_i = \otimes_{j=1}^n Q_{ij}$ , a tensor product of  $Z$  and  $\mathbb{I}$  i.e.  $Q_{ij} \in \{Z, \mathbb{I}\}$ .  $W$  is a diagonalizing Clifford circuit. Thus we get the following.

$$e^{-iHt} = W e^{\sum_i \alpha'_i Q_i} W^\dagger \quad (34)$$

**Lemma 2.3.** Let  $\mathcal{H} = \sum_i \alpha'_i Q_i$ , such that  $Q_i = \otimes_{j=1}^n Q_{ij}$ , where  $Q_{ij} \in \{Z, \mathbb{I}\}$ . With each  $Q_i$  we associate an  $n$ -length vector  $\mathbf{y}_i = (y_{i1}, \dots, y_{in}) \in \{0, 1\}^n$  such that  $(\mathbf{y}_i)_j = y_{ij} = 1$  if  $Q_{ij} = Z$ , else  $y_{ij} = 0$ . Let  $x_1, \dots, x_n \in \{0, 1\}$  and  $|0\rangle = [1, 0]^T$ ,  $|1\rangle = [0, 1]^T$ . The eigenvectors of  $\mathcal{H}$  are of the form  $|v\rangle = \otimes_{j=1}^n |x_j\rangle$  and the corresponding eigenvalue is

$$\phi_v = \sum_i \alpha'_i (-1)^{\oplus_{j=1}^n y_{ij} x_j} \quad (35)$$

*Proof.* The summands in  $\mathcal{H}$  are mutually commuting and so they have a common eigenbasis. Let us first consider  $Q_i$ . Since  $Q_{ij}|x\rangle = |x\rangle$  if  $Q_{ij} = \mathbb{I}$  and  $Q_{ij}|x\rangle = (-1)^{x_j}|x\rangle$  if  $Q_{ij} = Z$ , where  $x \in \{0, 1\}$ , so we have the following.

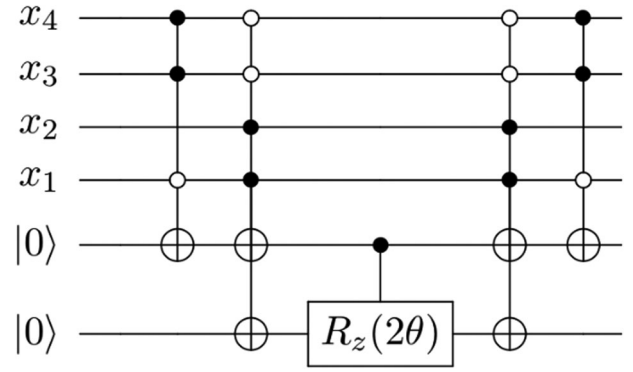
$$Q_i |v\rangle = \left( \bigotimes_{j=1}^n Q_{ij} \right) \left( \bigotimes_{j=1}^n |x_j\rangle \right) = \bigotimes_{j=1}^n Q_{ij} |x_j\rangle = (-1)^{\oplus_{j=1}^n y_{ij} x_j}$$

This implies that

$$\mathcal{H} |v\rangle = \left( \sum_i \alpha'_i Q_i \right) |v\rangle = \sum_i \alpha'_i (-1)^{\oplus_{j=1}^n y_{ij} x_j} |v\rangle.$$

We can also interpret  $\phi$  as the overall phase incurred between  $W$  and  $W^\dagger$ . For given values of  $\alpha'_i$ ,  $\phi$  is an  $n$ -variable Boolean function where  $x_j$  are the Boolean variables. We can evaluate a truth table and get all the  $2^n$  values of  $\phi$  for different values of  $(x_1, \dots, x_n) \in \{0, 1\}^n$ . For each distinct non-zero absolute phase value  $|\theta|$  (ignoring sign), we can have a sub-circuit  $C_{|\theta|}$  that has only one controlled rotation  $cR_z(2\theta)$  gate. The complete circuit can be obtained by combining these different sub-circuits (one for each  $|\theta| \neq 0$ ), in between the diagonalizing Clifford circuits  $W, W^\dagger$ . The ordering of the sub-circuits do not matter.

Now we discuss how to synthesize sub-circuit  $C_{|\theta|}$ , for one such distinct absolute value of  $\phi$ . Let  $\mathcal{M}_\theta$  be the set of binary values for



**Fig. 5** Circuit  $C_{|\theta|}$ . The circuit when  $\mathcal{M}_\theta = \{(1, 1, 0, 0), (1, 1, 1, 0)\}$  and  $\mathcal{M}_{-\theta} = \{(0, 0, 1, 1)\}$ .

variables  $x_1, x_2, \dots, x_n$ , such that  $\phi$  computes to  $\theta$  in Equation (35).

$$\mathcal{M}_\theta = \{(x_1, \dots, x_n) \in \{0, 1\}^n : \phi_{x_1, \dots, x_n} = \theta\} \quad (36)$$

Analogously we can define  $\mathcal{M}_{-\theta}$ . We can also associate  $\mathcal{M}_\theta$  and  $\mathcal{M}_{-\theta}$  with  $S_\theta$  and  $S_{-\theta}$ , the sets of eigenvectors with eigenvalues  $\theta$  and  $-\theta$  respectively, as obtained from Lemma 2.3. We define the following operators, which acts on the input vector space and the space of two ancillae -  $c$  and  $r$ , the latter being initialized to 0.

$$\begin{aligned} V_\theta &= \sum_{|v\rangle \in S_\theta} |v, c \oplus 1, 0\rangle \langle v, c, 0| + \sum_{|w\rangle \notin S_\theta} |w, c, 0\rangle \langle w, c, 0| \\ V_{-\theta} &= \sum_{|v\rangle \in S_{-\theta}} |v, c \oplus 1, 1\rangle \langle v, c, 0| + \sum_{|w\rangle \notin S_{-\theta}} |w, c, 0\rangle \langle w, c, 0| \end{aligned} \quad (37)$$

The circuit  $C_{|\theta|} = V_\theta V_{-\theta} (cR_z(2\theta))_r V_{-\theta}^\dagger V_\theta^\dagger$ . If the input vector is in  $S_\theta$  or  $S_{-\theta}$  then both these operators flip a control ancilla qubit ( $c$ ). Additionally, if the vector is in  $S_{-\theta}$  then the second ancilla  $r$  is flipped. We apply a  $cR_z(2\theta)$  gate on  $r$ , controlled on  $c$ . Thus if the input vector is in  $S_{-\theta}$  then we actually apply  $cR_z(-2\theta)$ . The ancillae  $c, r$  can be controlled by multi-controlled X gates, that can be further decomposed in terms of Toffoli and CNOT gates<sup>69,70</sup>. For example, let  $\mathcal{M}_\theta = \{(0, 0, 1, 1), (0, 1, 1, 1)\}$  and  $\mathcal{M}_{-\theta} = \{(1, 1, 0, 0)\}$ . The two Boolean min-terms of  $\mathcal{M}_\theta$  can be compressed to have a single term because when  $x_1 = 0, x_3 = x_4 = 1$  then  $\phi = \theta$ , irrespective of the value of  $x_2$ . We call it the 'don't care condition' for  $x_2$ . So, equivalently we can write  $\mathcal{M}_\theta = \{(0, *, 1, 1)\}$ , where  $*$  denotes the don't-care condition. In general, algorithms like Karnaugh map<sup>71</sup>, ESPRESSO<sup>72</sup> can be used to get compact set of Boolean min-terms. A circuit  $C_{|\theta|}$  has been shown in Fig. 5.

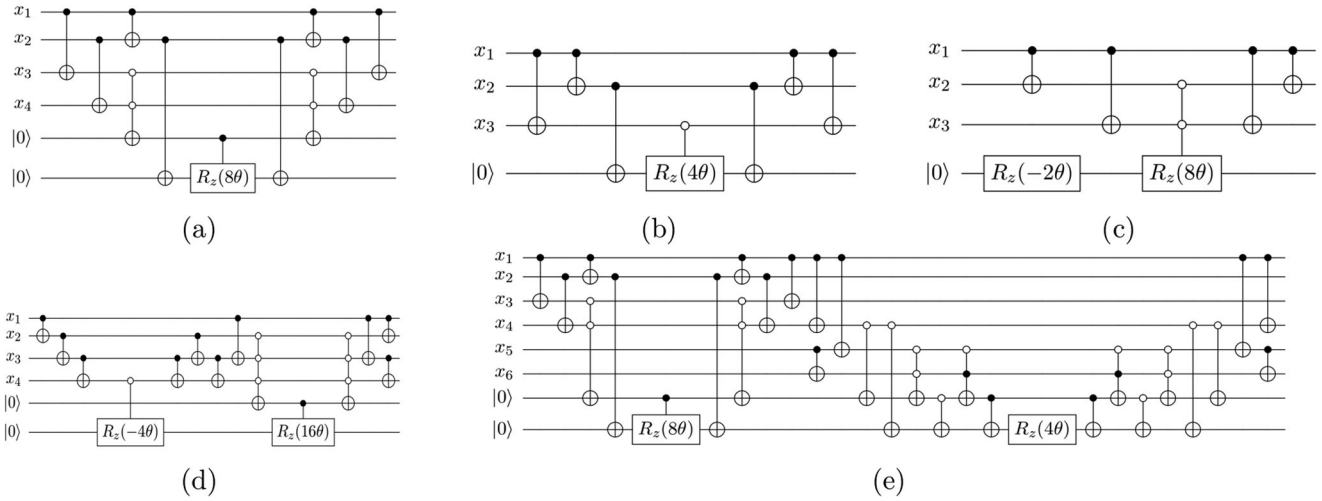
Hence, due to the invariance of the point spectrum of unitarily equivalent operators we have the following.

**Lemma 2.4.** Let  $H = \sum_i \alpha_i P_i$ , where  $P_i$  are mutually commuting  $n$ -qubit Pauli operators. We can implement a circuit for  $e^{-iHt}$  with at most  $m$  (controlled)-rotations, where  $m$  is the number of distinct non-zero eigenvalues (ignoring sign) of  $H$ .

*Illustration—Quantum Heisenberg and quantum Ising model.* We consider the problem of designing quantum circuits for simulating the quantum Heisenberg and Ising model with Hamiltonians  $H_H$  and  $H_I$  respectively. The Heisenberg Hamiltonian is widely used to study magnetic systems, where the magnetic spins are treated quantum mechanically<sup>60,73–76</sup>. Let  $G = (E, V)$  be the underlying graph with the vertex and edge set being  $V$  and  $E$ , respectively.

$$H_H = \sum_{(i,j) \in E} (J_x X_{(i)} X_{(j)} + J_y Y_{(i)} Y_{(j)} + J_z Z_{(i)} Z_{(j)}) + \sum_{i \in V} d_h Z_{(i)} \quad (38)$$

$$H_I = \sum_{(i,j) \in E} J_z Z_{(i)} Z_{(j)} + \sum_{i \in V} d'_h Z_{(i)} \quad (39)$$



**Fig. 6** Quantum circuit simulating  $e^{i\theta \sum_{(i,j) \in E} Z_{(i)} Z_{(j)}}$ . When the underlying graph  $G = (V, E)$  is (a) 4-point circle, (b) 3-point line, (c) triangle, (d) 4-point complete graph, (e) 6-point circle.

In the above  $J_x, J_y, J_z$  are coupling parameters, denoting the exchange interaction between nearest neighbor spins along the X,Y,Z-direction respectively.  $d_h, d'_h$  is the time amplitude of the external magnetic field along the Z-direction. One set of commuting Paulis are  $\{X_{(i)}X_{(j)}: (i, j) \in E\}$ ,  $\{Y_{(i)}Y_{(j)}: (i, j) \in E\}$  and  $\{Z_{(i)}Z_{(j)}: (i, j) \in E\}$  and  $\{Z_{(i)}: i \in V\}$ .

Let us first consider the set  $\{Z_{(i)}: i \in V\}$ . Following the previous discussions and Lemma 2.3, the overall phase incurred or the eigenvalues are as follows.

$$\phi' = d_h \sum_{i \in V} (-1)^{x_i} \quad x_i \in \{0, 1\} \quad (40)$$

For  $\mathbf{x} \in \{0, 1\}^{|V|}$ , one particular assignment of values to the Boolean variables, let  $T_0 = \{i \in V: x_i = 0\}$  and  $T_1 = \{i \in V: x_i = 1\}$ . So  $|T_0| + |T_1| = |V|$  and

$$\phi'_\mathbf{x} = d_h (|T_0| - |T_1|) = d_h (|V| - 2|T_1|). \quad (41)$$

So the number of distinct non-zero eigenvalues or absolute values of  $\phi'$  can be  $\lceil |V|/2 \rceil$ . Implementing each  $e^{-id_h Z_{(i)} t}$  would require  $|V|$  rotation gates. Thus, using Lemma 2.4, we have about 50% reduction in the rotation gate cost.

Now, let us consider the other commuting sets. Since  $H^\dagger X H = Z$  and  $(HSX)^\dagger Y (HSX) = Z$ , so each of the above sets can be diagonalized and we can focus on the problem of simulating a quantum circuit for the Hamiltonian:  $H = J \sum_{(i,j) \in E} Z_{(i)} Z_{(j)}$ , where  $J$  is a constant. Our aim is to derive an upper bound on the number of controlled rotations required to simulate  $e^{-iHt}$ . Following the previous discussions, the overall phase incurred between the diagonalizing Cliffords  $W, W^\dagger$  is as follows (Lemma 2.3).

$$\phi = J \sum_{(i,j) \in E} (-1)^{x_i \oplus x_j} \quad (42)$$

where  $x_i, x_j \in \{0, 1\}$  are variables denoting the state of the qubits after application of  $W$ . The quantum circuit has  $|V|$  qubits, corresponding to each vertex of  $G$ . Let  $\mathbf{x} = \{0, 1\}^{|V|}$  denote one particular assignment of values to the variables  $x_1, \dots, x_{|V|}$ .  $S_0 = \{(i, j) \in E: x_i = x_j = 1\}$  and  $S_1 = \{(i, j) \in E: x_i \neq x_j\}$ . If  $S' = \{(i, j) \in E: x_i = x_j = 0\}$ , then  $|S'| = |E| - |S_0| - |S_1|$ . Let  $\phi_\mathbf{x}$  be the value of the phase for this particular assignment.

$$\phi_\mathbf{x} = J(|S'| + |S_0| - |S_1|) = J(|E| - |S_0| - |S_1| + |S_0| - |S_1|) = J(|E| - 2|S_1|) \quad (43)$$

Let  $V_{1\mathbf{x}} = \{i \in V: x_i = 1\}$ ,  $V_{0\mathbf{x}} = \{i \in V: x_i = 0\}$  and  $\mathcal{N}(k)$  be the set of neighbouring vertices of  $k$  in  $G$ . Then

$$|S_1| = \sum_{k \in V_{1\mathbf{x}}} |\{\mathcal{N}(k) \setminus V_{1\mathbf{x}}\}| \quad (44)$$

Now for any assignment  $|S_1|$  can vary from  $1, \dots, |E|$ . So the number of distinct values of  $\phi$  is at most  $\lceil |E|/2 \rceil$ . And hence we need at most  $\lceil |E|/2 \rceil$  controlled- $R_z$  gates in the circuit simulating  $e^{-iHt}$ . Had we simulated each  $e^{-iJZ_{(i)}Z_{(j)}t}$ , we would have required  $|E|R_z$  gates. So we can achieve about 50% reduction in the rotation gate cost under the assumption that controlled- $R_z$  costs the same to implement as a single  $R_z$  gate.

*G is a cycle.* This is basically the translationally invariant 1-D spin chain. Let  $G_{V_{1\mathbf{x}}}$  be the subgraph induced by  $V_{1\mathbf{x}}$ , which is a union of paths. For each path  $p$ , let  $S_{1p} = \bigcup_{k \in p} \{\mathcal{N}(k) \setminus V_{1\mathbf{x}}\} \subseteq S_1$  be the set of vertices in this path. Each of the terminal vertices has one neighbour in  $V \setminus V_{1\mathbf{x}}$ . So  $|S_{1p}| = 2$ . Thus if  $\mathcal{P}$  is the set of all such paths in  $G_{V_{1\mathbf{x}}}$ , then

$$\phi_\mathbf{x} = J \left( |E| - 2 \sum_{p \in \mathcal{P}} |S_{1p}| \right) = J(|E| - 4|\mathcal{P}|) \quad (45)$$

Now  $|\mathcal{P}|$  can vary from  $1, \dots, \lceil |E|/4 \rceil$  to give  $\lceil |E|/4 \rceil$  distinct values of  $|\phi|$ . This implies a quantum circuit synthesizing  $e^{-iHt}$  will require at most  $\lceil |E|/4 \rceil cR_z$  gates. This is about 75% reduction in the cost of rotation gates, compared to synthesizing each  $e^{-iJZ_{(i)}Z_{(j)}t}$ .

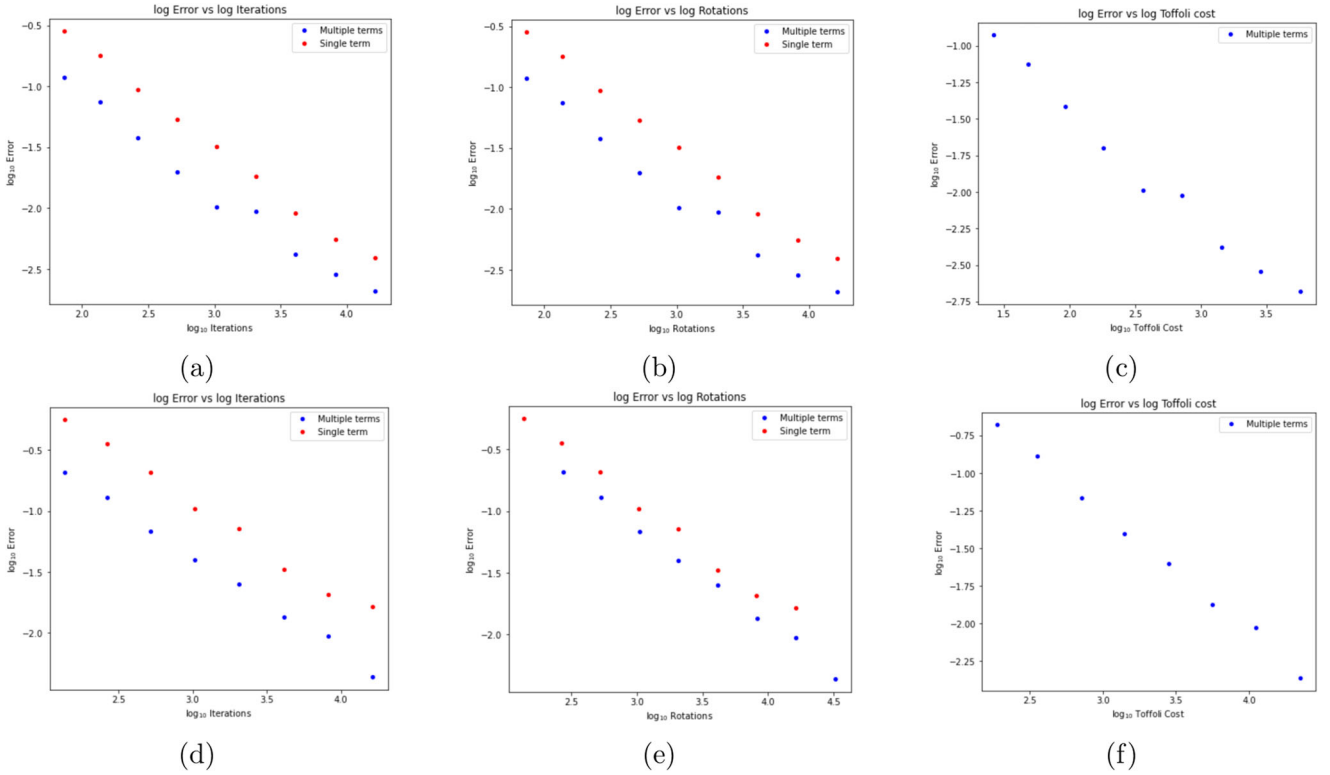
*G is a complete graph.* In this case for each  $k \in V_{1\mathbf{x}}$ , we have  $\mathcal{N}(k) \setminus V_{1\mathbf{x}} = V_{0\mathbf{x}}$ . So we have,

$$\phi_\mathbf{x} = J(|E| - 2|V_{1\mathbf{x}}||V_{0\mathbf{x}}|) = J(|E| - 2\|\mathbf{x}\|_1 (|V| - \|\mathbf{x}\|_1)) \quad (46)$$

So there can be  $\lceil |V|/2 \rceil$  distinct values of  $|\phi_\mathbf{x}|$  as  $\|\mathbf{x}\|_1$  varies from  $1, \dots, \lceil |V|/2 \rceil$ . And hence we require at most  $\lceil |V|/2 \rceil cR_z$  gates for simulating  $e^{-iHt}$ . If we simulate each  $e^{-iJZ_{(i)}Z_{(j)}t}$  then we require  $|E| = \frac{|V|(|V|-1)}{2} R_z$  gates. This indicates about  $100 \left(1 - \frac{2}{|V|-1}\right)\%$  reduction in the cost of rotation gates.

In Fig. 6 we have shown quantum circuits simulating  $e^{i\theta \sum_{(i,j) \in E} Z_{(i)} Z_{(j)}}$  for some simple graphs  $G = (V, E)$ . The circuits have been designed to optimize the number of Toffoli gates, as well.

*Reducing the number of Toffoli gates.* We discussed before that the T-count from the Toffolis may be a significant factor in high



**Fig. 7 Simulation of 4 and 6-qubit Heisenberg Hamiltonian.** Log-log plots showing number of iterations (**a, d**),  $R_z/cR_z$  (**b, e**), Toffoli-pairs (**c, f**) as function of error, while simulating the 4 (**a–c**) and 6-qubit (**d–f**) quantum Heisenberg Hamiltonian ( $H_H$ ) with qDRIFT. The red and blue curve shows the variation when sampling single and multiple commuting Paulis per iteration, respectively. The Y-axis label in all plots is  $\log_{10}$  Error. The X-axis label of (**a**), (**d**) is  $\log_{10}$  Iterations, (**b**), (**e**) is  $\log_{10}$  Rotations and (**c**), (**f**) is  $\log_{10}$  Toffoli Cost.

error regime as the logarithmic cost of rotation synthesis may not dominate the additive constant that arises from the Toffoli gates needed. In order to reduce the number of Toffolis we can do the following. We design circuits reducing Toffolis for Hamiltonians over smaller graphs, such as in Fig. 6a–e. Then we decompose a Hamiltonian over a large graph into Hamiltonians over these smaller graphs. For example, consider a 1-D cycle on  $N$  points and  $H_z = \theta \sum_{(i,j) \in E} Z_{(i)} Z_{(j)}$ . We break this cycle into smaller chains of length 3 i.e.  $H_z = \theta (Z_{(1)} Z_{(2)} + Z_{(2)} Z_{(3)} + Z_{(3)} Z_{(4)} + Z_{(4)} Z_{(5)} + \dots = \sum_i H_{z_i}$ . We have a quantum circuit that synthesizes each  $e^{iH_{z_i} t}$  with only one  $cR_z$  gate (Fig. 6b). So to synthesize  $e^{iH_z t}$  we require approximately  $N/2cR_z$ . This is about twice the number of controlled rotations required, had we synthesized without decomposing. But it does not require any extra Toffoli-pairs. We manage to get approximately 50% reduction, compared to synthesizing each summand i.e.  $e^{i\theta Z_{(i)} Z_{(j)}}$ .

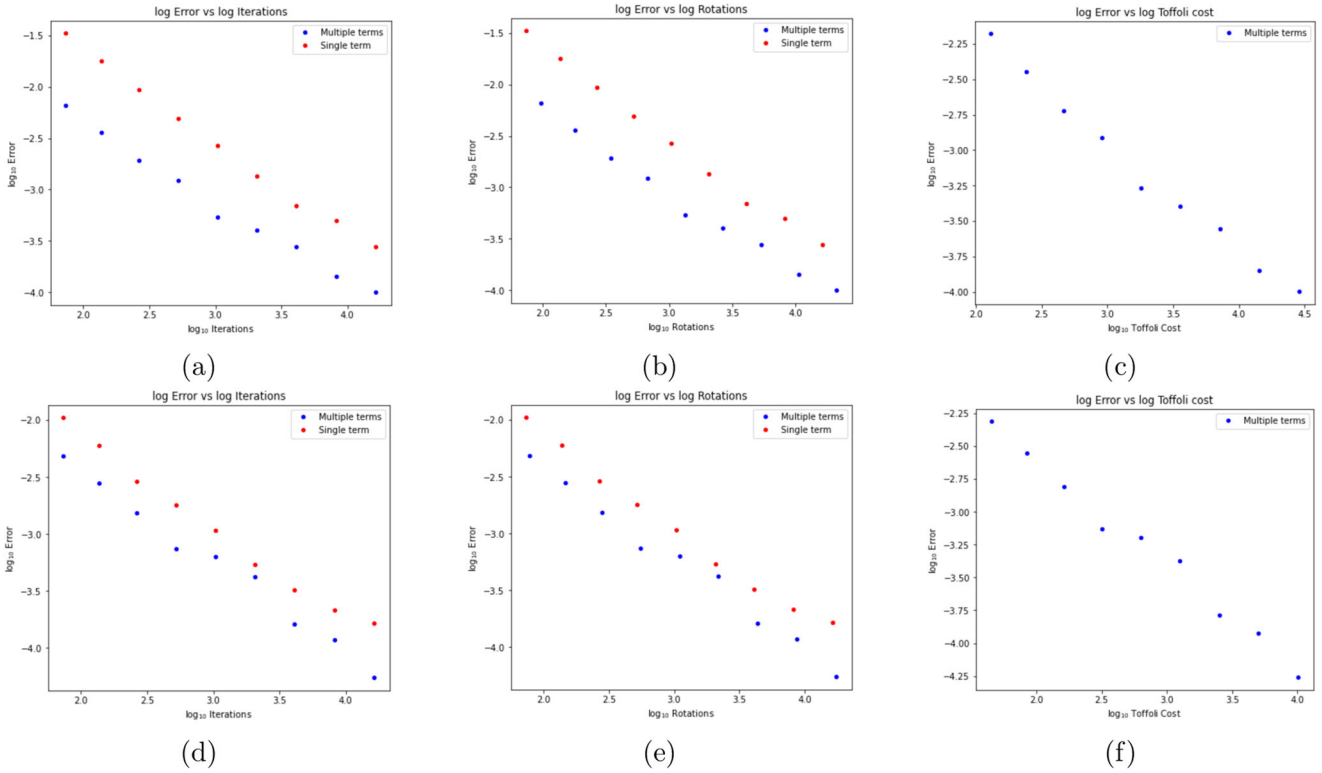
Now consider a large  $N \times N$  lattice which has  $N^2$  vertices and  $2N(N-1)$  edges and the Hamiltonian  $H_z = \theta \sum_{(i,j) \in E} Z_{(i)} Z_{(j)}$ . We can decompose this into  $(N-1)^2$  smaller interior cycles of 4-points and a bigger outer circle with  $2N + 2(N-2) = 4(N-1)$  points. From Fig. 6a, we know that we can design a circuit simulating the exponentiated Hamiltonian corresponding to each interior cycle with 1  $cR_z$  and 1 Toffoli pair. We can further decompose the outer circle (as explained in the previous paragraph) and have a circuit with approximately  $2(N-1)cR_z$  gates. Thus we require  $\approx (N-1)^2 + 2(N-1) = (N-1)(N+1)cR_z$  and  $(N-1)^2$  Toffoli-pairs. We have discussed before that for general graphs, number of  $cR_z$  required is  $\approx |E|/2 = N(N-1)$ , so we use  $\approx (N-1)$  more  $cR_z$  by decomposing, but the Toffoli cost reduces a lot. Had we synthesized each  $e^{i\theta Z_{(i)} Z_{(j)}}$ , we would have used  $2N(N-1)$   $R_z$ . Thus we manage to get a reduction of  $\approx (N-1)^2$  in the number of  $R_z/cR_z$ .

In Fig. 6e we gave a circuit for simulating  $e^{i\theta \sum_{(i,j) \in E} Z_{(i)} Z_{(j)}}$ , when the underlying graph is a 6-point cycle. We reduced the Toffoli-pairs by decomposing the graph into smaller cycles.

### Application : Simulating with qDRIFT

In this section we consider one simulation algorithm - qDRIFT. We focus on qDRIFT rather than Trotter for our experiments because qDRIFT is easier to analyze numerically. This is because Trotter errors subtly depend on operator ordering. Specifically we consider a Hamiltonian  $H = \sum_{j=1}^L h_j P_j$  and sample Pauli operators to apply in each short time step, as described earlier in the paper and in<sup>43</sup>. We can assume each  $h_j > 0$ , since the negation affects the angles of the rotation gates. In qDRIFT, in each iteration one Pauli term is sampled up to a total of  $N$  samples. The probability of sampling  $P_j$  is  $h_j / \sum_i h_i$  and is then simulated for a short time period. We consider another procedure where we re-write the Hamiltonian as  $H = \sum_{j=1}^L h'_j H_j$ , where each  $H_j = \sum_i P_{i,j}$  is a sum of commuting Paulis. In each iteration one  $H_j$  is sampled with probability  $\frac{h'_j}{\sum_i h'_i}$  and simulated for a short time period. Then we compare the growth of error, number of  $R_z/cR_z$ , Toffoli gates used in these two procedures - (i) one Pauli sampled in each iteration, (ii) group of commuting Paulis sampled in each iteration.

For our first set of experiments we examine the case of simulation of 4 and 6-qubit Heisenberg models. The coefficients  $J_x, J_y, J_z, d_h$  have been sampled from a 0 mean normal distribution with variance 1. In Fig. 7 we show that we achieve better scaling of  $R_z/cR_z$  when multiple commuting Pauli operators are sampled and evolved in each iteration. In fact, the error also scales well with the number of iterations, i.e. we can achieve the same error in less



**Fig. 8 Simulation of  $H_2$  and LiH Hamiltonian.** Log-log plots showing number of iterations (**a, d**),  $R_z/cR_z$  (**b, e**), Toffoli-pairs (**c, f**) as function of error, while simulating the  $H_2$  (**a–c**) and LiH (**d–f**) Hamiltonians with qDRIFT. The red and blue dots show the variation when sampling single and multiple commuting Paulis per qDRIFT iteration, respectively. The Y-axis label in all plots is  $\log_{10}$  Error. The X-axis label of (**a**), (**d**) is  $\log_{10}$  Iterations, (**b**), (**e**) is  $\log_{10}$  Rotations and (**c**), (**f**) is  $\log_{10}$  Toffoli Cost.

number of iterations, or in another way, it is possible to achieve much lower error in the same time (iterations) when multiple operators are sampled. We calculate error as:

$$\text{Error} = \mathbb{E}_\rho(\|\mathcal{E}_2(\rho) - \mathcal{E}_1(\rho)\|_{l_2})$$

Where  $\mathcal{E}_1 = e^{iHt}\rho e^{-iHt}$ ,  $\tau = t \cdot (\sum_j h_j)/N$  and  $\|\cdot\|_{l_2}$  is the induced Euclidean norm on matrices and  $\mathbb{E}_\rho$  is the Haar average over input states. We obtain  $\mathcal{E}_2$  through averaging  $M$  random qDRIFT protocols, where  $M$  varies from 100 to 3000 for our purposes. These values are chosen to ensure that the sampling error is small at the scale of the plots generated.

$$V_k = \prod_{j_k} e^{iH_{j_k} \tau}$$

$$\mathcal{E}_2 = \frac{1}{M} \sum_{k=1}^M V_k \rho V_k^\dagger$$

In our experiments  $\rho$  is randomly drawn rather than chosen to maximize the diamond distance. As a result, this does not give a tight upper bound on the error quantified by any induced channel norm. Further, all evolution is done using  $t = 1$  and the groupings are hand optimized using counts given in Supplementary Method 5. The data, tabulated in Fig. 7, shows that the number of iterations of the qDRIFT channel needed to simulate the dynamics to bound the error below a particular value, is reduced by a factor of 2.34 and 2.8 through the use of grouping commuting terms for the randomly chosen 4 and 6 qubit Heisenberg Hamiltonian respectively. The number of rotations is found to be reduced by a factor of roughly 2.34 for the 4 qubit ensemble but 1.8 for the 6 qubit case. This suggests that the groupings that we consider, while highly successful at reducing the number of iterations of qDRIFT needed, the number of gates

per iteration increases from the 4 to 6 qubit examples. This suggests that further computer aided optimization may be needed in order to see the full benefit of such groupings as we increase the size of models.

Similar observations can be made for our second set of experiments where we simulate the Hamiltonian of  $H_2$  and LiH (with freezing in the STO-3G basis). The plots in Fig. 8 show that in case of  $H_2$ , the number of iterations of the qDRIFT channel needed to simulate the dynamics to bound the error below a particular value, is reduced by a factor of 4 through the use of grouping commuting terms. For LiH this factor is nearly 2.1. The number of rotations is found to be reduced by a factor of roughly 3.2 for  $H_2$  and 2 for LiH.

For all the experiments that we consider the Toffoli-pair gate count is comparable with the  $R_z/cR_z$  count, so the Toffoli pairs do not contribute significantly to the overall T-count, as compared to the rotation gates. The number of gates depend on the diagonalizing circuits and the grouping into commuting Paulis. In this paper we have shown the set of results for the eigenbasis or grouping that were better among the options considered by us. In Supplementary Method 5 we have explicitly mentioned the Hamiltonians, the groupings and given a short description of how we obtained the rotation and Toffoli counts.

All plots, code, and data can be found online in our public repository <https://github.com/SNIPRS/hamiltonian>. All code was written in Python. Our results were obtained partly with computing resources in the Cedar cluster of Compute Canada. Specifically, our code was run on an Intel(R) Xeon(R) E5-2683 v4 CPU at 2.10 GHz, utilizing 48 cores, up to 12GBs of RAM, and running Gentoo Linux 2.6. For the Heisenberg Hamiltonians, our results were obtained using 12 cores of an Intel(R) Core(TM) i5-12600K CPU at 3.6 GHz running Ubuntu 20.04.4 and up to 32GBs of RAM.

## DISCUSSION

In this paper, we have considered the problem of designing efficient quantum circuits for exponential of Hamiltonians that can be expressed as sum of Paulis. In contrast with most previous approaches, we synthesize circuit for a sum of exponentiated commuting Paulis, rather than concatenate circuits for each exponentiated Pauli. These resulting circuits are observed, for some parameter combinations, to require far fewer non-Clifford operations than the standard circuits. We therefore propose an algorithm for greedily compiling a Trotter or qDRIFT simulation into a sequence of such simulations and observe that when multiple rotations are grouped we see at fixed error that a factor of roughly 1.8–3.2 fewer rotations are needed to simulate 6 and 4-qubit Heisenberg models,  $LiH$ ,  $H_2$ . Also, for simulation protocols like qDRIFT, it is possible to achieve a better performance, in the sense that the error accumulated per iteration is less if we sample multiple commuting Paulis. The overall non-Clifford gate cost of the entire protocol is also less.

There are a number of interesting avenues that are revealed by this work. The first is that a more complete set of rules for compiling Hamiltonian terms into sets that can be easily exponentiated reveals the potential for more efficient simulation compilation of Hamiltonians. These replacement rules, once identified, can be used inside a more systematic Hamiltonian compiler package that would allow more substantial optimizations of the Hamiltonian for the given simulation method. This raises a second issue, while in this work we focus on the case of optimizing Trotter and related simulation methods, similar considerations could be performed for optimizing the prepare and select circuits used in LCU/qubitization simulation algorithms. Such procedures are harder to optimize as the simulation algorithm does not factorize as nicely into independent simulations; however, the importance of these simulation methods makes the development of compilation strategies essential.

Finally, an important avenue hinted at by this work is the possibility that approximate unitary synthesis methods can be combined with quantum simulation routines to further reduce the cost. If fermionic swaps are used, for example, simulation reduces to implementing a series of 4-local Hamiltonians and optimal circuits can be in principle constructed for such Hamiltonians using existing approaches. The computational overheads required for optimal (approximate) synthesis of these unitaries makes this a daunting task; however, if a sufficient lexicon of cheap unitaries are found for such simulation then it will not only lead to lower costs for quantum simulation using Trotter/qDRIFT: it will also unify Hamiltonian compilation with circuit synthesis into a single conceptual framework.

## DATA AVAILABILITY

All plots and data can be found online in our public repository <https://github.com/SNIPRS/hamiltonian>.

## CODE AVAILABILITY

All code can be found online in our public repository <https://github.com/SNIPRS/hamiltonian>.

Received: 8 September 2022; Accepted: 10 March 2023;

Published online: 03 April 2023

## REFERENCES

- Feynman, R. P. et al. Simulating physics with computers. *Int. J. Theor. Phys.* **21**, 467–488 (1982).
- Lloyd, S. Universal quantum simulators. *Science* **273**, 1073–1078 (1996).
- Childs, A. M., Maslov, D., Nam, Y., Ross, N. J. & Su, Y. Toward the first quantum simulation with quantum speedup. *Proc. Natl. Acad. Sci.* **115**, 9456–9461 (2018).
- Suzuki, M. General theory of fractal path integrals with applications to many-body theories and statistical physics. *J. Math. Phys.* **32**, 400–407 (1991).
- Trotter, H. F. On the product of semi-groups of operators. *Proc. Am. Math. Soc.* **10**, 545–551 (1959).
- Berry, D. W. & Childs, A. M. Black-box Hamiltonian simulation and unitary implementation. *Quantum Inf. Comput.* **12**, 29–62 (2012).
- Childs, A. M. & Wiebe, N. Hamiltonian simulation using linear combinations of unitary operations. *Quantum Inf. Comput.* **12**, 901–924 (2012).
- Berry, D. W., Childs, A. M., Cleve, R., Kothari, R. & Somma, R. D. Simulating Hamiltonian dynamics with a truncated Taylor series. *Phys. Rev. Lett.* **114**, 090502 (2015).
- Low, G. H. & Chuang, I. L. Optimal Hamiltonian simulation by quantum signal processing. *Phys. Rev. Lett.* **118**, 010501 (2017).
- Babbush, R. et al. Low-depth quantum simulation of materials. *Phys. Rev. X* **8**, 011044 (2018).
- Cao, Y. et al. Quantum chemistry in the age of quantum computing. *Chem. Rev.* **119**, 10856–10915 (2019).
- Jordan, S. P., Lee, K. S. M. & Preskill, J. Quantum algorithms for quantum field theories. *Science* **336**, 1130–1133 (2012).
- Lanyon, B. P. et al. Towards quantum chemistry on a quantum computer. *Nat. Chem.* **2**, 106–111 (2010).
- McArdle, S., Endo, S., Aspuru-Guzik, A., Benjamin, S. C. & Yuan, X. Quantum computational chemistry. *Rev. Mod. Phys.* **92**, 015003 (2020).
- Poulin, D. et al. The Trotter step size required for accurate quantum simulation of quantum chemistry. *Quantum Inf. Comput.* **15**, 361–384 (2015).
- Wecker, D., Bauer, B., Clark, B. K., Hastings, M. B. & Troyer, M. Gate-count estimates for performing quantum chemistry on small quantum computers. *Phys. Rev. A* **90**, 022305 (2014).
- Keever, C. M. & Lubasch, M. Classically optimized Hamiltonian simulation. Preprint at <https://arxiv.org/quant-ph/2205.11427> (2022).
- Abrams, D. S. & Lloyd, S. Quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors. *Phys. Rev. Lett.* **83**, 5162 (1999).
- Peruzzo, A. et al. A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **5**, 1–7 (2014).
- Wang, D., Higgott, O. & Brierley, S. A generalised variational quantum eigensolver. Preprint at <https://arxiv.org/quant-ph/1802.00171> (2018).
- O'Brien, T. E., Tarasinski, B. & Terhal, B. M. Quantum phase estimation for noisy, small-scale experiments. Preprint at <https://arxiv.org/quant-ph/1809.09697> (2018).
- Berry, D. W. High-order quantum algorithm for solving linear differential equations. *J. Phys. A Math. Theor.* **47**, 105301 (2014).
- Brandao, F. G. S. L. & Svore, K. M. Quantum speed-ups for solving semidefinite programs. In: *Proc. 58th Ann. Symp. on Foundations of Computer Science* 415–426 (IEEE, 2017).
- Childs, A. M. et al. Exponential algorithmic speedup by a quantum walk. In: *Proc. 35th Ann. Symp. on Theory of Computing* 59–68 (ACM, 2003).
- Farhi, E., Goldstone, J. & Gutmann, S. A quantum algorithm for the Hamiltonian NAND tree. *Theory Comput.* **4**, 169–190 (2008).
- Harrow, A. W., Hassidim, A. & Lloyd, S. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.* **103**, 150502 (2009).
- Mosca, M. & Mukhopadhyay, P. A polynomial time and space heuristic algorithm for T-count. *Quantum Sci. Technol.* **7**, 015003 (2021).
- Gheorghiu, V., Mosca, M. & Mukhopadhyay, P. A (quasi-)polynomial time heuristic algorithm for synthesizing T-depth optimal circuits. *NPJ Quantum Inf.* **8**, 110 (2022).
- Gheorghiu, V., Mosca, M. & Mukhopadhyay, P. T-count and T-depth of any multi-qubit unitary. *NPJ Quantum Inf.* **8**, 141 (2022).
- Amy, M., Maslov, D. & Mosca, M. Polynomial-time T-depth optimization of Clifford +T circuits via matroid partitioning. *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.* **33**, 1476–1489 (2014).
- Duncan, R., Kissinger, A., Perdrix, S. & Van De Wetering, J. Graph-theoretic simplification of quantum circuits with the ZX-calculus. *Quantum* **4**, 279 (2020).
- Häner, T. & Soeken, M. Lowering the T-depth of quantum circuits by reducing the multiplicative depth of logic networks. Preprint at <https://arxiv.org/quant-ph/2006.03845> (2020).
- Patel, K. N., Markov, I. L. & Hayes, J. P. Optimal synthesis of linear reversible circuits. *Quantum Inf. Comput.* **8**, 282–294 (2008).
- Amy, M., Azimzadeh, P. & Mosca, M. On the controlled-NOT complexity of controlled-NOT-phase circuits. *Quantum Sci. Technol.* **4**, 015002 (2018).
- Gheorghiu, V., Jiayin, H., Li, S. M., Mosca, M. & Mukhopadhyay, P. Reducing the CNOT count for Clifford+T circuits on NISQ architectures. *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.* (2022).
- Fowler, A. G., Stephens, A. M. & Groszkowski, P. High-threshold universal quantum computation on the surface code. *Phys. Rev. A* **80**, 052312 (2009).
- Aliferis, P., Gottesman, D. & Preskill, J. Quantum accuracy threshold for concatenated distance-3 codes. *Quantum Inf. Comput.* **6**, 97–165 (2006).



38. Fowler, A. G., Mariantoni, M., Martinis, J. M. & Cleland, A. N. Surface codes: Towards practical large-scale quantum computation. *Phys. Rev. A* **86**, 032324 (2012).
39. Bravyi, S. & Gosset, D. Improved classical simulation of quantum circuits dominated by Clifford gates. *Phys. Rev. Lett.* **116**, 250501 (2016).
40. Bravyi, S., Smith, G. & Smolin, J. A. Trading classical and quantum computational resources. *Phys. Rev. X* **6**, 021043 (2016).
41. Paetznick, A. & Reichardt, B. W. Universal fault-tolerant quantum computation with only transversal gates and error correction. *Phys. Rev. Lett.* **111**, 090505 (2013).
42. Kitaev, A. Y. Fault-tolerant quantum computation by anyons. *Ann. Phys.* **303**, 2–30 (2003).
43. Campbell, E. Random compiler for fast Hamiltonian simulation. *Phys. Rev. Lett.* **123**, 070503 (2019).
44. Huggins, W. J. et al. Efficient and noise resilient measurements for quantum chemistry on near-term quantum computers. *NPJ Quantum Inf.* **7**, 1–9 (2021).
45. Jordan, P. & Wigner, E. P. in *The Collected Works of Eugene Paul Wigner* 109–129 (Springer, 1993).
46. Bravyi, S. B. & Kitaev, A. Y. Fermionic quantum computation. *Ann. Phys.* **298**, 210–226 (2002).
47. Jones, C. Low-overhead constructions for the fault-tolerant Toffoli gate. *Phys. Rev. A* **87**, 022328 (2013).
48. Gidney, C. Halving the cost of quantum addition. *Quantum* **2**, 74 (2018).
49. He, Y., Luo, M. X., Zhang, E., Wang, H. K. & Wang, X. F. Decompositions of n-qubit Toffoli gates with linear circuit complexity. *Int. J. Theor. Phys.* **56**, 2350–2361 (2017).
50. Reiher, M., Wiebe, N., Svore, K. M., Wecker, D. & Troyer, M. Elucidating reaction mechanisms on quantum computers. *Proc. Natl. Acad. Sci.* **114**, 7555–7560 (2017).
51. van den Berg, E. & Temme, K. Circuit optimization of Hamiltonian simulation by simultaneous diagonalization of Pauli clusters. *Quantum* **4**, 322 (2020).
52. Kawase, Y. & Fujii, K. Fast classical simulation of Hamiltonian dynamics by simultaneous diagonalization using Clifford transformation with parallel computation. *Comput. Phys. Commun.* (2023).
53. Whitfield, J. D., Biamonte, J. & Aspuru-Guzik, A. Simulation of electronic structure Hamiltonians using quantum computers. *Mol. Phys.* **109**, 735–750 (2011).
54. Barkoutsos, P. K. L. et al. Quantum algorithms for electronic structure calculations: Particle-hole Hamiltonian and optimized wave-function expansions. *Phys. Rev. A* **98**, 022322 (2018).
55. Kivlichan, I. D. et al. Quantum simulation of electronic structure with linear depth and connectivity. *Phys. Rev. Lett.* **120**, 110501 (2018).
56. Ganzhorn, M. et al. Gate-efficient simulation of molecular eigenstates on a quantum computer. *Phys. Rev. Appl.* **11**, 044092 (2019).
57. Gard, B. T. et al. Efficient symmetry-preserving state preparation circuits for the variational quantum eigensolver algorithm. *NPJ Quantum Inf.* **6**, 1–9 (2020).
58. Lepage, H. V., Lasek, A. A., Arvidsson-Shukur, D. R. M. & Barnes, C. H. W. Entanglement generation via power-of-SWAP operations between dynamic electron-spin qubits. *Phys. Rev. A* **101**, 022329 (2020).
59. Yordanov, Y. S., Arvidsson-Shukur, D. R. M. & Barnes, C. H. W. Efficient quantum circuits for quantum computational chemistry. *Phys. Rev. A* **102**, 062612 (2020).
60. Gulania, S., Peng, B., Alexeev, Y. & Govind, N. Quantum time dynamics of 1D-Heisenberg models employing the Yang-Baxter equation for circuit compression. Preprint at <https://arxiv.org/quant-ph/2112.01690> (2021).
61. Childs, A. M., Su, Y., Tran, M. C., Wiebe, N. & Zhu, S. Theory of Trotter error with commutator scaling. *Phys. Rev. X* **11**, 011020 (2021).
62. Nielsen, M. A. & Chuang, I. L. *Quantum Computation and Quantum Information* (Cambridge University Press, 2010).
63. Berry, D. W., Ahokas, G., Cleve, R. & Sanders, B. C. Efficient quantum algorithms for simulating sparse Hamiltonians. *Commun. Math. Phys.* **270**, 359–371 (2007).
64. Szabo, A. & Ostlund, N. S. *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory* (Courier Corporation, 2012).
65. Helgaker, T., Jorgensen, P. & Olsen, J. *Molecular Electronic-structure Theory* (John Wiley & Sons, 2014).
66. Feller, D. The role of databases in support of computational chemistry calculations. *J. Comput. Chem.* **17**, 1571–1586 (1996).
67. Schuchardt, K. L. et al. Basis set exchange: a community database for computational sciences. *J. Chem. Inf. Model* **47**, 1045–1052 (2007).
68. Verstraete, F., Cirac, J. I. & Latorre, J. I. Quantum circuits for strongly correlated quantum systems. *Phys. Rev. A* **79**, 032316 (2009).
69. Barenco, A. et al. Elementary gates for quantum computation. *Phys. Rev. A* **52**, 3457 (1995).
70. da Silva, A. J. & Park, D. K. Linear-depth quantum circuits for multi-qubit controlled gates. *Phys. Rev. A* **106**, 042602 (2022).
71. Karnaugh, M. The map method for synthesis of combinational logic circuits. *Trans. Am. Inst. Elect. Eng. I: Commun. Electron.* **72**, 593–599 (1953).
72. Brayton, R. K., Hachtel, G. D., Hemachandra, L. A., Newton, A. R. & Sangiovanni-Vincentelli, A. L. M. A comparison of logic minimization strategies using espresso: An APL program package for partitioned logic minimization. In: *Proc. Int. Symposium on Circuits Systems* 42–48 (IEEE, 1982).
73. Fazekas, P. *Lecture Notes on Electron Correlation and Magnetism* Vol. 5 (World Scientific, 1999).
74. de PR Moreira, I. & Illas, F. A unified view of the theoretical description of magnetic coupling in molecular chemistry and solid state physics. *Phys. Chem. Chem. Phys.* **8**, 1645–1659 (2006).
75. Skomski, R. *Simple Models of Magnetism* (Oxford University Press, 2008).
76. Pires, A. S. T. & Sergio, A. *Theoretical Tools for Spin Models in Magnetic Systems* (IOP Publishing Bristol, 2021).

## ACKNOWLEDGEMENTS

P.M. wishes to thank NTT Research for their financial and technical support. This work was supported in part by Canada's NSERC. Research at IQC is supported in part by the Government of Canada through Innovation, Science and Economic Development Canada. This research was enabled in part by support provided by WestGrid ([www.westgrid.ca](http://www.westgrid.ca)) and Compute Canada Calcul Canada ([www.computecanada.ca](http://www.computecanada.ca)). N.Wiebe acknowledges funding from the Google Quantum Research Program, the Natural Sciences and Engineering Research Council of Canada and this work on this project was supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Co-Design Center for Quantum Advantage under contract number DE-SC0012704.

## AUTHOR CONTRIBUTIONS

The ideas were given by P.M. and N.W. The implementations were done by H.T.Z. All the authors contributed in the preparation of the manuscript.

## COMPETING INTERESTS

The authors declare no competing interests.

## ADDITIONAL INFORMATION

**Supplementary information** The online version contains supplementary material available at <https://doi.org/10.1038/s41534-023-00697-6>.

**Correspondence** and requests for materials should be addressed to Priyanka Mukhopadhyay.

**Reprints and permission information** is available at <http://www.nature.com/reprints>

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



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023