

Received September 12, 2020; accepted October 24, 2020; date of publication November 4, 2020;
 date of current version December 17, 2020.

Digital Object Identifier 10.1109/TQE.2020.3035814

$O(N^3)$ Measurement Cost for Variational Quantum Eigensolver on Molecular Hamiltonians

PRANAV GOKHALE¹, OLIVIA ANGIULI²,
 YONGSHAN DING³(Graduate Student Member, IEEE), KAIWEN GUI³,
 TEAGUE TOMESH⁴, MARTIN SUCHARA⁵,
 MARGARET MARTONOSI⁴(Fellow, IEEE),
 AND FREDERIC T. CHONG³(Senior Member, IEEE)

¹Super.tech, Chicago, IL 60616 USA

²University of California, Berkeley, CA 94720 USA

³University of Chicago, Chicago, IL 60637 USA

⁴Princeton University, Princeton, NJ 08544 USA

⁵Argonne National Laboratory, Lemont, IL 60439 USA

Corresponding author: Pranav Gokhale (e-mail: pranav@super.tech)

This work was supported in part by EPiQC, a National Science Foundation (NSF) Expedition in Computing, under Grant CCF-1730449/1730082 and in part by STAQ under Grant NSF Phy-1818914. The work of Kaiwen Gui and Martin Suchara was supported by the U.S. Department of Energy, Office of Science, under Contract DE-AC02-06CH11357.

ABSTRACT Variational quantum eigensolver (VQE) is a promising algorithm for near-term quantum machines. It can be used to estimate the ground state energy of a molecule by performing separate measurements of $O(N^4)$ terms. This quartic scaling appears to be a significant obstacle to practical applications. However, we note that it empirically reduces to $O(N^3)$ when we partition the terms into linear-sized commuting families that can be measured simultaneously. We confirm these empirical observations by studying the MIN-COMMUTING-PARTITION problem at the level of the fermionic Hamiltonian and its encoding into qubits. Moreover, we provide a fast, precomputable procedure for creating linearly sized commuting partitions by solving a round-robin scheduling problem via flow networks. In addition, we demonstrate how to construct the quantum circuits necessary for simultaneous measurement, and we discuss the statistical implication of simultaneous measurement. Our results are experimentally validated by a ground state estimation of deuteron with low shot budget on a 20-qubit IBM machine.

INDEX TERMS Quantum computing, variational quantum eigensolver (VQE).

I. INTRODUCTION

The present Noisy Intermediate-Scale Quantum (NISQ) era [1] is distinguished by the advent of quantum computers comprising tens of qubits, with hundreds of qubits expected in the next five years. Although several thousand logical error-corrected qubits, backed by millions of device-level physical qubits, are needed to realize the originally envisioned quantum applications, such as factoring [2] and database search [3], a new generation of *variational* algorithms has been recently introduced to match the constraints of NISQ hardware.

Variational quantum eigensolver (VQE) [4] is one such algorithm that is widely considered a top contender, if not the top contender, for demonstrating useful quantum speedups. VQE is used to approximate the lowest eigenvalue of a

matrix that is exponentially sized in the number of qubits. This is a very generic eigenvalue problem with a wide class of applications such as molecular ground state estimation [4], maximum three-satisfiability, market split, traveling salesperson [5], and maximum cut [6]. In this article, we focus on the molecular ground state estimation problem, which has already been demonstrated experimentally, though we underscore that the full range of VQE applications is very broad.

VQE solves a similar problem as quantum phase estimation (QPE) [7], [8], an older algorithm that requires large gate counts and long qubit coherence times that are untenable for near-term quantum computers. VQE mitigates these quantum resource requirements by shifting some computational burden to a classical coprocessor. As a result, VQE achieves low gate count circuits and error resilience, but at the cost of

requiring many iterations where each iteration measures one of $O(N^4)$ terms.

This is a daunting scaling factor that poses practical limitations. It was observed that this N^4 scaling could be partly mitigated by performing simultaneous measurement: when two terms correspond to commuting observables, they can be measured in a single state preparation. The article starts from this observation, and we seek to exploit this idea to minimize the total number of state preparations needed.

Our specific contributions include the following.

- 1) Efficient approximation algorithms for partitioning the N^4 terms into commuting families, i.e., approximating the MIN-COMMUTING-PARTITION. This includes an algorithm that partitions the terms into linearly sized families that can be measured simultaneously such that the total measurement cost is only $O(N^3)$.
- 2) A circuit synthesis tool for simultaneous measurement.
- 3) Statistical analysis of simultaneous measurement and a procedure for guarding against harmful covariance terms.
- 4) Validation of these techniques through benchmarks, simulations, and experiments.

The rest of this article is structured as follows. Section II presents relevant background material, and Section III surveys prior work. Section IV analyzes the commutativity of the terms of interest (Pauli strings), and Section V presents a technique for minimizing the number of state preparations by mapping MIN-COMMUTING-PARTITION to a MIN-CLIQUE-COVER instance that can be approximated. Sections VI and VII focus on a molecular Hamiltonian structure and demonstrate approximations of the MIN-COMMUTING-PARTITION that execute in linear time or generate linear-sized partitions, respectively. The latter approach has the effect of reducing the total VQE measurement cost from $O(N^4)$ to $O(N^3)$ for molecular Hamiltonians.

Section VIII shows and analyzes the circuit synthesis procedure that allows simultaneous measurements between commuting Pauli strings. Section IX presents results for our techniques on benchmark molecules, and Section X demonstrates experimental validation. Section XI studies the underlying statistics and discusses a strategy for detecting and correcting course if a partition is harmed by covariance terms. We make concluding remarks and propose future work in Section XII.

II. BACKGROUND

We assume an introductory-level knowledge of quantum computing. We refer newer readers to one of many excellent resources such as [9], [10], or [11].

A. QUANTUM MEASUREMENT

A standard procedure in quantum algorithms is to measure a qubit. In hardware, the standard measurement that can be performed is a measurement in the Z-basis, or computational basis. Fig. 1 depicts such a measurement. The qubit's state

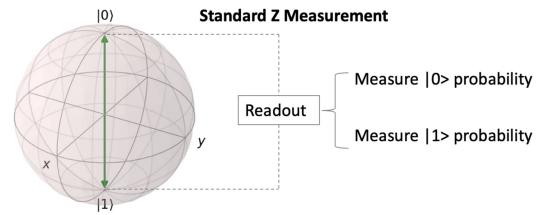


FIGURE 1. Z-basis (computational basis) measurement of a qubit yields $|0\rangle$ or $|1\rangle$ with a probability corresponding to the latitude of the qubit on the Bloch sphere.

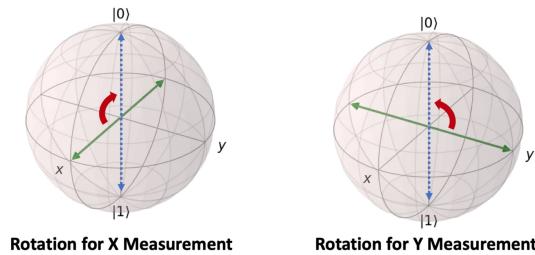


FIGURE 2. Measurement of the X or Y Pauli matrices requires us to first apply a unitary rotation operation that rotates the X- or Y-axis to align with the Z-axis. Subsequently, a standard Z-basis measurement yields the outcome of the X or Y measurement.

is a point on the surface of the Bloch sphere—states with northern latitudes are close to the $|0\rangle$ state and southern latitudes are close to the $|1\rangle$ state. Measurement, or readout, causes the qubit to collapse to either the $|0\rangle$ or $|1\rangle$ state, with a probability dependent on the latitude.

At a more mathematical level, the deeper meaning of measuring a qubit in the

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

basis is to project the qubit's state onto the eigenvectors of the Z operator, which are $|0\rangle$ and $|1\rangle$. In the same sense, we can measure other observables such as the other two *Pauli matrices*

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

The eigenvectors of X are termed $|+\rangle$ and $|-\rangle$, and they are antipodal points along the X-axis of the Bloch sphere. Similarly, Y 's eigenvectors, $|i\rangle$ and $|-i\rangle$, are antipodal along the Y-axis. Since hardware cannot directly measure along these axes, measurements of X (Y) are performed by first rotating the Bloch sphere with a unitary matrix so that the X (Y)-axis becomes aligned with the Z-axis. These rotations are depicted in Fig. 2. Subsequently, a standard Z-basis measurement can be performed, whose outcome can then be mapped to an effective X (Y) measurement.

The specific rotation that accomplishes the X-to-Z axis change is the $R_y(-\pi/2)$ transformation, which is typically captured in quantum circuits by the similar H gate/matrix. The Y-to-Z axis change is accomplished by the $R_x(\pi/2)$

transformation, which is typically captured [12] in quantum circuits by the HS^\dagger gates/matrix.

The same general principle applies toward measuring observables across multiple qubits: measurement is accomplished by applying a quantum circuit that rotates the eigenvectors of the target observable onto the computational basis vectors. The unitary matrix for such a transformation is simply the one that has the orthonormal eigenvectors of the observable as column vectors. In this article, we will be interested in measuring Pauli strings, which are tensor products of Pauli matrices across multiple qubits.

B. SIMULTANEOUS MEASUREMENT AND COMMUTATIVITY

From the preceding discussion, we can see that two observables can be measured simultaneously if they share a common eigenbasis, i.e., they are simultaneously diagonalizable. In this case, they can be measured simultaneously by applying the unitary transformation that rotates their shared eigenbasis onto the computational basis. In the case of Hermitian operators, such as the Pauli strings of interest to us, two observables share an eigenbasis if and only if they commute [13, Ch. 1], i.e., the order of their product is interchangeable.

Moreover, this relationship extends beyond simple pairs: given a family of pairwise commuting observables, there exists a shared eigenbasis that simultaneously diagonalizes *all* of the observables (rather than it merely being a situation in which *each* pair has a separate shared eigenbasis) [14, Th. 1.3.21].

In this article, we will exploit this property to simultaneously measure multiple Pauli string observables with a single state preparation and measurement circuit. Notice that this problem is nontrivial because commutativity is not transitive (and hence, *not* an equivalence class). Consequently, finding optimal partitions of commuting families is a hard problem, as we formalize later.

C. QUANTUM COMPUTATIONAL CHEMISTRY

Quantum computational chemistry has been a long targeted problem on the classical computer. Due to the limits of classical computing resources, we are only able to perform *approximate* classical simulations. Examples include Hartree Fock ($O(N^4)$ runtime [15]), density functional theory ($O(N^3)$ runtime [16]), and coupled cluster single-double ($O(N^6)$)+runtime [17] only considers single and double excitations).

The way to achieve chemical accuracy (1 kcal/mol) is to use full configuration interactions (CIs), which considers all necessary orbitals. Classically, this will generally require $O(\binom{M}{N}) \rightarrow$ exponential runtime [18]. On the other hand, quantum computation is able to encode an exponential amount of molecular information into a polynomial number of qubits and thereby achieve full CI in polynomial time [19].

D. VARIATIONAL QUANTUM EIGENsolver (VQE)

As mentioned previously, VQE can be applied to a wide class of problems that are solvable as minimum-eigenvalue

estimation [5], [6]. In this article, we focus on the application that has received the most commercial and experimental interest: estimating molecular ground state energy. Within the molecular context, we use VQE to approximate the lowest eigenvalue of a matrix, called the Hamiltonian, that captures the molecule's energy configuration. The lowest eigenvalue is the ground state energy, which has important implications in chemistry such as determining reaction rates [20] and molecular geometry [21].

The Hamiltonian matrix for a molecule can be written in the second quantized fermionic form as [22]

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

where a^\dagger (a) is the fermionic raising (lowering) operator, and N is the number of qubits and also the number of molecular basis wavefunctions considered. The h_{pq} and h_{pqrs} terms can be computed classically via electron integral formulas implemented by several software packages [23]–[25]. The second sum in (1) indicates that the fermionic form of the Hamiltonian has $O(N^4)$ terms [26], [27]. It can be translated to qubit form by an encoding such as Jordan–Wigner [28], parity [29], or Bravyi–Kitaev [30], as we will discuss further in Section VI. The resulting qubit form will also have $O(N^4)$ terms, where each term is a Pauli string.

It is difficult to directly and efficiently estimate $\langle H \rangle$, the expected energy of the Hamiltonian under an input state vector. The approach of VQE is to estimate it *indirectly* but efficiently, by employing linearity of expectation to decompose $\langle H \rangle$ into a sum of $O(N^4)$ expectations of Pauli strings, which can each be computed efficiently. In the standard and original formulation of VQE, each of these Pauli strings is measured via a separate state preparation [4].

At its core, VQE can be described as a guess-check-repeat algorithm. Initially, the algorithm *guesses* the minimum energy eigenvector of the Hamiltonian H . Then, it *checks* the actual energy for the guessed eigenvector by summing expected values over $O(N^4)$ directly measurable Pauli strings, as previously described. Finally, it *repeats* by trying a new guess for the minimum energy eigenvector, with the assistance of a classical optimizer that guides the next guess based on past results. The potential quantum speedup in VQE arises from the fact that checking the energy on a classical computer would require matrix multiplication of an exponentially sized state vector; by contrast, the energy can be estimated efficiently with a quantum computer by summing over the expected values of the $O(N^4)$ Pauli strings.

Algorithm 1 presents the pseudocode for VQE, under the standard “Naive” formulation where each Pauli string is measured separately. The resource complexity of VQE is clear from this code: the inner `for` loops run $O(N^4/\epsilon^2)$ times and each iteration requires a separate state preparation and measurement. The outer `while` loop termination condition is dependent on both the classical optimizer and the ansatz—we discuss the latter next.

Algorithm 1: Variational Quantum Eigensolver (VQE).

Result: Approximate ground state energy,

$$\min_{\vec{\theta}} \langle H \rangle_{\psi(\vec{\theta})}$$

$\vec{\theta}_1 \leftarrow$ random angles or initial guess;

$i \leftarrow 1$;

while (*not* classical optimizer termination condition)

do

for $j \in [O(N^4)]$ **do**

for $O(1/\epsilon^2)$ repetitions **do**

 Prepare $\psi(\vec{\theta}_i)$;

 Measure $\langle H_j \rangle_{\psi(\vec{\theta}_i)}$;

end

end

$\langle H \rangle_{\psi(\vec{\theta}_i)} \leftarrow \sum_j \langle H_j \rangle_{\psi(\vec{\theta}_i)}$;

 Record $(\theta_i, \langle H \rangle_{\psi(\vec{\theta}_i)})$;

i++;

 Pick new θ_i via classical optimizer;

end

E. UNITARY COUPLED CLUSTER SINGLE DOUBLE ANSATZ

Since the number of possible state vectors spans an exponentially large and continuous Hilbert space, we seek to restrict the family of candidate energy-minimizing states. Such a family is called an ansatz, and the ansatz state $|\psi(\vec{\theta})\rangle$ is parameterized by a vector of independent parameters $\vec{\theta}$. Since VQE aims to run in polynomial time, the number of parameters should be polynomial. While in this article is applicable to any ansatz, we focus our attention to the unitary coupled cluster single double (UCCSD) ansatz, which has generally been the leading contender for molecular ground state estimation. In addition to having a sound theoretical backing (the coupled cluster approach is the gold standard for computational chemistry [22], [31]), UCCSD is more resilient to barren plateaus in the optimization landscape that are experienced by hardware-oriented ansatzes [22], [32]. Recent work has also demonstrated the experimental superiority of UCCSD to other ansatz types [33].

In terms of the number of qubits (which is also the number of molecular basis wavefunctions) N , the total gate count of UCCSD is $O(N^4)$ [34], [35], which can be parallelized in execution to $O(N^3)$ circuit depth. As a concrete scaling example, a recent 4-qubit, 2-electron UCCSD circuit construction required circuit depth of 100 gates, spanning 150 total gates [33]. This is already out of range of present machines—the experimental work thus far has required many symmetry reductions and approximations to implement UCCSD. The number of parameters in UCCSD with respect to the number of electrons and wavefunctions is $O(N^2\eta^2)$, or $O(N^4)$ under the standard assumption that these two terms are asymptotically related by a constant.

F. MUTUALLY UNBIASED BASES (MUB)

Finally, we give a brief overview of mutually unbiased bases (MUB) [36], [37], a concept in quantum information theory that is connected to our overarching question of maximizing the information learned from a single measurement. In the

TABLE 1. MUB for Two Qubits

Operator 1	Operator 2	Operator 3	Shared Eigenbasis
ZZ	IZ	ZI	Separable
XX	IX	XI	Separable
YY	IY	YI	Separable
XY	ZX	YZ	Entangled
YX	ZY	XZ	Entangled

For the first three bases, the shared eigenbases have fully separable eigenvectors. The last two bases have fully entangled eigenvectors.

case of qubits, MUBs describe a partitioning of the $4^N - 1$ N -qubit Pauli strings (identity is excluded) into commuting families of maximal size. For example, Table 1 shows an MUB for the two-qubit Pauli strings. Notice that each row corresponds to a commuting family. Also note that not all rows are created equal—in the first three rows, the shared eigenbasis features separable eigenvectors. In the last two rows, the shared eigenbasis has entanglement between the two qubits.

It is known that for N qubits, there exists an MUB with $2^N + 1$ rows and $2^N - 1$ Pauli strings per row. This is optimal in the sense that $2^N - 1$ is the maximum possible number of distinct Pauli strings (excluding identity) within a commuting family. In Section V, this result will give us insight into the bounds on our MIN-COMMUTING-PARTITION approach.

III. PRIOR WORK

Some of the theoretical aspects of this article were concurrently and independently developed by two other research groups. The four relevant papers, [38] from Waterloo and [39]–[41] from Toronto all share with this article a high level goal of reducing the cost of VQE by exploiting the simultaneous measurability of commuting Pauli strings. In particular, Jena *et al.* [38] mapped the measurement cost reduction goal to a graph coloring problem. The research works in [39]–[41], which respectively consider qubitwise commutativity and general commutativity (GC) (defined in Section IV), treat measurement cost reduction as a minimum clique cover problem. The core ideas of these four papers can be compared to Sections IV and V in this article.

This article is differentiated by a systems perspective that gives explicit attention to the classical computation costs for compilation and transpilation, as well as quantum overheads. The graph algorithms discussed in [38]–[41] incur impractical classical costs that may undo potential speedups from simultaneous measurement. We remedy this issue by introducing problem-aware techniques that operate on molecular Hamiltonian graphs in polynomial time and hence preserve speedups, as discussed in Sections VI and VII. Also, in Section VIII, we introduce a synthesis tool for simultaneous measurement circuits, in recognition of the fact that simultaneous measurement does incur a quantum overhead in additional gates and coherence requirements. To the best of our knowledge, this is the first synthesis tool that constructs simultaneous measurement circuits efficiently in both the

TABLE 2. State Preparation and Measurement Costs From Prior VQE Experiments That Performed Pauli String Partitioning on an Ad Hoc Basis

Molecule	# Pauli Strings	QWC	GC
H ₂ [43]	4	2	2
LiH [43]	99	25	9
BeH ₂ [43]	164	44	8
H ₂ (Bravyi-Kitaev) [34]	5	3	2
H ₂ (Jordan-Wigner) [34]	14	5	2
H ₂ O [44]	21	3	3

#Pauli strings indicates the number of measurement partitions that would be needed naively. QWC expresses the number of qubit-wise commuting partitions that were actually measured via *ad hoc* inspection—we propose a more formal partitioning procedure in Section V. GC foreshadows the general commuting partitions that our techniques described in Sections IV-C, V, and VI achieve.

classical compilation cost and in the quantum circuit complexity. Sections IX and X present benchmark results and experimental results validating that the classical and quantum costs of simultaneous measurement are worthwhile. Additionally, we study the statistics of simultaneous measurement in Section XI and demonstrate a constructive procedure to guard against corruption from covariance terms.

Prior to 2019, strategies for simultaneous measurement in VQE had not been studied formally, aside from the initial suggestion of measurement partitioning in [42]. Most experimental implementations of VQE, for instance [34], [43], [44], [45], did at least perform measurement partitioning on an *ad hoc* basis, via inspection of the Hamiltonian terms. Inspection is insufficient for larger molecules because the underlying problem is NP-Hard, as described in Section V-A. The improvement in these experimental works due to simultaneous measurement is indicated by the reduction from the # Pauli Strings to QWC (qubitwise commutativity) column in Table 2. The last column considers general commutation (GC) partitioning, which we introduce and evaluate in this article. Even for the small molecules that have been studied experimentally thus far, GC achieves significant cost reductions over both Naive and QWC partitions.

In software implementations, both the OpenFermion [25] and Rigetti PyQuil [46] libraries include augmented with functions for simultaneous measurement via qubitwise commutation: `group_into_tensor_product_basis_sets()` and `group_experiments()`, respectively. However, these software implementations do not consider GC and suffer from at least N^8 scaling in runtime, which may undo the potential speedup from simultaneous measurement.

An alternative perspective on the reduction of measurement cost in VQE was introduced in [47], which takes the approach of transforming molecular Hamiltonians to *create* commutativity and reduce the number of qubits needed. Another prior paper [48] operated in a related mathematical setting, using feedforward measurements to create QWC (though we note that feedforward measurements are equivalent to standard unitary transformations by the principle of deferred measurement [11]).

Aside from state preparation and measurement costs, recent work has focused on improving other elements of the

TABLE 3. Partitioning of $\binom{N=8}{4} = 70$ $a_p^\dagger a_q^\dagger a_r a_s$ Terms Into $\binom{N-1=7}{3} = 35$ Subsets, With Disjoint Indices Between the Two Terms in Each Subset

$a_7^\dagger a_5^\dagger a_3 a_0$	$a_6^\dagger a_4^\dagger a_2 a_1$	$a_7^\dagger a_3^\dagger a_1 a_0$	$a_6^\dagger a_5^\dagger a_4 a_2$
$a_6^\dagger a_5^\dagger a_3 a_0$	$a_7^\dagger a_4^\dagger a_2 a_1$	$a_7^\dagger a_4^\dagger a_2 a_0$	$a_6^\dagger a_5^\dagger a_3 a_1$
$a_7^\dagger a_6^\dagger a_3 a_0$	$a_5^\dagger a_4^\dagger a_2 a_1$	$a_6^\dagger a_4^\dagger a_1 a_0$	$a_7^\dagger a_5^\dagger a_3 a_2$
$a_7^\dagger a_4^\dagger a_3 a_0$	$a_6^\dagger a_5^\dagger a_2 a_1$	$a_6^\dagger a_3^\dagger a_1 a_0$	$a_7^\dagger a_5^\dagger a_4 a_2$
$a_7^\dagger a_5^\dagger a_4 a_0$	$a_6^\dagger a_3^\dagger a_2 a_1$	$a_7^\dagger a_3^\dagger a_1 a_0$	$a_5^\dagger a_6^\dagger a_4 a_2$
$a_6^\dagger a_4^\dagger a_3 a_0$	$a_7^\dagger a_5^\dagger a_2 a_1$	$a_5^\dagger a_3^\dagger a_1 a_0$	$a_7^\dagger a_5^\dagger a_3 a_1$
$a_6^\dagger a_5^\dagger a_4 a_0$	$a_7^\dagger a_3^\dagger a_2 a_1$	$a_6^\dagger a_4^\dagger a_2 a_0$	$a_7^\dagger a_5^\dagger a_3 a_1$
$a_7^\dagger a_6^\dagger a_4 a_0$	$a_5^\dagger a_3^\dagger a_2 a_1$	$a_5^\dagger a_4^\dagger a_1 a_0$	$a_7^\dagger a_6^\dagger a_3 a_2$
$a_5^\dagger a_4^\dagger a_3 a_0$	$a_7^\dagger a_6^\dagger a_2 a_1$	$a_4^\dagger a_3^\dagger a_1 a_0$	$a_7^\dagger a_6^\dagger a_5 a_2$
$a_7^\dagger a_6^\dagger a_5 a_0$	$a_4^\dagger a_3^\dagger a_2 a_1$	$a_6^\dagger a_3^\dagger a_2 a_0$	$a_7^\dagger a_5^\dagger a_4 a_1$
$a_7^\dagger a_5^\dagger a_1 a_0$	$a_6^\dagger a_4^\dagger a_3 a_2$	$a_7^\dagger a_2^\dagger a_1 a_0$	$a_6^\dagger a_5^\dagger a_4 a_3$
$a_7^\dagger a_5^\dagger a_2 a_0$	$a_6^\dagger a_4^\dagger a_3 a_1$	$a_5^\dagger a_3^\dagger a_2 a_0$	$a_7^\dagger a_6^\dagger a_4 a_1$
$a_6^\dagger a_5^\dagger a_1 a_0$	$a_7^\dagger a_4^\dagger a_3 a_2$	$a_6^\dagger a_2^\dagger a_1 a_0$	$a_7^\dagger a_5^\dagger a_4 a_3$
$a_6^\dagger a_5^\dagger a_2 a_0$	$a_7^\dagger a_5^\dagger a_3 a_1$	$a_5^\dagger a_2^\dagger a_1 a_0$	$a_7^\dagger a_6^\dagger a_4 a_3$
$a_7^\dagger a_6^\dagger a_1 a_0$	$a_5^\dagger a_4^\dagger a_3 a_2$	$a_5^\dagger a_4^\dagger a_2 a_0$	$a_7^\dagger a_6^\dagger a_3 a_1$
$a_7^\dagger a_4^\dagger a_1 a_0$	$a_6^\dagger a_5^\dagger a_3 a_2$	$a_4^\dagger a_2^\dagger a_1 a_0$	$a_7^\dagger a_6^\dagger a_5 a_3$
$a_7^\dagger a_6^\dagger a_2 a_0$	$a_5^\dagger a_4^\dagger a_3 a_1$	$a_4^\dagger a_3^\dagger a_2 a_0$	$a_7^\dagger a_6^\dagger a_5 a_1$
$a_3^\dagger a_2^\dagger a_1 a_0$	$a_7^\dagger a_6^\dagger a_5 a_4$		

Such a partitioning is guaranteed to exist for all N divisible by 4, per Baranyai's theorem [63].

VQE pipeline. In the classical stage, the research works in [5], [42], and [49] described improvements to the classical optimizer and those in [50] and [51] presented techniques for optimized pulse-level compilation. At the quantum stage, the research works in [35] and [52] proposed improvements to ansatzes and those in [42] and [53] demonstrated procedures for error mitigation. We note that all of these techniques apply to orthogonal stages of the VQE pipeline, and therefore, can compose directly on top of this article.

IV. ANALYSIS OF COMMUTATIVITY

We analyze the commutativity of the terms present in Hamiltonian decompositions. Two terms A and B commute, if their *commutator* is 0

$$[A, B] := AB - BA = 0 \rightarrow AB = BA.$$

As mentioned in Section II-B, two commuting terms are simultaneously diagonalizable by a shared eigenbasis.

In our case, the terms in an N -qubit Hamiltonian are Pauli strings, which are N -fold tensor products of the Pauli matrices

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

Pauli strings are also referred to in other literature as members of the Pauli group, i.e., G_N .

We seek to analyze when two Pauli strings commute. While most of these results are known, they are usually discussed in the context of the stabilizer formalism and quantum

error correction. We present the elements relevant to VQE here, with foreshadowing of our key techniques.

A. SINGLE-QUBIT CASE

First, let us note the following commutation relations for single-qubit Pauli matrices.

- 1) I commutes with everything else. Specifically, $[I, I] = [I, X] = [I, Y] = [I, Z] = 0$.
- 2) X , Y , and Z commute with themselves. $[X, X] = [Y, Y] = [Z, Z] = 0$.
- 3) The other pairs form a cyclic ordering. In particular, $[X, Y] = 2iZ$, $[Y, Z] = 2iX$, $[Z, X] = 2iY$. Flipping the commutator bracket order negates the result.

B. QUBITWISE COMMUTATIVITY (QWC)

The simplest type of commutativity is qubitwise commutativity (QWC). Two Pauli strings QWCommute if at each index the corresponding two Pauli matrices commute. For instance, $\{XX, IX, XI, II\}$ is a QWC partition, because for any pair of Pauli strings, both indices feature commuting Pauli matrices.

As mentioned in Section III, QWC has been leveraged in past experimental work for small molecules [34], [43]–[45] by *ad hoc* inspection of the Hamiltonian terms. However, Section V-A demonstrates that optimally partitioning Pauli strings into QWC families is NP-hard, so an efficient approximation algorithm is needed for larger Hamiltonians with more Pauli strings.

QWC is also referred to in other work as tensor product basis (TPB) [25], [43], [46], recognizing the fact that for a family of QWC Pauli strings, the vectors in the simultaneous eigenbasis can be expressed as a tensor product across each qubit index, with no entanglement. As shown in Section VIII, this makes simultaneous measurement very easy for QWC partitions.

C. GENERAL COMMUTATIVITY (GC)

QWC is sufficient but not necessary for commutation between Pauli strings. For example, $\{XX, YY, ZZ\}$ is a commuting family, even though none of the pairs are QWC—at both indices, the Pauli matrices always fail to commute. The most general rule for commutation of two Pauli strings is that they must fail to commute at an *even* number of indices—2 in the example of $\{XX, YY, ZZ\}$. We refer to this most general form of commutativity as GC, and its proof is below. Note that QWC is simply the subset of GC corresponding to the case where the number of noncommuting indices is 0 (which is even).

Theorem 1: Consider two N -qubit Pauli strings

$$A = \bigotimes_{j=1}^N A_j \quad B = \bigotimes_{j=1}^N B_j$$

where $A_j, B_j \in \{I, X, Y, Z\}$. A and B commute (GC) iff A_j and B_j fail to commute on an even number of indices.

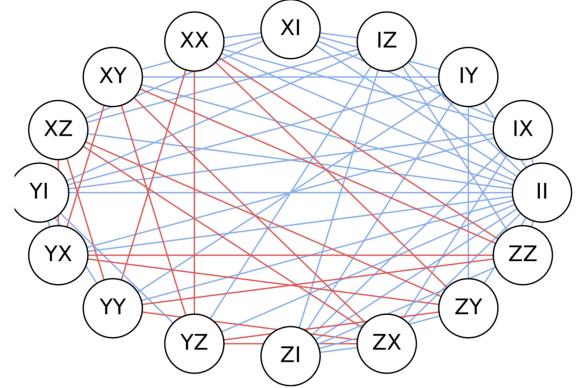


FIGURE 3. This is the commutation graph (also known as a compatibility graph [54]) for all 16 two-qubit Pauli strings. An edge appears when two Pauli strings commute. The blue edges indicate Pauli strings that commute under QWC (which is a subset of GC). The red edges commute under GC-but-not-QWC.

Proof: For Pauli matrices that do not commute, $A_i B_i = -B_i A_i$. Thus, we can write AB as

$$AB = \bigotimes_{j=1}^N A_j B_j = \bigotimes_{j=1}^N \begin{cases} B_j A_j & \text{if } [A_j, B_j] = 0 \\ -B_j A_j & \text{if } [A_j, B_j] \neq 0 \end{cases} = (-1)^k BA$$

where k is the number of indices where $[A_j, B_j] \neq 0$. For AB to equal BA , we require $(-1)^k = 1$, which requires k to be even. Thus, A and B commute iff A_j and B_j fail to commute on an even number of indices. ■

Fig. 3 depicts the commutation relationships between all 16 two-qubit Pauli strings. Edges are drawn between Pauli strings that commute—a blue edge indicates that the pair is QWC and a red edge indicates that the pair is GC-but-not-QWC. The II identity term QWCommutes with every other Pauli string.

V. MIN-CLIQUE-COVER ON HAMILTONIAN

We refer to our core problem of interest as MIN-COMMUTING-PARTITION: given a set of Pauli strings from a Hamiltonian, we seek to partition the strings into commuting families such that the total number of partitions is minimized. While the underlying structure of Pauli matrices and their commutation relationships raises the possibility that MIN-COMMUTING-PARTITION may be efficiently solvable, it turns out to be NP-hard, as we show in Section V-A. Moreover, MIN-COMMUTING-PARTITION is hard even when we only consider the restricted commutativity of QWC. Thus, the *ad hoc* QWC partitioning techniques from past experimental work [34], [43]–[45] are likely to have limited potential for larger molecules.

Instead of solving MIN-COMMUTING-PARTITION exactly, we approximately solve it by mapping to a graph problem as suggestively expressed by the graph representation in Fig. 3. Observe that cliques (fully connected subgraphs where each pair of Pauli strings commutes) are relevant because all of the strings in a clique can be measured

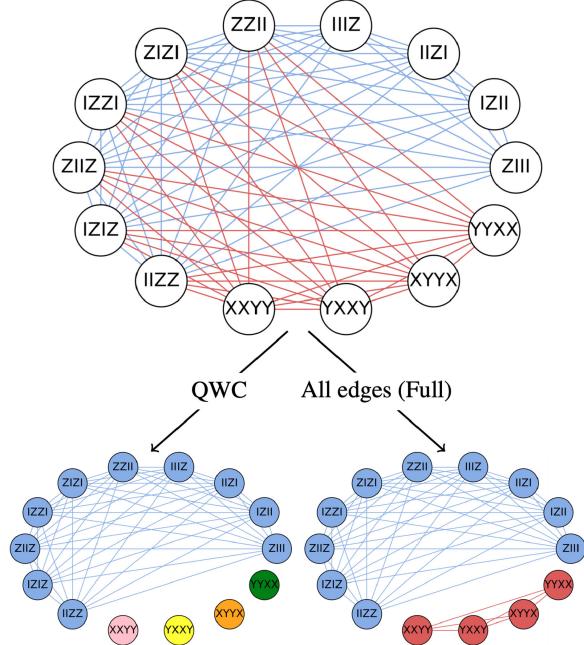


FIGURE 4. Top commutation graph shows both QWC (blue) and GC-but-not-QWC commuting (red) relationships between the Pauli string's in LiH's Hamiltonian. The vertex colors in the bottom two graphs indicate MIN-CLIQUE-COVERs using only QWC edges (left) or using all edges (right). The reduction in measurement partitions from Naive (measuring each Pauli string separately) to QWC to GC is $14 \rightarrow 5 \rightarrow 2$.

simultaneously. Therefore, we seek the MIN-CLIQUE-COVER, i.e., the smallest possible set of cliques whose union spans all vertices. As an example, Fig. 4 shows the commutation graph for LiH's 4-qubit Hamiltonian and its MIN-CLIQUE-COVERs using QWC edges and using GC edges.

MIN-CLIQUE-COVER, in its decision version, is one of the classic Karp NP-complete problems [55], so efficiently finding the minimal possible clique cover for a general graph is unlikely. Moreover, finding a guaranteed “good” clique cover approximation is also NP-hard for general graphs [56]. However, molecular Hamiltonian graphs are highly structured owing both to features of the Pauli commutation graph [57] and to patterns in the Pauli strings that arise in molecular Hamiltonians (we explicitly address and exploit the latter in Section VI). This suggests that MIN-CLIQUE-COVER approximation algorithms may yield reasonably good results. Before discussing the approximation algorithms we used, we discuss bounds on the MIN-CLIQUE-COVER and the relationship to whether the partitions are QWC or GC.

A. NP-HARDNESS OF MIN-COMMUTING-PARTITION

We first show that MIN-COMMUTING-PARTITION is NP-hard. Given a set of operators o_1, o_2 , and, o_n , the MIN-COMMUTING-PARTITION problem partitions the operator set into k subsets such that all operators in each subset pairwise commute and k is minimized. The corresponding decision problem is in NP as it is easy to verify pairwise commutativity for each subset of operators. To show NP completeness, it remains to show the problem is NP-hard.

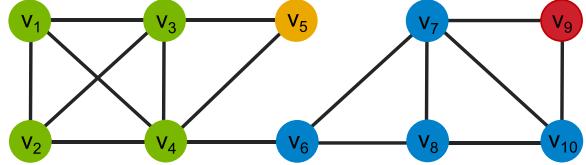


FIGURE 5. Instance of MIN-CLIQUE-COVER (top) and MIN-COMMUTING-PARTITION (bottom).

This can be done by reducing from MIN-CLIQUE-COVER. Given a graph $G = (V, E)$ with n vertices that represents an instance of MIN-CLIQUE-COVER, we produce an instance of MIN-COMMUTING-PARTITION consisting of a set of operators o_1, o_2 , and o_n where each operator o_i has n Paulis, and the j th Pauli is Z if $j = i$, X if $j > i$ and $(v_i, v_j) \notin E$, and I otherwise. This is illustrated in Fig. 5. It is easy to see that a commuting subset of operators corresponds to a clique, which concludes the proof. Notice that the commutativity relationships required in this reduction are only qubitwise commutative, meaning that even the QWC-restricted MIN-COMMUTING-PARTITION problem is NP-hard.

B. MIN-CLIQUE-COVER BOUNDS VIA MUBS

Note that 2^N separate Pauli strings can be measured via a single simultaneous measurement. For instance, consider the 2^N set of Pauli strings of form $(I \text{ or } Z)^{\otimes N}$. All such Pauli strings can be simultaneously measured by simply measuring in the Z -basis on each qubit. This example is suggestive of the power of simultaneous measurement. In the graph picture, it means that cliques exist of size 2^N , which means that simultaneous measurement can lead to an exponential reduction in quantum cost relative to Naive separate measurements.

In the case of VQE, we will consider graphs that have only a polynomially sized ($O(N^4)$) number of Pauli strings. It is still enlightening to consider the MIN-CLIQUE-COVER on the N -qubit graph comprising all $4^N - 1$ possible Pauli strings (in this analysis, we exclude $I^{\otimes N}$, which commutes with everything else). Per the MUB formalism introduced in Section II-F and as suggested in the previous paragraph, a clique of Pauli strings can contain at most $2^N - 1$ vertices. This suggests that at least $2^N + 1$ cliques are needed to cover all $4^N - 1$ possible Pauli strings on N qubits. In fact, this lower bound is exactly attainable—an MUB is exactly such a covering of all N -qubit Pauli strings by disjoint

cliques. Again, this illustrates the potential of simultaneous measurement—a square root reduction is achieved in the total number of state preparations and measurements needed to cover all possible N -qubit Pauli strings.

Many of the partitions produced by MUBs have entanglement in the shared eigenbasis: for example, the bottom two rows of the MUB in Table 1. This means that the MIN-CLIQUE-COVER corresponding to an MUB requires GC edges and not just QWC edges. Next, we further discuss the advantage of GC over QWC.

C. QWC VERSUS GC

GC captures a much denser commutation graph than QWC does, and therefore has more opportunities for larger cliques and thereby smaller clique covers.

We first consider the commutation graph of QWC, over all possible N -qubit Pauli strings; this graph has 4^N vertices. Given a Pauli string with I on k indices, it QWC commutes with exactly $4^k \cdot 2^{N-k} - 1 = 2^{N+k} - 1$ other Pauli strings: on the “partner” string, the k indices are unrestricted and the $N - k$ indices can either match the original Pauli matrix or be I (we subtract 1 to not count the original Pauli string). Since there are $\binom{N}{k} 3^{N-k}$ terms with I on exactly k indices, we see that

$$|E| = \sum_{k=0}^N \frac{\binom{N}{k} 3^{N-k} (2^{N+k} - 1)}{2} = \frac{10^N - 4^N}{2}.$$

This corresponds to an asymptotic graph density of

$$\begin{aligned} \lim_{N \rightarrow \infty} \frac{|E|}{|V|(|V| - 1)/2} &= \lim_{N \rightarrow \infty} \frac{(10^4 - 4^N)/2}{4^N(4^N - 1)/2} \\ &= \lim_{N \rightarrow \infty} (5/8)^N = 0. \end{aligned}$$

In other words, the QWC graph is extremely sparse. By contrast, the GC graph is dense: consider two random Pauli strings. The indicator variable denoting whether the two strings commute on the i th index is a Bernoulli random variable. Therefore, the GC commutation graph corresponds to when the sum over N such independent variables is even, i.e., when a Binomial random variable is even. Asymptotically, this occurs with $\frac{1}{2}$ probability—thus the asymptotic graph density for GC is $\frac{1}{2}$, much denser than for QWC.

Although GC leads to smaller MIN-CLIQUE-COVERs than QWC, QWC does have cheaper simultaneous measurement circuits, as we will see in Section VIII. However, the cost of GC simultaneous measurement will still turn out to be favorable, because circuit costs in VQE are dominated by the ansatz preparation.

D. APPROXIMATION ALGORITHMS TESTED

In our benchmarking, we performed MIN-CLIQUE-COVERs using the Boppana–Halldórsson algorithm [58] included in the NetworkX Python package [59], as well as the Bron–Kerbosch algorithm [60], which we implemented ourselves. These heuristics approximate

a MAX-CLIQUE whose vertices are marked; we then recurse on the residual unmarked graph, repeating until all vertices are marked. We also used the `group_into_tensor_product_basis_sets()` approximation implemented by OpenFermion [25]—this approximation is a nongraph-based randomized algorithm that only finds QWC partitions. Section IX presents results across a range of molecules and Hamiltonian sizes.

While the benchmark results indicate promising performance in terms of finding large partitions, it is critical to also consider the classical computation cost of performing the MIN-CLIQUE-COVER approximation. First, the Bron–Kerbosch algorithm has a worst case exponential runtime. Therefore, its optimality should be interpreted as a soft upper bound on how well other standard approximation algorithms can approximate a MIN-CLIQUE-COVER. The Boppana–Halldórsson algorithm’s runtime is polynomial but is not well studied. Our benchmarks and theoretical analysis indicate roughly quadratic scaling in graph size. Some polynomial benchmarks considered in the other concurrent work scale as much as cubically in the graph size.

However, this poses a problem—the Hamiltonian graph has N^4 terms, so a quadratic or cubic runtime in the number of vertices implies N^8 or N^{12} scaling in classical precomputation time. Beyond simply implying impractical scaling rates, these runtime ranges may exceed the quantum invocation cost of VQE, in which case, we would be better off just running VQE in the Naive fashion. In particular, recall that the UCCSD ansatz has $O(N^3)$ circuit depth after parallelization and that naively, $O(N^4)$ state preparations are needed per ansatz. The total quantum invocation cost of VQE therefore scales as N^7 multiplied by the number of ansatz states explored, though we note that both the ansatz exploration and the Naive $O(N^4)$ measurements could be parallelized given multiple quantum machines. The number of ansatz states explored is an open question that depends on the classical optimizer, the ansatz type, and the variational landscape. Nonetheless, we can make rough estimates by noting that the VQE ansatz has $O(N^4)$ parameters, and rough theoretical results suggest anywhere from $O(N^4)$ iterations under the default SciPy optimization settings [61] to $O(N^{12})$ under matrix inversion techniques. Further work is needed to understand the exact cost of VQE, but there is a strong case that standard graph approximation algorithms may have higher asymptotic cost than simply executing VQE naively without simultaneous measurement optimization. In the case of many expensive MIN-CLIQUE-COVER approximation algorithms, it seems likely that it would be better to simply skip the partitioning step and just measure the Pauli strings naively.

In the next two sections, we remedy this concern by presenting MIN-COMMUTING-PARTITION approximations that exploit our knowledge of the structure of molecular Hamiltonians and their encodings into qubits. The resulting approximation in Section VI algorithm runs in $O(N^4)$ time (linear in the number of Pauli strings, i.e., the graph size),

which is safely below the quantum invocation cost of VQE. The algorithm in Section VII also runs in polynomial time, but guarantees to find linearly sized commuting families.

VI. LINEAR-TIME PARTITIONING

As discussed in the previous section, standard MIN-CLIQUE-COVER approximations may be unsuitable since the classical cost of partitioning can exceed the quantum cost from naively running VQE. This motivates us to inspect features of molecular Hamiltonians and develop a new partitioning strategy accordingly. At a high level, our new strategy is context-aware and attacks the MIN-COMMUTING-PARTITION problem at a different abstraction level, namely the encoding stage from fermionic Hamiltonian to qubit Hamiltonian. By contrast, the previous approximations are unaware of molecular properties.

For convenience, we repeat (1) for the molecular Hamiltonians

$$H = \sum_p^N \sum_q^N h_{pq} a_p^\dagger a_q + \sum_p^N \sum_q^N \sum_r^N \sum_s^N h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s$$

where a^\dagger and a denote raising and lowering operators that act on fermionic modes.

The N^4 scaling of the number of terms in the Hamiltonian is clear from the second summation. In particular, the asymptotically dominant terms are of form $a_p^\dagger a_q^\dagger a_r a_s$ with $p \neq q \neq r \neq s$. These $O(N^4)$ terms are known as the *double excitation* operators [62]. At the scale of smaller molecules, the $O(N)$ terms of form $a_p^\dagger a_p$ and the $O(N^2)$ terms of form $a_p^\dagger a_q^\dagger a_p a_q$ are frequent. These are termed the *number* and *number-excitation* operators, respectively. We will treat both the asymptotically dominant terms and the frequent-for-small-molecules terms in this section.

The commutation relationships of fermions are different from the commutation relationships of qubits. Thus, an encoding step is needed to convert the fermionic Hamiltonian into a qubit Hamiltonian. We consider the most common [22] such encodings: Jordan–Wigner [28], parity [29], and Bravyi–Kitaev [30].

A. JORDAN–WIGNER

Under the Jordan–Wigner encoding, we make the fermion-to-qubit transformations

$$a_p \rightarrow \frac{X_p + iY_p}{2} Z_{p-1}, \dots, Z_0, \quad a_p^\dagger \rightarrow \frac{X_p - iY_p}{2} Z_{p-1}, \dots, Z_0$$

with I on every other index.

1) DOUBLE EXCITATION OPERATORS

For the asymptotically dominant $O(N^4)$ terms of form $a_p^\dagger a_q^\dagger a_r a_s$ (WLOG, $p > q > r > s$), we end up with the 16 Pauli strings matching the regular expression

$$(X_p|Y_p)Z_{p-1}, \dots, Z_{q+1}(X_q|Y_q)(X_r|Y_r)Z_{r-1}, \dots, Z_{s+1}(X_s|Y_s).$$

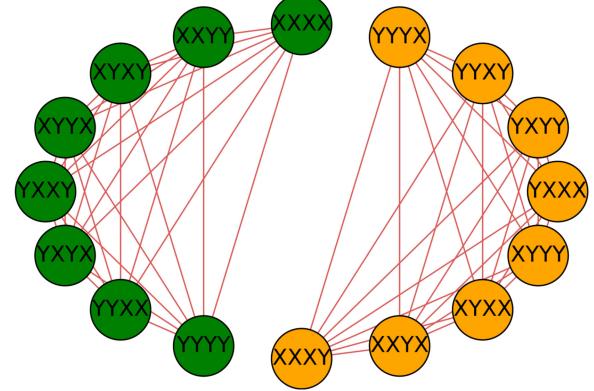


FIGURE 6. 16 relevant Pauli strings in the Jordan–Wigner encoding of $a_p^\dagger a_q^\dagger a_r a_s$ have a MIN-CLIQUE-COVER of size 2.

Thus, we see that the Jordan–Wigner transformation turns each of the N^4 fermionic terms into a sum over 16 Pauli strings. Moreover, these 16 Pauli strings are disjoint from the ones generated by a $a_{p'}^\dagger a_{q'}^\dagger a_{r'} a_{s'}$ term. Consider the commutation graph of the 16 Pauli strings. All indices except for p, q, r , and s immediately commute, so the commutativity graph only needs to consider the p, q, r , and s indices. Fig. 6 depicts the commutation graph, which has a MIN-CLIQUE-COVER of 2. Thus, this yields a strategy for reducing the number of measurement partitions by 8×: we collect all Pauli strings from fermionic terms of form $a_p^\dagger a_q^\dagger a_r a_s$ (and from the 4! permutations of the indices) and measure them using 2 GC partitions instead of 16 Naive partitions.

For molecular Hamiltonians, we generally expect to have $h_{pqrs} = h_{srqp}$, because of the nature of these calculations via integrals and the fact that electrons are indistinguishable. In this case, only eight terms arise (as noted in another context by Whitfield *et al.* [62]), specifically the green 8-clique in Fig. 6. Thus again, we can achieve an 8× reduction.

2) NUMBER AND NUMBER-EXCITATION OPERATORS

While the eightfold reduction in the partitions of the $O(N^4)$ $pqrs$ terms is the asymptotic bottleneck, we also note a useful reduction for the smaller terms that are significant for smaller molecules.

For the $O(N)$ number operators of form $a_p^\dagger a_p$, multiplying out the Jordan–Wigner encoding yields the Pauli string Z_p . For the $O(N^2)$ number-excitation operators of form $a_p^\dagger a_q^\dagger a_p a_q$, the Jordan–Wigner encoding yields the Pauli string $Z_p Z_q$.

Observe that all of these Pauli strings commute and therefore can be simultaneously measured. Moreover, they are QWC, so the simultaneous measurements are cheap, as we will see in Section VIII. While this result may appear obvious from inspection of small molecular Hamiltonians, which have many Pauli strings of form I, \dots, IZI, \dots, I , we underscore that it is not obvious to a context-unaware MIN-CLIQUE-COVER approximation.

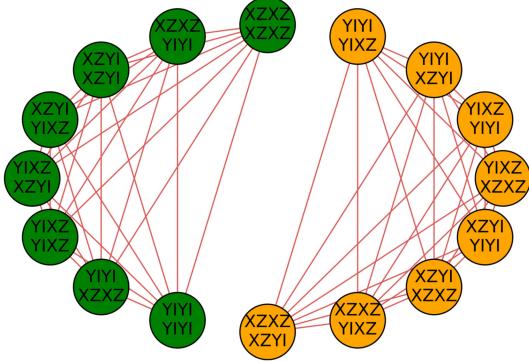


FIGURE 7. Similar to the Jordan–Wigner case, the 16 relevant Pauli strings in the parity encoding of $a_p^\dagger a_q^\dagger a_r a_s$ have a MIN-CLIQUE-COVER of size 2.

B. PARITY ENCODING

For the parity encoding, we make the transformations

$$a_p = X_{N-1}, \dots, X_{p+1} \frac{X_p Z_{p-1} + i Y_p I_{p-1}}{2}$$

$$a_p^\dagger = X_{N-1}, \dots, X_{p+1} \frac{X_p Z_{p-1} - i Y_p I_{p-1}}{2}.$$

1) DOUBLE EXCITATION OPERATORS

WLOG, suppose $p-1 > q, q-1 > r, r-1 > s$. Multiplying out $a_p^\dagger a_q^\dagger a_r a_s$, we see that the parity encoding creates Pauli strings matching the regular expression

$$(X_p Z_{p-1} | Y_p I_{p-1}) X_{p-2} \cdots X_{q+1} (X_q Z_{q-1} | Y_q I_{q-1}) \cdots$$

$$\cdots (X_r Z_{r-1} | Y_r I_{r-1}) X_{r-2} \cdots X_{s+1} (X_s Z_{s-1} | Y_s I_{s-1}).$$

Only indices $p, p-1, q, q-1, r, r-1, s$, and $s-1$ are relevant for commutativity. Once again expanding the resulting 16 Pauli strings, we see that the commutation graph has a MIN-CLIQUE-COVER of size 2, as depicted in Fig. 7. Thus, we can again achieve an 8× reduction in the number of partitions by performing simultaneous measurement across these indices. However, note that the simultaneous measurement circuit now involves eight indices, so it will be more expensive than the simultaneous measurement circuit for the Jordan–Wigner encoding.

2) NUMBER AND NUMBER-EXCITATION OPERATORS

We also again consider the $O(N)$ and $O(N^2)$ operators that are frequent in smaller molecules. The parity encoding on the number and number-excitation operators gives rise to Pauli strings of form $Z_p Z_{p-1}$ and $Z_p Z_{p-1} Z_q Z_{q-1}$, respectively. Again, we see that for small molecules, the parity encoding creates a large set of QWC Pauli strings.

C. BRAVYI-KITAEV

The Bravyi–Kitaev coding is asymptotically favorable for Hamiltonian simulation because it requires asymptotically fewer non- I operators per Pauli string by only selecting

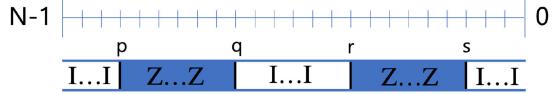


FIGURE 8. Pictorial representation of the Jordan–Wigner encoding of $a_p^\dagger a_q^\dagger a_r a_s$. Repeating Zs span the blue rectangles between p and q and between r and s . The other three ranges have repeating Is. At indices p, r , and s , which are denoted by the black vertical bars between the blue and white rectangles, we can have either X or Y. Thus, there are $2^4 = 16$ Pauli strings involved in the Jordan–Wigner encoding.

a subset of indices to perform partial sums needed in the fermion-to-qubit encoding. As a result, every a_p or a_j^\dagger term involves a subset of indices ($> p$) that carry the X update, and a subset of the indices ($< j$) that require the phase correction. This complicates the commutation structure of $a_p^\dagger a_q^\dagger a_r a_s$ and there is not an immediately obvious clique cover strategy—we identify this as an open question.

VII. $O(N^3)$ MEASUREMENT COST VIA $O(N)$ -SIZED PARTITIONS

We now present an algorithm that runs in $O(N^5 \log N)$ time and produces $O(N)$ -sized commuting families. This reduces the measurement cost of VQE on molecular Hamiltonians from $O(N^4)$ to $O(N^3)$.

A. COMMUTATIVITY OF INDEX-DISJOINT TERMS

We focus on the $O(N^4)$ terms with $p \neq q \neq r \neq s$ in the second sum of (1), because these terms are asymptotically dominant; the number of other terms is only $O(N^3)$. Without loss of generality, let us suppose that $p > q > r > s$, and likewise $i > j > k > l$. We denote the set of Pauli strings in the Jordan–Wigner encoding of $a_p^\dagger a_q^\dagger a_r a_s$ as $\{a_p^\dagger a_q^\dagger a_r a_s\}_{\text{JW}}$.

Our core observation is that if two $a^\dagger a^\dagger aa$ terms have disjoint indices, then the terms in their qubit encodings commute. In particular, consider the following theorem.

Theorem 2: If $\{p, q, r, s\} \cap \{i, j, k, l\} = \emptyset$, then

$$[\{a_p^\dagger a_q^\dagger a_r a_s\}_{\text{JW}}, \{a_i^\dagger a_j^\dagger a_k a_l\}_{\text{JW}}] = 0$$

where the commutator is taken to apply between all pairs of elements between the two sets.

Theorem 2 can be verified by inspecting the form of the Pauli string terms in $\{a^\dagger a^\dagger aa\}_{\text{JW}}$. Under the Jordan–Wigner encoding [28], we perform the transformations

$$a_p \rightarrow \frac{X_p + i Y_p}{2} Z_{p-1} \cdots Z_0, \quad a_p^\dagger \rightarrow \frac{X_p - i Y_p}{2} Z_{p-1} \cdots Z_0.$$

Carrying out the transformation for $a_p^\dagger a_q^\dagger a_r a_s$ yields the 16 Pauli strings matching the regular expression

$$(X_p | Y_p) \bar{Z}_{p:q} (X_q | Y_q) (X_r | Y_r) \bar{Z}_{r:s} (X_s | Y_s)$$

where $\bar{Z}_{p:q}$ denotes Z on each index between p and q , exclusive of endpoints. Fig. 8 shows this pattern as a pictorial representation: the repeating Zs are blue rectangles and the $\{p, q, r, s\}$ indices are the black vertical bars demarcating the blue and white rectangles.



FIGURE 9. Pictorial representation of the commutation on each index between two $\{a_i^\dagger a_j^\dagger a_k a_l\}_{\text{JW}}$ rectangles. All indices commute except possibly the eight indices with black bars—these indices anticommute when the black bar (X or Y) is vertically aligned with a blue rectangle Z . In this example, there are an even (4) number of anticommuting terms, so the two patterns commute.

To evaluate the commutativity between a term in $\{a_p^\dagger a_q^\dagger a_r a_s\}_{\text{JW}}$ and a term in $\{a_i^\dagger a_j^\dagger a_k a_l\}_{\text{JW}}$, we simply need to count the number of indices that anticommute. If the number of anticommuting indices is even, then the two Pauli strings commute. For all indices other than p, q, r, s, i, j, k, l , the Pauli matrices at the indices commute, because $[I, I] = [I, Z] = [Z, I] = [Z, Z] = 0$. On the remaining eight indices, the commutation depends on whether $(X|Y)$ is matched to I (commutes) or Z (anticommutes). Fig. 9 depicts this: when one of the black bars ($X|Y$) is vertically aligned with a blue rectangle (Z), the index does not commute, as marked by the red cross. When the black bar is vertically aligned with a white rectangle (I), the index commutes.

The commutativity between $\{a_p^\dagger a_q^\dagger a_r a_s\}_{\text{JW}}$ and $\{a_i^\dagger a_j^\dagger a_k a_l\}_{\text{JW}}$ terms can be verified by considering all possible interleaved orderings of the eight indices, subject to the constraint that $p > q > r > s$ and $i > j > k > l$. There are $\binom{8}{4} = 70$ such cases that can be explicitly checked (or 35 cases, accounting for symmetry) to prove Theorem 2. Fig. 10 demonstrates four representative cases, which provide useful intuition for the general case. In particular, when sliding one of the $\{p, q, r, s\}$ indices while keeping $\{i, j, k, l\}$ fixed, the parity of the number of anticommuting indices is invariant. Thus, this parity is always even, and two $\{a_p^\dagger a_q^\dagger a_r a_s\}_{\text{JW}}$ and $\{a_i^\dagger a_j^\dagger a_k a_l\}_{\text{JW}}$ terms with disjoint indices always commute, as claimed in Theorem 2.

B. EXISTENCE OF LINEARLY SIZED PARTITIONS

Consider the set of Pauli strings contained in

$$\begin{aligned} &\{a_N^\dagger a_{N-1}^\dagger \dots a_{N-2}^\dagger a_{N-3}\}_{\text{JW}} \cup \{a_8^\dagger a_7^\dagger a_6 a_5\}_{\text{JW}} \cup \dots \cup \\ &\times \{a_4^\dagger a_3^\dagger a_2 a_1\}_{\text{JW}} \end{aligned}$$

for N divisible by 4. There are $16 \frac{N}{4} = 4N \in O(N)$ Pauli strings in this set. However, since the indices are disjoint, Pauli strings from each of the $\frac{N}{4}$ subsets can be measured simultaneously by Theorem 2. In particular, the Pauli strings can be partitioned into $16 \in O(1)$ measurement families. In fact, they can even be partitioned into just two measurement families by noting that the MIN-COMMUTING-PARTITION within each $\{a_p^\dagger a_q^\dagger a_r a_s\}_{\text{JW}}$ term is 2, as described previously in Section VI.

A natural question is whether all $\binom{N}{4} p > q > r > s$ terms in (1) can be partitioned in such a fashion—if so, then this constitutes a partitioning of the $O(N^4)$ terms into $\binom{N}{4}/\frac{N}{4} =$

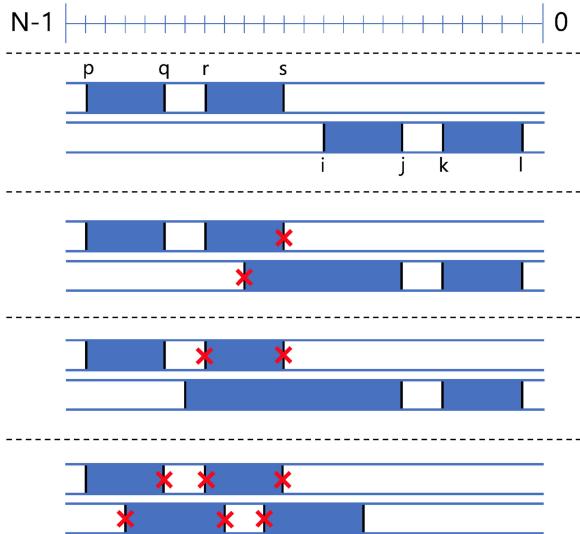


FIGURE 10. Four representative examples illustrating why $\{a_p^\dagger a_q^\dagger a_r a_s\}_{\text{JW}}$ and $\{a_i^\dagger a_j^\dagger a_k a_l\}_{\text{JW}}$ terms always commute (have an even number of anticommuting indices) when $\{p, q, r, s\} \cap \{i, j, k, l\} = \emptyset$. At the top, no black bars align with blue rectangles, so there are 0 anticommuting indices. Below, $r > i > s > j$, so there are two anticommuting indices: i and s . Below that, observe that sliding the i endpoint into the interval between q and r does not change the parity of the number of anticommuting indices. The bottom example shows a case with the maximal number of anticommuting indices, i.e., 6.

$\binom{N-1}{3} \in O(N^3)$ commuting families. Intuitively, this is the same problem as trying to schedule a round-robin tournament of N players with four players-per-game into $\binom{N-1}{3}$ rounds. We can think of each index as a player, and four-player games can be scheduled simultaneously if they do not share players. Equivalently, these problems can be bijected to a graph theory problem: does the four-uniform complete hypergraph on N vertices admit a 1-factorization?

The answer to all of these questions is affirmative, per Baranyai's theorem [63]. In our case, it means that for N divisible by 4, the $\binom{N}{4} \in O(N^4)$ terms can be partitioned into $\binom{N-1}{3} \in O(N^3)$ sets, such that the $\frac{N}{4}$ terms within each set have disjoint indices. Table 1 demonstrates such a partitioning for $N = 8$ qubits. Each of the $\binom{8-1}{3} = 35$ rows has two fermionic terms with disjoint indices—thus, their corresponding Jordan–Wigner qubit encodings can be measured simultaneously.

C. CONSTRUCTION OF LINEARLY SIZED PARTITIONS

Prior literature refers to Baranyai's original proof as either being nonconstructive [64], [65] or providing an exponential-time construction [66] (prior literature varies in what exactly is considered Baranyai's proof). In order for Baranyai's proof to be useful to us, we need a fast polynomial-time algorithm for partitioning the $\binom{N}{4}$ subsets of N into $\binom{N-1}{3}$ groups, each containing $N/4$ disjoint subsets. Fortunately, due to later work by Brouwer and Schrijver [67], a proof was provided that leads to an efficient construction [68]. The proof is based on maximum flows in network flow graphs.

Algorithm 2: $O(N^5 \log N)$ Baranyai Construction.

```

input :  $N$ 
output:  $\binom{N-1}{3}$  sets where each set contains  $\frac{N}{4}$ 
    disjoint size-4 subsets, and no term is
    repeated
for  $i \in [1, 2, \dots, N]$  do
    Create flow network with two layers:  $O(N^3)$ 
    partition nodes and  $O(N^3)$  subset nodes;
    Set capacities for  $O(N^4)$  edges per [67, 69]
    construction;
    Set fractional maxflow of value  $\binom{N-1}{3}$ , saturating
    nodes out of source and into destination;
    Round fractional maxflow into integral maxflow;
    Update subset nodes based on integral maxflow;
end
Return schedule of  $\binom{N}{4}$  subsets, based on final flow;

```

We refer readers to [69] for a lucid explanation and to [70] for an implementation in code. This implementation was used to generate Table 3. The pseudocode is given in Algorithm 2. An outer loop is called N times, and each iteration solves for maximum flow on a network with $O(N^3)$ vertices and $O(N^4)$ directed edges. Since the maximum flow in the proof construction has a value of $O(N^3)$, solving for it with the Ford–Fulkerson algorithm would incur a cost of $O(N^7)$ per loop iteration [71]. However, due to work on flow rounding [72]–[75], this runtime is reduced to $O(N^4 \log N)$. This is because for each flow network, a *fractional* solution is known that can be rounded to an integral solution faster than computing an integral solution from scratch. Thus, the total runtime of the Baranyai constructive proof is $O(N^5 \log N)$.

A useful aspect of the Baranyai-based approach to molecular Hamiltonian partitioning is that it depends only on N and not on the h_{pq} and h_{pqrs} coefficients in (1). In this sense, it is precomputable—for instance, the $N = 8$ partitioning in Table 1 will apply to all 8-qubit Hamiltonians. By contrast, MIN-COMMUTING-PARTITION techniques in prior work operate on the specific molecular Hamiltonians of interest. Thus, the partitionings are not precomputable and the classical cost of partitioning must be accounted for in time to solution.

In summary, we have demonstrated that Jordan–Wigner encoded molecular Hamiltonians can be partitioned into $O(N^3)$ commuting families, each containing $O(N)$ Pauli strings. Our proof stems from Baranyai’s theorem, which has a constructive form that efficiently yields partitionings, per Algorithm 2. Since commuting families can be measured simultaneously, this constitutes a reduction in the measurement cost of VQE from $O(N^4)$ naively to $O(N^3)$ with these partitions. Beyond VQE, this technique may be useful in other quantum computational chemistry applications. For example, the simulation of Hamiltonian *dynamics* could be improved by partitioning into commuting families [76].

VIII. CIRCUITS FOR SIMULTANEOUS MEASUREMENT

Once an approximate MIN-COMMUTING-PARTITION solution has been generated, a natural question arises of how to actually perform the necessary simultaneous measurement for each commuting partition. In the case of Naive partitions where each Pauli string is measured separately, the measurement circuit is trivial. In particular, recall from Section II that we simply perform the H and HS^\dagger operations on the indices with X or Y , respectively, and then we measure every qubit in the Z -basis. Thus, we need just $O(N)$ fully parallelizable single-qubit gates; more specifically, we require $k \leq N$ single-qubit gates, where k is the number of indices in the Pauli string that equal X or Y .

Simultaneous measurement is also similarly straightforward in the case of QWC partitions. Each index of a QWC partition is characterized by a measurement basis. For example, consider the task of simultaneously measuring the two QWC Pauli strings $XIYIZI$ and $IXIYIZ$. We simply apply H to the left two qubits and HS^\dagger to the right two qubits. The resulting qubits can all be measured in the standard Z -basis, and the corresponding outcomes indicate the X, X, Y, Y, Z , and Z outcomes as desired. In terms of circuit cost, QWC measurement is essentially identical to Naive measurement: $O(N)$ single-qubit gates are required, and the gates are fully parallelizable to constant depth.

While Naive and QWC partition measurements are straightforward, GC partition measurements are nontrivial. We now introduce a circuit synthesis procedure enabling these measurements, and we analyze both the quantum and classical costs of this procedure. To the best of our knowledge, this is the first work explicitly demonstrating how to perform simultaneous measurement in the general case of GC Pauli strings. We implemented our circuit synthesis tool as a Python library [77] and validated it across a wide range of molecular Hamiltonians.

A. BACKGROUND

As discussed in Section II, performing a simultaneous measurement amounts to applying a unitary transformation in which the columns of the unitary matrix are the simultaneous eigenvectors of the commuting Pauli strings in the partition. After applying such a transformation and then performing standard Z -basis measurements, the outcomes are mapped directly to measurements of the Pauli strings of interest. One approach to synthesize a simultaneous measurement circuit would be to explicitly compute the matrix of simultaneous eigenvectors and then apply one of many possible unitary decomposition techniques [78]–[83] to this matrix. However, this approach is not sufficient for two reasons. First, in general, decomposition techniques tradeoff between requiring intractable quantum circuit depth, requiring intractable classical compilation time, and yielding only approximations to the desired transformation. Second, and most importantly, these techniques require us to compute the simultaneous eigenvectors and input them to the decomposer. In general, the simultaneous eigenvectors resulting from GC can be fully

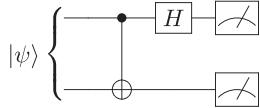


FIGURE 11. Bell basis measurement circuit that simultaneously measures XX , YY , and ZZ on the $|\psi\rangle$ state. After application of these two gates, the measurements of the top and bottom qubits correspond to outcomes for XX and ZZ , respectively. The YY outcome is obtained from $YY = -(XX)(ZZ)$.

entangled across all N indices, and they are represented by a 2^N -sized column vector. The corresponding unitary matrix would be doubly exponentially sized in N , erasing any potential quantum advantage.

With this in mind, it is clear that any decomposition technique must avoid explicitly computing eigenvectors and writing out exponentially sized unitary matrices. Fortunately, the stabilizer formalism—typically applied to quantum error correction—provides us such a mechanism. Before proceeding, we note that this article is built upon the language of stabilizers introduced in [84] and expanded upon in [85]. While these two papers were applied to error correction and quantum simulation, the core techniques also apply to our use case. Also, Seyfarth and Ranade [86] and [87] leveraged these stabilizer techniques to perform MUB measurements. Our circuit constructions are drawn from these two papers as well as [88], but stem from a different context and end goal.

B. EXAMPLE: (XX, YY, ZZ)

We begin with a well-known example. Consider the task of trying to simultaneously measure XX , YY , and ZZ , a GC (but not QWC) partition. The simultaneous eigenvectors of these Pauli strings are known as the four *Bell states*

$$\begin{aligned} |\Phi^+\rangle &= \frac{|00\rangle + |11\rangle}{\sqrt{2}} & |\Phi^-\rangle &= \frac{|00\rangle - |11\rangle}{\sqrt{2}} \\ |\Psi^+\rangle &= \frac{|01\rangle + |10\rangle}{\sqrt{2}} & |\Psi^-\rangle &= \frac{|01\rangle - |10\rangle}{\sqrt{2}}. \end{aligned}$$

These eigenvectors are linearly independent and span all possible 2-qubit states—hence, they are a basis. Unlike the vectors in the standard computational basis of $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, the eigenvectors in the Bell basis feature entanglement between the two qubits. As a result, measurement in the Bell basis requires interaction between the two qubits, unlike the Naive and QWC measurements described previously. The quantum circuit in Fig. 11 is a well-known circuit that performs Bell basis measurement, i.e., simultaneous measurement of XX , YY , and ZZ .

To understand why this circuit measures XX and ZZ (and also $YY = -(XX)(ZZ)$), we observe that our ultimate goal is to transform a target measurement of $[XX, ZZ]$ into $[ZI, IZ]$ —the latter captures the outcomes we actually measure directly via standard Z-basis measurement. An important background result is that after applying some unitary operation U , a target measurement M on the original state

has become equivalent to a measurement of UMU^\dagger [11], [84] on the new state. This is known as *unitary conjugation*.

In the Bell basis measurement circuit, we first apply $U = \text{CNOT}$. By computing UMU^\dagger , we can see that target measurements of $[XX, ZZ]$ are transformed under conjugation to measurements of

$$[XX, ZZ] \xrightarrow[U=\text{CNOT}]{UMU^\dagger} [UXXU^\dagger, UZZU^\dagger] = [XI, IZ].$$

Finally, after applying the Hadamard gate on the top qubit, the measurements are transformed to

$$[XI, IZ] \xrightarrow[U=H\otimes I]{UMU^\dagger} [UXIU^\dagger, UIZU^\dagger] = [ZI, IZ].$$

Thus, this CNOT, $H \otimes I$ gate sequence performs the desired transformation of rotating a measurement of $[XX, ZZ]$ into the computational basis, i.e., $[ZI, IZ]$. The ordering of the elements is important and indicates that measurement of the top qubit (ZI) corresponds to the XX outcome and measurement of the bottom qubit (IZ) corresponds to the ZZ outcome. As mentioned previously, YY follows as $-(XX)(ZZ)$.

C. STABILIZER MATRICES

In order to consider the general case, we now switch to the formalism of stabilizer matrices. Our notation and terminology is similar to previous work [84]–[88], with some deviations for clarity. Within the stabilizer formalism, every N -qubit Pauli string maps to a $2N$ -entry column vector. The top N entries indicate whether each corresponding index “contains” a Z . The bottom N entries correspond to X s. The Y Pauli matrix corresponds to having a 1 in both the Z and X entries, since $Y = iZX$. The stabilizer matrix for a list of Pauli strings is simply the concatenation of the column vectors. As an instructive example, the stabilizer matrix for $[XXX, YYY, ZZZ, XYZ]$ is

$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix}.$$

For convenience and clarity, we will refer to the top N rows as the *Z-matrix* and the bottom N rows as the *X-matrix*. Recall that our goal is to transform a target set of Pauli strings for simultaneous measurement into the computational basis measurements, $[ZII \cdots I, IZI \cdots I, \dots, III \cdots Z]$. We see that the stabilizer matrix for this computational basis simply has an $N \times N$ identity as the *Z-matrix* and all zeroes in the *X-matrix*.

We now seek a procedure to transform the target stabilizer matrix into this computational basis stabilizer matrix. To see how to accomplish such a transformation, it is useful to know unitary conjugation relationships for a basic gate set. Tables 4

TABLE 4. Result of Conjugation of Z and X by Single-Qubit Gates
 $U = H$ or S

	UZU^\dagger	UXU^\dagger
$U = H$	X	Z
$U = S$	Z	Y

TABLE 5. Result of Conjugation of ZI , IY , XI , or IX by Two-Qubit Gates
 $U = \text{CNOT}$, CZ , or SWAP

	$UZIU^\dagger$	$UIZU^\dagger$	$UXIU^\dagger$	$UIXU^\dagger$
$U = \text{CNOT}$	ZI	ZZ	XX	IX
$U = \text{CZ}$	ZI	IY	XZ	ZX
$U = \text{SWAP}$	ZI	ZI	IX	XI

Note that H can be thought of as a “NOT gate” between X and Z . The S (phase) gate does not affect Z , but does transform X into Y .

and 5 list the unitary conjugations of important Pauli strings for one- and two-qubit unitary gates, respectively.

Based on these tables, we can interpret the action of each of these unitaries on a stabilizer matrix. These rules can be verified directly from the tables and are also explained in [85] and [87].

- 1) H on the i th qubit swaps the i th and $i + N$ th rows of the stabilizer matrix (i.e., swaps between corresponding rows of the Z and X matrices). It is helpful to think of H as a “NOT gate” that flips Z and X measurements.
- 2) S on the i th qubit sets the (i, i) diagonal entry in the Z -matrix to 0.
- 3) CNOT controlled on the i th qubit and targeted on the j th qubit adds the j th row to the i th row and adds the $i + N$ th row to the $j + N$ th row. All additions are performed modulo 2.
- 4) CZ between the i th and j th qubits sets the (i, j) and (j, i) symmetric off-diagonal entries of the Z -matrix to 0.
- 5) SWAP between the i th and j th qubits swaps the i th and j th rows of both the Z and X matrices. This can be seen from the fact that $\text{SWAP} = (\text{CNOT})(\text{NOTC})(\text{CNOT})$ and two rows can be swapped with three alternating binary additions.

D. CIRCUIT SYNTHESIS PROCEDURE

We now have the tools we need for circuit synthesis, which amounts to transforming the stabilizer matrix for a commuting family of Pauli strings into the computational basis stabilizer matrix (which has identity for the Z -matrix and zeros for the X -matrix). For simplicity, we describe the procedure for the case when the partition of N -qubit Pauli strings is complete and contains N linearly independent elements. This is the hardest case—if the partition is incomplete, the measurement procedure is similar but has more slack.

The circuit synthesis procedure is described in Algorithm 3. To develop its intuition, we demonstrate its application to the problem of simultaneously measuring $[IYX, ZZZ, XIX, ZXY]$, which is a GC (but not QWC) family. We initialize the algorithm by setting the stabilizer matrix to a basis of this partition. Note that the fourth term

Algorithm 3: Circuit Synthesis for Sim. Measurement.

```

input :  $\{P_i\}$ , a complete GC family of Pauli strings
output: Circuit for simultaneous measurement of
 $\{P_i\}$ 
 $M \in F_2^{2N \times N} \leftarrow$  basis of  $\{P_i\}$ ;
Full-rankify  $X$ -matrix by applying  $H$  gates;
Gaussian eliminate  $X$ -matrix using CNOT & SWAP
gates;
for each diagonal element in  $Z$ -matrix do
| if element is 1 then apply  $S$  to corresponding
| qubit;
end
for each element below diagonal of  $Z$ -matrix do
| if element is 1 then apply  $CZ$  to the row-col
| qubits;
end
Apply  $H$  to each qubit;
Measure each qubit;
```

is linearly dependent on the first three, so we exclude it to yield such a basis; in general, we use Gaussian elimination to perform this distillation of the Pauli strings into a basis. The stabilizer matrix for this resulting list of Pauli strings, i.e., $[IYX, ZZZ, XIX]$, is

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

The first step of the simultaneous measurement circuit synthesis is to apply H gates as needed to transform the X -matrix to have full rank (it is currently only rank 2). Such a transformation is always possible and can be found efficiently by Gaussian elimination [85, Lemma 6]. In this case, applying H to the first qubit swaps the first and fourth rows of the stabilizer matrix, yielding an X -matrix of full rank 3

$$\xrightarrow{\begin{array}{c} \boxed{H} \\ \hline \hline \end{array}} \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

Now that the X -matrix is of full rank, we can apply standard Gaussian elimination to row reduce it into the identity matrix. The CNOT and SWAP gates give us the elementary row operations needed: add one row to another and swap rows. In this example, the X -matrix can be row reduced to the identity by first adding its second row to the third row, and then swapping the first and second rows. Breaking this down, we first observe the effect of the CNOT on the stabilizer

matrix

$$\begin{array}{c} \text{---} \\ \text{---} \\ \bullet \\ \text{---} \\ \oplus \end{array} \rightarrow \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Finally, the SWAP completes the row reduction, leaving the X-matrix as the following identity:

$$\begin{array}{c} \text{---} \\ \text{---} \\ \times \\ \text{---} \\ \text{---} \end{array} \rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Notice that the CNOT and SWAP also affected the Z-matrix, which is now a symmetric matrix; this is guaranteed to occur [87]. Now, our desired transformation is almost complete. The on-diagonal 1 is erased with S on the first qubit, and the two off-diagonal 1s are erased with a CZ between the second and third qubits. These two operations have no effect on the X-matrix, as shown in the following:

$$\begin{array}{c} \boxed{S} \\ \text{---} \\ \bullet \\ \text{---} \end{array} \rightarrow \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Finally, we apply an H to each qubit, which swaps the Z and X matrices, leaving us in the computational basis stabilizer matrix, as desired

$$\begin{array}{c} \boxed{H} \\ \boxed{H} \\ \boxed{H} \end{array} \rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

The full circuit and resulting transformation is shown as follows:

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} \rightarrow \begin{array}{c} \boxed{H} \quad \times \quad \boxed{S} \quad \boxed{H} \\ \text{---} \quad \text{---} \\ \bullet \quad \times \quad \bullet \quad \boxed{H} \\ \text{---} \quad \text{---} \\ \oplus \quad \text{---} \quad \bullet \quad \boxed{H} \end{array} \rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

E. CIRCUIT COMPLEXITY

The efficiency of Algorithm 3 and the overarching stabilizer formalism stems from the fact that the stabilizer matrices are of size $2N \times N$, and all manipulations are on this tractably sized matrix. This averts the exponential cost that manipulating simultaneous eigenvectors would entail. In terms of classical cost, the synthesis tool is fast because its slowest step is the Gaussian elimination, which has time complexity of $O(N^3)$ [89].

The actual circuit produced by the synthesis procedure requires only $O(N^2)$ gates in the worst case, as also noted in related results [87], [88]. This follows because the Gaussian elimination can require $O(N^2)$ elementary row operations, which entails $O(N^2)$ CNOT gates. The erasure of off-diagonal elements in the Z-matrix also requires $O(N^2)$ CZ gates.

While the $O(N^2)$ gate count for GC measurement is worse scaling than the $O(N)$ gate count for Naive or QWC measurement, we emphasize that the measurement circuit is preceded by an ansatz preparation circuit that dominates gate counts and depth. In particular, the UCCSD ansatz has $O(N^4)$ gate count and $O(N^3)$ depth after parallelization. Therefore, the cost of simultaneous measurement is asymptotically insignificant. As discussed, we base our studies on UCCSD because the coupled cluster approach is the gold standard for quantum computational chemistry [22], [31]. Moreover, UCCSD has shown experimental and theoretical promise, unlike hardware-driven ansatz, which were shown to suffer from “barren plateaus” in the optimization landscape [22], [32]. Even in the case of other nonhardware-driven ansatzes, gate counts and depths generally scale at least as N^3 in order to achieve high accuracy. Thus, the quadratic cost of GC measurement appears to be benign.

We also underscore that the $O(N^2)$ gate count scaling of simultaneous measurement is a worst case scenario, where our partition is dominated by GC-but-not-QWC edges. In practice, this is not the case and we see QWC on many, if not most indices. For example, in the linear-time MIN-COMMUTING-PARTITION $8\times$ approximations presented in Section VI only a constant (4 or 8) number of Pauli string indices have a GC-but-not-QWC relationship in the simultaneous measurements. The remaining $N - 4$ or $N - 8$ Pauli string indices are QWC. Thus, under this MIN-COMMUTING-PARTITION approximation, the simultaneous measurement circuit gate count is still $O(N)$ and the depth is still parallelizable to $O(1)$.

For reference, we show in Fig. 12 the simultaneous measurement circuit for the 4 GC-but-not-QWC qubits in the Pauli partition for the Jordan–Wigner transformation. Specifically, this measurement circuit is used to measure the green 8-clique in Fig. 6. The other $N - 4$ qubits are QWC and require single-qubit gates for measurement—this is why the simultaneous measurement gate complexity is still just $O(N)$.

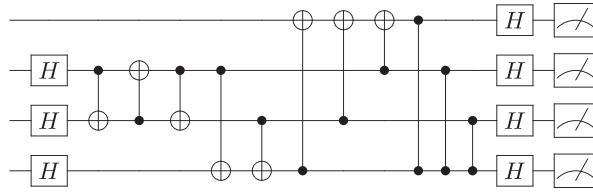


FIGURE 12. Simultaneous measurement circuit generated by our software for the green 8-clique in Fig. 6. It transforms the measurements of $XXXX$, $XYYX$, $XYXY$, $YXXX$ (which is a basis for the Pauli strings in the green eight-clique) to measurements of $ZIII$, $IZII$, $IIIZ$, $IIIZ$.

F. MEASUREMENT CIRCUIT OPTIMIZATIONS

While the circuit synthesis procedure in Algorithm 3 yields a correct simultaneous measurement circuit, it is not necessarily the most optimal circuit possible. For instance, in Fig. 12, the SWAP (implemented as three CNOTs) between qubits 2 and 3 can be omitted from the circuit and instead implemented by swapping their subsequent gates, and then accounting for the SWAP classically after the measurements are performed. In other words, the SWAPs in our circuit constructions can be accomplished by simple classical relabeling of qubit indices.

We also observe that many gates can be parallelized. For example, the depth of Fig. 12 can be reduced by parallelizing the execution of the CZ gates with the execution of the CNOT gates.

IX. BENCHMARK RESULTS

We tested the performance of our simultaneous measurement strategies in Section V on multiple molecular benchmarks, whose Hamiltonians we obtained via OpenFermion [25]. Our benchmark results encompass both the reduction in number of partitions relative to Naive, as well as the classical computation runtime required to produce the partitioning.

As mentioned, in Section V, the Bron–Kerbosch based MIN-CLIQUE-COVER approximation has exponential worst case runtime and should thus be considered a soft bound on the optimality of partitions produced by other graph approximation algorithms. Fig. 13 indicates the performance of Bron–Kerbosch in terms of number of commuting partitions (cliques) found using both QWC and GC edges, in comparison to the Naive VQE implementation in which each Pauli string is in a singleton partition. The improvement from Naive to QWC is consistently about $4\text{--}5\times$ —a significant reduction especially considering that QWC measurement is cheap. The improvement from Naive to GC ranges from $7\times$ to $12\times$ from H_2 to CH_4 (methane). This suggests that the state preparation cost reduction factor from GC partitioning improves for larger molecules.

Figs. 14 and 15 examine partitioning efficacy when we vary the qubit encodings and the number of active spaces considered for the H_2 molecule. Across the qubit encodings, performance is roughly consistent with a $3\times$ improvement from QWC partitions and a $10\times$ improvement from GC partitions. We do note one outlier in that the performance is particularly promising for the Brayvi–Kitaev superfast encoding

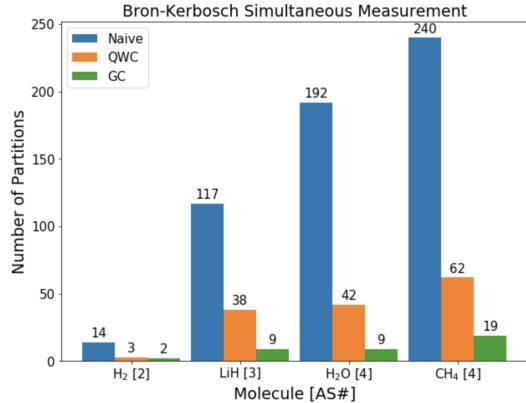


FIGURE 13. Number of QWC and GC partitions (which we are attempting to minimize) generated by Bron–Kerbosch for four representative molecules. AS# indicates the number of active spaces for the molecular Hamiltonian.

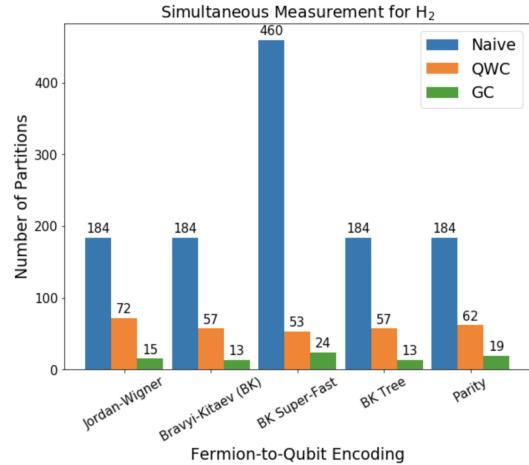


FIGURE 14. Number of QWC and GC partitions generated by Bron–Kerbosch for the H_2 molecule, under different fermion-to-qubit encodings.

[30], which achieves a $20\times$ reduction in the number of partitions from Naive to GC. Across the varying active spaces, we again see evidence that the GC partitioning advantage scales with Hamiltonian size, ranging from $3\times$ to $12\times$ as the number of active spaces is increased. This is important and encouraging, because prior work demonstrated that a relatively large number of active spaces are needed to achieve chemical accuracy [90].

Along with the Bron–Kerbosch approximations as a loose upper bound on the expected partitioning optimality, we also benchmarked another MIN-CLIQUE-COVER approximation: the Boppana–Halldórsson algorithm, applied to both QWC- and GC-edge graphs. In addition, we also benchmarked with the QWC partitioning heuristic provided by the OpenFermion electronic structure package. We tested each of these algorithms on problem sizes ranging from 4 to 5237 terms in the molecular Hamiltonian. These Hamiltonians correspond to the H_2 , LiH , H_2O , and CH_4 molecules with varying numbers of active spaces. We recorded both the number of partitions generated and the runtime for each

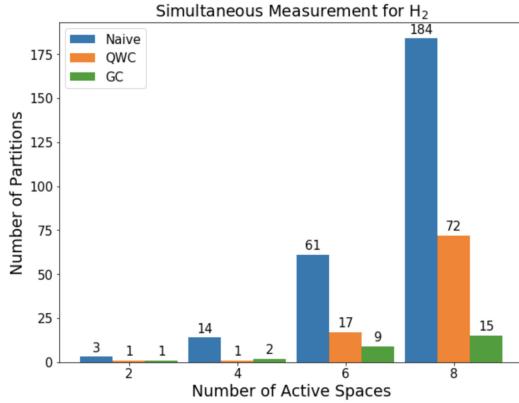


FIGURE 15. Number of QWC and GC partitions generated by Bron–Kerbosch for the H_2 molecule, under different numbers of active spaces.

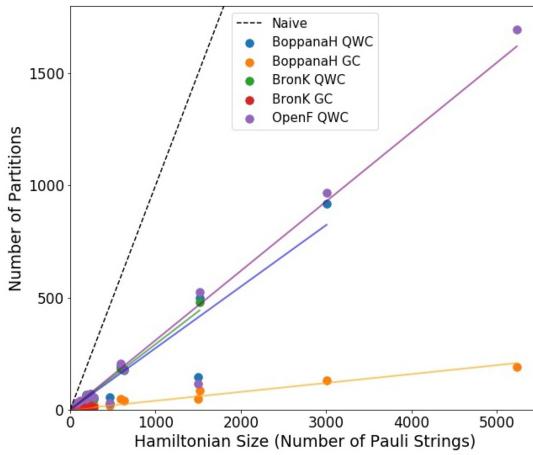


FIGURE 16. Number of partitions found for each algorithm–benchmark pair. Under Naive measurement, the number of partitions would exactly equal the Hamiltonian size (number of Pauli strings). Thus, these techniques all achieve a 4–20× reduction in state preparations and measurements relative to the Naive strategy.

algorithm–benchmark pair. Fig. 16 shows the number of partitions generated for Hamiltonians with up to 5237 Pauli strings. Note that some of the benchmarks were unable to be run due to prohibitive runtime costs on the order of days (e.g., Bron–Kerbosch for $|H| > 1519$ Pauli strings). Fig. 17 shows a zoom-in for molecules with up to 630 Pauli strings; the y -axis now shows the reduction factor in number of partitions. The plots generally align with our expectations: GC leads to much more optimal partitioning than QWC (recall the arguments in Section V-C, and Bron–Kerbosch GC achieves the fewest number of partitions generated although Boppana–Halldórsson GC has comparable optimality). Among the QWC methods, we consistently see 3–4× reductions in number of partitions over Naive separate measurements, and our Boppana–Halldórsson QWC algorithm marginally outperforms the OpenFermion heuristic.

Fig. 18 plots the wall clock runtimes for each of the algorithm–benchmark pairs; Fig. 19 focuses on the 0–630 Hamiltonian size range. These plots corroborate the

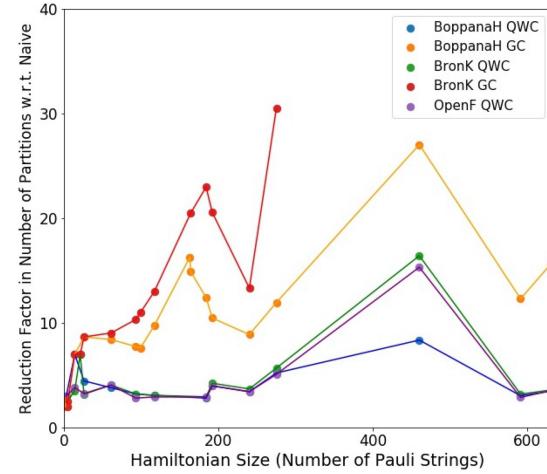


FIGURE 17. Factor of improvement (which we are attempting to maximize) over Naive for each of the algorithms benchmarked for Hamiltonian sizes up to 630 terms.

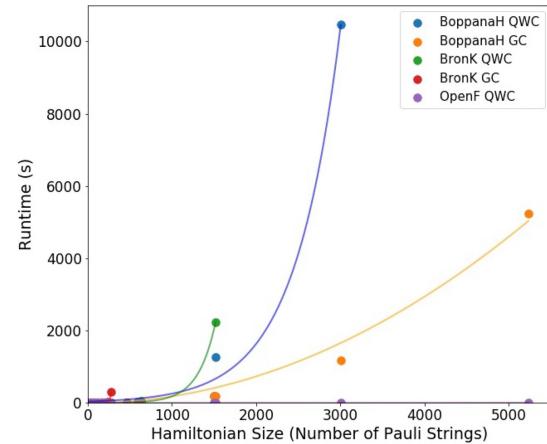


FIGURE 18. Classical computer runtimes for each partitioning algorithm + benchmark pair. Bron–Kerbosch has exponential and Boppana–Halldórsson has quadratic runtime scaling. This partitioning step runs as a compilation procedure before the actual quantum invocations of VQE.

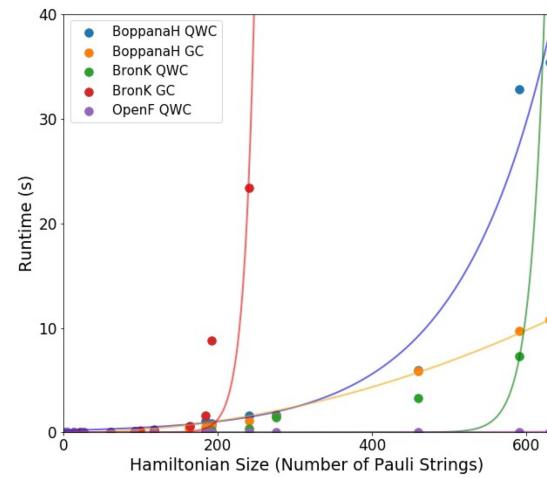


FIGURE 19. Zoom-in of Fig. 18 for Hamiltonian sizes up to 630 terms.

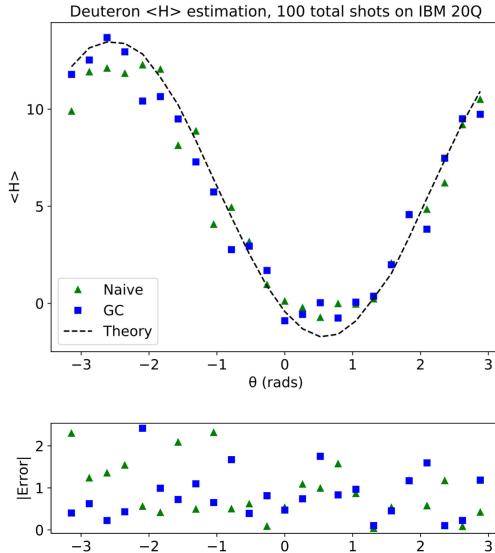


FIGURE 20. Deuteron energy estimation under Naive and GC partitions, as executed on IBM Q20 with a low total shot budget of 100. The energies are in MeV. Average error is 11% lower with GC simultaneous measurement than with Naive separate measurements.

exponential worst case scaling of Bron–Kerbosch and suggest quadratic runtime scaling for the Boppana–Halldósson algorithm. OpenFermion’s function is clearly the fastest of the algorithms explored, but is also consistently the worst approximation to the MIN-COMMUTING-PARTITION.

X. EXPERIMENTAL RESULTS

We validated our techniques with a proof of concept demonstration by experimentally replicating a recent result [91]: ground state energy estimation of deuteron, the nucleus of an uncommon isotope of hydrogen. We performed our experiments via the IBM Q Tokyo 20-qubit quantum computer [92], which is cloud accessible.

Following [91], deuteron can be modeled with a 2-qubit Hamiltonian spanning four Pauli strings:¹ IZ , ZI , XX , and YY . Under Naive measurement, each Pauli string is measured in a separate partition. Under GC, we can partition into just two commuting families: $\{ZI, IZ\}$ and $\{XX, YY\}$. Recall that the former partition is QWC and can be measured with simple computational basis measurements. The latter partition can be measured by the Bell basis measurement circuit in Fig. 11.

To establish a fair comparison between Naive measurement and simultaneous measurement, we performed experiments in which both settings were allocated an equal budget in total number of shots (trials) allowed. We first considered a resource-constrained setting with a budget of 100 total shots. This corresponds to 25 shots per partition in Naive measurement and 50 shots per partition in GC simultaneous measurement. Fig. 20 plots our results for a simplified unitary coupled cluster ansatz with a single parameter and

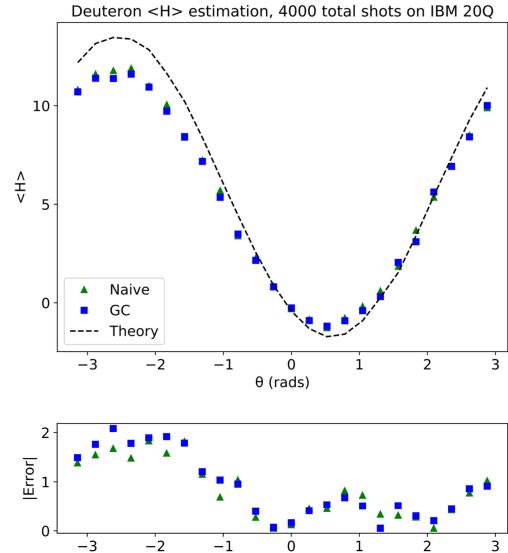


FIGURE 21. Deuteron energy estimation under Naive and GC partitions, as executed on IBM Q20 with a total shot budget of 4000. The energies are in MeV. Average error is 7% lower with Naive separate measurements than with GC simultaneous measurements.

just three gates (two single-qubit rotations and one CNOT), as described in [91].

The results indicate reasonable agreement between Naive measurement, GC measurement, and the true (theory) values. The deviation from theory stems both from statistical variance due to the low shot budget, as well as systematic noise in the quantum processes. As Fig. 20’s lower—error—plot indicates, for 13 of the 24 values swept across the θ range, GC measurement had lower error than Naive measurement. On average, the GC measurements had an error of 835 keV—11% less than the average error of 940 keV for Naive measurement.

We also ran another experiment with a much higher total shot budget of 4000 (i.e., 1000 shots per partition in Naive and 2000 for GC). In this regime, errors due to systematic quantum noise should dominate over errors from statistical variation. We expect GC simultaneous measurement to exhibit more systematic noise because it requires an extra CNOT gate as per the Bell measurement circuit in Fig. 11. Therefore, we expect better results from Naive measurement than from GC simultaneous measurement. Fig. 21 plots the experimental results.

For 17 of the 24 values swept across the θ range, Naive measurement does indeed outperform GC simultaneous measurement in terms of lower error. The respective average errors are 848 and 914 KeV, indicating a 7% higher accuracy with Naive measurement.

These results are presented as proof-of-concept that simultaneous measurement achieves higher accuracy when the shot budget is limited. Equivalently, we can achieve equal accuracy with fewer shots (i.e., fewer state preparations) when the shot budget is limited. For several reasons, we note that these experimental results *underestimate* the potential of simultaneous measurement, especially as higher quantum

¹There is also an II term, but this does not actually require any measurement—it just adds a constant offset to the Hamiltonian.

volume devices emerge. In particular, the following statements hold.

- 1) The unitary coupled cluster ansatz of [91] is highly simplified and does not yet exhibit the asymptotic $O(N^4)$ scaling. Our argument that simultaneous measurement is cheap hinges on the comparison between $O(N^4)$ ansatz gate count and $O(N^2)$ simultaneous measurement gate count. For this simplified ansatz and small N , simultaneous measurement essentially doubled the gate count. As lower error devices emerge with the ability to support the full UCCSD ansatz gate count and larger qubit count N , simultaneous measurement circuits will become a negligible cost.
- 2) For a small Hamiltonian like the one considered here, the partitioning gain from GC is only $2\times$. As indicated in the benchmark results in Section IX, we expect up to $30\times$ gains for larger Hamiltonians and possibly a gain factor that continues to linearly increase for larger molecules, based on extrapolation of the benchmark results.
- 3) For current machines, the number of jobs is far more costly than the number of shots for practical purposes, since executions are scheduled at the granularity of jobs. In our executions, we saw this as an immediate and practical advantage of simultaneous measurement. Our total latency was dominated by the number of jobs rather than the number of shots, so our simultaneous measurement results were collected much more rapidly than Naive measurement results, even though both settings had equal total shot budgets.

We reiterate that these results should only be interpreted as a proof of concept. As machines improve, we expect to see dramatically better results, for the aforementioned reasons.

XI. STATISTICS OF SIMULTANEOUS MEASUREMENT: GUARDING AGAINST COVARIANCES

We have now shown both how to approximate a MIN-COMMUTING-PARTITION and how to actually construct the requisite simultaneous measurement circuits. Finally, we now address an important question regarding the statistics of simultaneous measurement. This question was first raised by McClean *et al.* [42, Sec. IV B2], which proved that simultaneous measurement can actually *underperform* separate measurements due to the presence of covariance terms. In particular, while simultaneous measurement does not bias the estimate $\langle \hat{H} \rangle$, it can increase the variance of the estimator, relative to separate measurements.

In this section, we first show a specific example from [42] in which simultaneous measurement is *suboptimal*. Then, we prove that such examples are atypical and that the MIN-COMMUTING-PARTITION is still optimal when we have no prior on the ansatz state. Finally, we demonstrate an adaptive strategy for detecting and correcting course in the atypical case when a simultaneous measurement should be split into separate measurements.

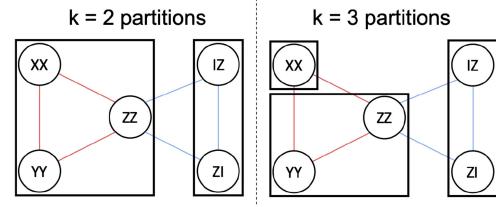


FIGURE 22. Commuting-family partitions of $H = IZ + ZI - XX - YY + ZZ$ with $k = 2$ and $k = 3$.

A. EXAMPLE

Consider the Hamiltonian, $H = IZ + ZI - XX - YY + ZZ$, following the example of [42]. The commutation graph has a bowtie shape. Fig. 22 depicts two possible clique partitionings with $k = 2$ and $k = 3$ commuting-family partitions, respectively.

Thus far, we have worked under the assumption that estimating $\langle H \rangle$ is more efficient with simultaneous measurement than with separate measurements and we have therefore targeted MIN-COMMUTING-PARTITIONs. However, consider a case in which the ansatz state is $|01\rangle$, for the previously stated Hamiltonian.

Since the outcomes of our measurements are random, we quantify the uncertainty around our estimate of the expectation value by $\text{Var}(\langle H \rangle)$. Our end goal is to determine the expected value of the Hamiltonian to a target accuracy level ϵ . The expected number of state preparations, n_{expect} , needed to achieve this accuracy for a k -way partitioning is [42]

$$n_{\text{expect}} = \frac{k \sum_{i=1}^k \text{Var}(\text{Partition } i)}{\epsilon^2}. \quad (2)$$

The variance from each partition can be computed from the formula for the variance of a sum of terms

$$\text{Var}\left(\left\{\sum_{i=1}^n M_i\right\}\right) = \sum_{i=1}^n \text{Var}(M_i) + 2 \sum_{1 \leq i < j \leq n} \text{Cov}(M_i, M_j)$$

where $\text{Cov}(M_1, M_2) = \langle M_1 M_2 \rangle - \langle M_1 \rangle \langle M_2 \rangle$ and $\text{Var}(M) = \text{Cov}(M, M)$.

In our case with $|\psi\rangle = |01\rangle$, the primitives evaluate to: $\text{Var}(IZ) = \text{Var}(ZI) = \text{Var}(ZZ) = 0$ and $\text{Var}(-XX) = \text{Var}(-YY) = 1$. All covariances are 0 except for $\text{Cov}(-XX, -YY) = 1$.

For the $k = 2$ partitioning, we have

$$\begin{aligned} n_{\text{expect}} &= \frac{2 [\text{Var}(\{-XX, -YY, ZZ\}) + \text{Var}(\{ZI, IZ\})]}{\epsilon^2} \\ &= 2 [\text{Var}(-XX) + \text{Var}(-YY) + \text{Var}(ZZ) \\ &\quad + 2\text{Cov}(-XX, -YY) + 2\text{Cov}(-XX, ZZ) \\ &\quad + 2\text{Cov}(-YY, ZZ) \\ &\quad + \text{Var}(ZI) + \text{Var}(IZ) + 2\text{Cov}(IZ, ZI)] / \epsilon^2 \\ &= \boxed{8/\epsilon^2}. \end{aligned}$$

For the $k = 3$ partitioning, we have

$$\begin{aligned} n_{\text{expect}} &= \frac{3 [\text{Var}(\{-XX\}) + \text{Var}(\{-YY, ZZ\}) + \text{Var}(\{IZ, ZI\})]}{\epsilon^2} \\ &= 3 [\text{Var}(-XX) + \text{Var}(-YY) + \text{Var}(ZZ) \\ &\quad + 2\text{Cov}(-YY, ZZ) + \text{Var}(ZI) + \text{Var}(IZ) \\ &\quad + 2\text{Cov}(IZ, ZI)] / \epsilon^2 \\ &= \boxed{6/\epsilon^2}. \end{aligned}$$

Thus, due to the contribution of positive covariance between $-XX$ and $-YY$, the $k = 3$ partitioning is better than the $k = 2$ partitioning for this $(H, |\psi\rangle)$ combination.

This phenomenon motivates us to pay close attention to covariances within each partitioning. The worst case scenario is that we end up with positive covariances within each partition. In a best case scenario, we will have negative covariances within each partitioning, which could dramatically reduce the number of state preparations needed to achieve some desired error on $\langle H \rangle$.

B. TYPICAL CASE

We now observe that examples such as the previous one, in which the MIN-COMMUTING-PARTITION is suboptimal, are atypical. Below, we prove that when we have no prior on the ansatz state $|\psi\rangle$, the expected covariance between two commuting Pauli strings is 0. This validates the general goal of finding the MIN-COMMUTING-PARTITION, because under 0 covariances, the only strategy for reducing n_{expect} in (2) is to minimize the total number of partitions k .

Theorem 3: Given M_1 and M_2 , two commuting but non-identical Pauli strings, $\mathbb{E}[\text{Cov}(M_1, M_2)] = 0$, where the expectation is taken over a uniform distribution over all possible state vectors (the Haar distribution [93], [94]).

Proof: We consider the following two exhaustive cases.

- 1) Either M_1 or M_2 is I . WLOG, suppose $M_1 = I$. Then, $\text{Cov}(M_1, M_2) = \langle I \cdot M_2 \rangle - \langle I \rangle \langle M_2 \rangle = 0$.
- 2) Neither M_1 nor M_2 is I . Since M_1 and M_2 are Pauli strings that have only $+1$ and -1 eigenvalues, the eigenspace can be split into $M_1, M_2 = (-1, -1), (-1, +1), (+1, -1)$, and $(+1, +1)$ subspaces. Moreover, these subspaces are equally sized (proof follows from stabilizer formalism [11, Ch. 10.5.1]). Let us write $|\psi\rangle$ as a sum over projections into these subspaces

$$|\psi\rangle = a |\psi_{-1,-1}\rangle + b |\psi_{-1,+1}\rangle + c |\psi_{+1,-1}\rangle + d |\psi_{+1,+1}\rangle$$

Under this state, the covariance is $\text{Cov}(M_1, M_2)_{|\psi\rangle} = \langle M_1 M_2 \rangle - \langle M_1 \rangle \langle M_2 \rangle = (|a|^2 - |b|^2 - |c|^2 + |d|^2) - (-|a|^2 - |b|^2 + |c|^2 + |d|^2)(-|a|^2 + |b|^2 - |c|^2 + |d|^2)$.

Now, consider the matching state

$$\begin{aligned} |\psi'\rangle &= b |\psi_{-1,-1}\rangle + a |\psi_{-1,+1}\rangle + d |\psi_{+1,-1}\rangle \\ &\quad + c |\psi_{+1,+1}\rangle. \end{aligned}$$

Under $|\psi'\rangle$, the covariance is $\text{Cov}(M_1, M_2)_{|\psi'\rangle} = \langle M_1 M_2 \rangle - \langle M_1 \rangle \langle M_2 \rangle = (|b|^2 - |a|^2 - |d|^2 + |c|^2) - (-|b|^2 - |a|^2 + |d|^2 + |c|^2)(-|b|^2 + |a|^2 - |d|^2 + |c|^2)$. Thus, $\text{Cov}(M_1, M_2)_{|\psi\rangle} = -\text{Cov}(M_1, M_2)_{|\psi'\rangle}$. Since each $|\psi\rangle$ is matched by this symmetric $|\psi'\rangle$ state, and our expectation is over a uniform distribution of all possible state vectors, we conclude that $\mathbb{E}[\text{Cov}(M_1, M_2)] = 0$. ■

C. MITIGATING COVARIANCES: PARTITION SPLITTING

While we have now secured the top level goal of initially performing measurements under the MIN-COMMUTING-PARTITION approximation, it is still important to detect and correct course if covariances do turn out to harm our measurement statistics. We now introduce such a strategy that adaptively splits partitions to mitigate harmful covariances.

Our strategy is based on building sample covariance matrices of commuting Pauli strings. If M_1 , M_2 , and M_3 are Pauli strings, recall that the covariance matrix, $\text{Cov}([M_1, M_2, M_3])$, under a fixed state is expressed as follows:

$$\begin{pmatrix} \text{Var}(M_1) & \text{Cov}(M_1, M_2) & \text{Cov}(M_1, M_3) \\ \text{Cov}(M_2, M_1) & \text{Var}(M_2) & \text{Cov}(M_2, M_3) \\ \text{Cov}(M_3, M_1) & \text{Cov}(M_3, M_2) & \text{Var}(M_3) \end{pmatrix}.$$

Or, in shorthand notation, where $\text{Var}(M_1) = \sigma_{M_1}^2$ and $\text{Cov}(M_1, M_2) = \sigma_{M_1 M_2}$

$$\begin{pmatrix} \sigma_{M_1}^2 & \sigma_{M_1 M_2} & \sigma_{M_1 M_3} \\ \sigma_{M_2 M_1} & \sigma_{M_2}^2 & \sigma_{M_2 M_3} \\ \sigma_{M_3 M_1} & \sigma_{M_3 M_2} & \sigma_{M_3}^2 \end{pmatrix}.$$

Note that for commuting matrices M_1 and M_2 , we have $\text{Cov}(M_1, M_2) = \langle M_1 M_2 \rangle - \langle M_1 \rangle \langle M_2 \rangle = \langle M_2 M_1 \rangle - \langle M_2 \rangle \langle M_1 \rangle = \text{Cov}(M_2, M_1)$, so covariance matrices are symmetric around the main diagonal.

We now return to the pathological example from Section XI-A. Since the variance of a partitioning is the sum of all entries in each partition's covariance matrix, the sum of the shaded terms below represents the variance of the $k = 2$ partitioning ($\{-XX, -YY, ZZ\}, \{ZI, IZ\}$)

$$\begin{pmatrix} \sigma_{-XX}^2 & \sigma_{-XX, -YY} & \sigma_{-XX, ZZ} & \sigma_{-XX, ZI} & \sigma_{-XX, IZ} \\ \sigma_{-YY, -XX} & \sigma_{-YY}^2 & \sigma_{-YY, ZZ} & \sigma_{-YY, ZI} & \sigma_{-YY, IZ} \\ \sigma_{ZZ, -XX} & \sigma_{ZZ, -YY} & \sigma_{ZZ}^2 & \sigma_{ZZ, ZI} & \sigma_{ZZ, IZ} \\ \sigma_{ZI, -XX} & \sigma_{ZI, -YY} & \sigma_{ZI, ZZ} & \sigma_{ZI}^2 & \sigma_{ZI, IZ} \\ \sigma_{IZ, -XX} & \sigma_{IZ, -YY} & \sigma_{IZ, ZZ} & \sigma_{IZ, ZI} & \sigma_{IZ}^2 \end{pmatrix}.$$

The sum of the shaded terms below represents the variance of the $k = 3$ partitioning ($\{-XX\}$, $\{-YY, ZZ\}$, $\{ZI, IZ\}$)

$$\begin{pmatrix} \sigma_{-XX}^2 & \sigma_{-XX,-YY} & \sigma_{-XX,ZZ} & \sigma_{-XX,ZI} & \sigma_{-XX,IZ} \\ \sigma_{-YY,-XX} & \sigma_{-YY}^2 & \sigma_{-YY,ZZ} & \sigma_{-YY,ZI} & \sigma_{-YY,IZ} \\ \sigma_{ZZ,-XX} & \sigma_{ZZ,-YY} & \sigma_{ZZ}^2 & \sigma_{ZZ,ZI} & \sigma_{ZZ,IZ} \\ \sigma_{ZI,-XX} & \sigma_{ZI,-YY} & \sigma_{ZI,ZZ} & \sigma_{ZI}^2 & \sigma_{ZI,IZ} \\ \sigma_{IZ,-XX} & \sigma_{IZ,-YY} & \sigma_{IZ,ZZ} & \sigma_{IZ,ZI} & \sigma_{IZ}^2 \end{pmatrix}.$$

Therefore, it is favorable (fewer state preparations needed to achieve a target accuracy) to break the $-XX$ term out of the $\{-XX, -YY, ZZ\}$ partition if the condition atop the next page holds. The matrices represent a sum over enclosed terms, and the multiplicative factors of $k = 2$ and $k = 3$ follow from (2)

$$2 \begin{pmatrix} \sigma_{-XX}^2 & \sigma_{-XX,-YY} & \sigma_{-XX,ZZ} \\ \sigma_{-YY,-XX} & \sigma_{-YY}^2 & \sigma_{-YY,ZZ} \\ \sigma_{ZZ,-XX} & \sigma_{ZZ,-YY} & \sigma_{ZZ}^2 \\ & & \sigma_{ZI}^2 & \sigma_{ZI,IZ} \\ & & \sigma_{IZ,ZI} & \sigma_{IZ}^2 \end{pmatrix}$$

$$> 3 \begin{pmatrix} \sigma_{-XX}^2 & \sigma_{-YY}^2 & \sigma_{-YY,ZZ} \\ \sigma_{ZZ,-YY} & \sigma_{ZZ}^2 & \sigma_{ZI}^2 & \sigma_{ZI,IZ} \\ & & \sigma_{IZ,ZI} & \sigma_{IZ}^2 \end{pmatrix}$$

or equivalently, if

$$\begin{pmatrix} 2\sigma_{-XX,-YY} & 2\sigma_{-XX,ZZ} \\ 2\sigma_{-YY,-XX} \\ 2\sigma_{ZZ,-XX} \end{pmatrix}$$

$$> \begin{pmatrix} \sigma_{-XX}^2 & \sigma_{-YY}^2 & \sigma_{-YY,ZZ} \\ \sigma_{ZZ,-YY} & \sigma_{ZZ}^2 & \sigma_{ZI}^2 & \sigma_{ZI,IZ} \\ & & \sigma_{IZ,ZI} & \sigma_{IZ}^2 \end{pmatrix}. \quad (3)$$

Informally, notice that the left-hand side of (3) is a multiple of the sum of the covariances that exist in the expression for $\text{Var}(k = 2)$ but not $\text{Var}(k = 3)$ (which we will call the “broken terms”), whereas the right-hand side is a multiple of the sum of the variances and covariances that exist in both the $\text{Var}(k = 2)$ and $\text{Var}(k = 3)$ expressions (the “unbroken terms”). This pattern generalizes such that it is favorable to switch from a partitioning with k partitions to a clique-splitting partitioning with $k' > k$ partitions if

$$k \times (\sum \text{broken terms}) > (k' - k) \times (\sum \text{unbroken terms}).$$

A similar strategy was described in [43, Sec. V.A.], for the special case of comparing Naive partitions (with no covariances) with QWC partitions; this article generalizes to

the case of comparing two non-Naive partitions where both sides have covariance terms.

D. STRATEGIES FOR COVARIANCE ESTIMATION

As demonstrated in Section XI-A, the expected number of state preparations needed to determine $\langle H \rangle$ to an accuracy level ϵ can be calculated if the variances and pairwise covariances of commuting Pauli terms under an ansatz state are known.

In practice, the true theoretical values of these variances cannot be known beforehand, as doing so would require computations involving the exponentially sized ansatz state vector. However, just as we use repeated measurements from partitions of commuting terms to approximate the expected value of their sum, *we can use these same measurements to approximate the covariance matrices of Pauli strings in the same partition*. This estimation of covariance is termed “sample covariance,” since its value is calculated via a sample from the theoretical distribution. This key idea of adaptively building a sample covariance matrix, using the measurements we are already making, allows us to adaptively detect and correct for harmful covariance terms.

Note that the theoretical variance of $\langle M \rangle$ is $\text{Var}(M) = \langle M^2 \rangle - \langle M \rangle^2$, and is approximated by the sample variance, $\widehat{\text{Var}}(M) = \frac{1}{n-1} \sum_{i=1}^n (m_i - \bar{m})$, where $\{m_1, \dots, m_n\}$ represent the n observed measurements of M , and where $\bar{m} = \frac{1}{n} \sum_{i=1}^n m_i$ is the sample mean. Similarly, the theoretical covariance $\text{Cov}(M_1, M_2) = \langle M_1 M_2 \rangle - \langle M_1 \rangle \langle M_2 \rangle$ is approximated by the sample covariance $\widehat{\text{Cov}}(M_1, M_2) = \frac{1}{n-1} \sum_{i=1}^n (m_{1i} - \bar{m}_1)(m_{2i} - \bar{m}_2)$ where $\{m_{11}, \dots, m_{1n}\}$ and $\{m_{21}, \dots, m_{2n}\}$ are the n observed measurements of M_1 and M_2 , respectively.

Since covariance terms can only be approximated if terms are simultaneously measured, we ideally want to start our measurements in a setting with MIN-COMMUTING-PARTITIONS. Fortunately, this is exactly the optimal starting strategy that we initialize with, as per the argument in Section XI-B. Once we collect sufficiently many observations that the sample covariance matrices stabilize, this will enable us to identify opportunities to split partitions in order to lower variances and thus reduce the number of requisite state preparations.

To make this concrete, let us again consider the $k = 2$ partitionings from the previous example, $\{-XX, -YY, ZZ\}$, $\{ZI, IZ\}$. As we accumulate more observations, we can empirically build up an approximation of each partition’s sample covariance matrices, so

$$\begin{pmatrix} \hat{\sigma}_{-XX}^2 & \hat{\sigma}_{-XX,-YY} & \hat{\sigma}_{-XX,ZZ} \\ \hat{\sigma}_{-YY,-XX} & \hat{\sigma}_{-YY}^2 & \hat{\sigma}_{-YY,ZZ} \\ \hat{\sigma}_{ZZ,-XX} & \hat{\sigma}_{ZZ,-YY} & \hat{\sigma}_{ZZ}^2 \\ & & \hat{\sigma}_{ZI}^2 & \hat{\sigma}_{ZI,IZ} \\ & & \hat{\sigma}_{IZ,ZI} & \hat{\sigma}_{IZ}^2 \end{pmatrix}.$$

Since the sample covariance matrix $\widehat{\text{Var}}(k = 2)$ contains a superset of the terms needed to calculate $\widehat{\text{Var}}(k = 3)$, we can

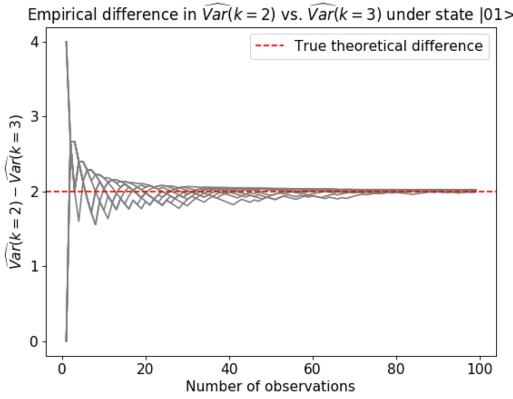


FIGURE 23. Convergence of the empirical difference, $\widehat{\text{Var}}(k = 2) - \widehat{\text{Var}}(k = 3)$, to the true difference in variances under $|01\rangle$. Since $\text{Var}(k = 2) - \text{Var}(k = 3)$ is positive, this signals that the $k = 3$ partitioning will lead to a lower variance estimator.

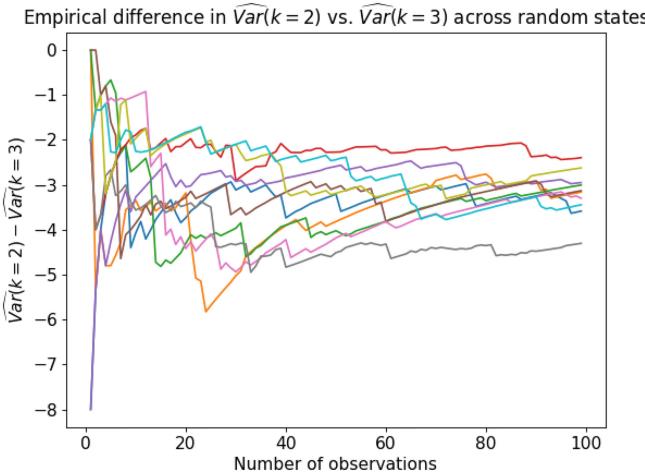


FIGURE 24. Empirical difference in $\widehat{\text{Var}}(k = 2) - \widehat{\text{Var}}(k = 3)$ across ten Haar-randomly-chosen states. While the convergence value differs across states, it is negative in all ten cases. This contrasts with the atypical case of convergence to a positive value in the example of Fig. 23 under state $|01\rangle$.

use observations from the $k = 2$ setting to explore whether further partitions would be beneficial.

Each of the gray lines in Fig. 23 depicts the value of $\widehat{\text{Var}}(k = 2) - \widehat{\text{Var}}(k = 3)$ as it evolves with a set of 100 observed measurements under the $|01\rangle$ state. The plot illustrates that the empirical difference, i.e., $\widehat{\text{Var}}(k = 2) - \widehat{\text{Var}}(k = 3)$, converges to the true theoretical difference, i.e., $\text{Var}(k = 2) - \text{Var}(k = 3) = 2$ after around 30 observations. The positive sign of this difference indicates that $\text{Var}(k = 3) < \text{Var}(k = 2)$, and therefore the $k = 3$ partitioning should be favored due to its lower variance.

When we broaden analysis of the $k = 2$ versus $k = 3$ setting across many different random states, we observe that the state $|01\rangle$ is indeed atypical and pathological, as suggested in Section XI-B. Under the vast majority of states, the variance of the $k = 2$ setting is lower than the $k = 3$ setting, as observed by the negative values of $\widehat{\text{Var}}(k = 2) - \widehat{\text{Var}}(k = 3)$ in

Fig. 24, and therefore the $-XX$ term should not be split into a separate partition.

This discussion naturally leads to the question of how many observations are necessary for the sample covariance matrix to be a good approximation of the true theoretical covariance matrix. To answer this question, we need to formalize a notion of the accuracy of a sample covariance matrix. Several candidate measures may be considered, which we are exploring in ongoing work.

- 1) Enforcing a minimum number of “burn-in” observations. This acts as a proxy of the sample observations being sufficiently representative of the true theoretical distribution.
- 2) Enforcing that the distance between the sample covariance matrix after $n - 1$ observations and after n observations be less than a prespecified threshold. This acts as an alternative proxy of the stability of the observations on which the sample covariance matrix is based.
- 3) Enforcing that a hypothesis test between the sample variance of the full partitioning and the sample variance of the split-up partitioning returns a p -value below a prespecified significance level.

The last candidate measure is the most attractive because p -values can be compared across different experimental settings. By contrast, appropriate cutoff values for the first two measures vary with H and $|\psi\rangle$. Formalization of the last measure will require further work to confirm the distribution of the sample variance and covariance terms.

XII. CONCLUSION

Our techniques and demonstrations show that simultaneous measurement substantially reduces the cost of VQE by allowing state preparations to cover several Pauli strings simultaneously. We demonstrate algorithms that achieve up to $30\times$ reductions in the number of requisite state preparations on specific molecules, and an $O(N)$ reduction asymptotically. This reduces the overall measurement cost of VQE on molecular Hamiltonians from $O(N^4)$ to $O(N^3)$. Our systems emphasis includes explicit attention to the overhead of the MIN-COMMUTING-PARTITION approximation runtime, as well as the overhead incurred by simultaneous measurement circuits. To address the latter, we develop a circuit synthesis procedure, which we have implemented and tested in software [77]. We also study the statistics of simultaneous measurement, and ensure that the top-level goal of finding MIN-COMMUTING-PARTITIONS is statistically justified. Our statistical analysis also yields a strategy for detecting and correcting course when simultaneous measurements are harmed by covariance terms. Our theoretical and benchmark/simulation results are accompanied by a proof-of-concept experimental validation on the IBM 20Q quantum computer.

ACKNOWLEDGMENT

The authors would like to thank K. Brown, P. Love, and W. Kirby for helpful discussions about VQE. They are also grateful to O. Di Matteo for introducing us to MUBs and clarifying the construction of simultaneous measurement circuits. Additionally, they would like to acknowledge H. Hoffmann for advising us on the time to solution for classical optimizers.

REFERENCES

- [1] J. Preskill, “Quantum computing in the NISQ era and beyond,” *Quantum*, vol. 2, 2018, Art. no. 79, doi: [10.22331/q-2018-08-06-79](https://doi.org/10.22331/q-2018-08-06-79).
- [2] P. W. Shor, “Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer,” *SIAM Rev.*, vol. 41, no. 2, pp. 303–332, 1999, doi: [10.1137/S0036149X95293172](https://doi.org/10.1137/S0036149X95293172).
- [3] L. K. Grover, “A fast quantum mechanical algorithm for database search,” *Proc. 28th Annu. ACM Symp. Theory Comput.*, 1996, pp. 212–219, doi: [10.1145/237814.237866](https://doi.org/10.1145/237814.237866).
- [4] A. Peruzzo *et al.*, “A variational eigenvalue solver on a photonic quantum processor,” *Nat. Commun.*, vol. 5, 2014, Art. no. 4213, doi: [10.1038/ncomms5213](https://doi.org/10.1038/ncomms5213).
- [5] G. Nannicini, “Performance of hybrid quantum-classical variational heuristics for combinatorial optimization,” *Phys. Rev. E*, vol. 99, no. 1, 2019, Art. no. 013304, doi: [10.1103/PhysRevE.99.013304](https://doi.org/10.1103/PhysRevE.99.013304).
- [6] N. Moll *et al.*, “Quantum optimization using variational algorithms on near-term quantum devices,” *Quantum Sci. Technol.*, vol. 3, no. 3, 2018, Art. no. 030503, doi: [10.1088/2058-9565/2Faab822](https://doi.org/10.1088/2058-9565/2Faab822).
- [7] A. Y. Kitaev, “Quantum measurements and the Abelian stabilizer problem,” 1995, *arXiv:quant-ph/9511026*.
- [8] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, “Quantum algorithms revisited,” in *Proc. Roy. Soc. London A, Math., Phys. Eng. Sci.*, vol. 454, no. 1969, pp. 339–354, 1998, doi: [10.1098/rspa.1998.0164](https://doi.org/10.1098/rspa.1998.0164).
- [9] T. S. Metodi, A. I. Faruque, and F. T. Chong, *Quantum Computing for Computer Architects* (Synthesis Lectures on Computer Architecture), vol. 6, no. 1. San Rafael, CA, USA: Morgan & Claypool, 2011, pp. 1–203, doi: [10.2200/S00331ED1V01Y201101CAC013](https://doi.org/10.2200/S00331ED1V01Y201101CAC013).
- [10] A. Matuschak and M. A. Nielsen, “Quantum computing for the very curious,” 2019. [Online]. Available: <https://quantum.country/qcvc>
- [11] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information (10th Anniversary Version)*. Cambridge U.K.: Cambridge Univ. Press, 2010, doi: [10.1119/1.463744](https://doi.org/10.1119/1.463744).
- [12] Microsoft, “Pauli measurements (# docs),” Dec. 2017. [Online]. Available: <https://docs.microsoft.com/en-us/quantum/concepts/pauli-measurements?view=qsharp-preview>.
- [13] R. Shankar, *Principles of Quantum Mechanics*. New York, NY, USA: Springer, 2012, doi: [10.1007/978-1-4757-0576-8](https://doi.org/10.1007/978-1-4757-0576-8).
- [14] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge U.K.: Cambridge Univ. Press, 2012, doi: [10.1017/CBO9780511810817](https://doi.org/10.1017/CBO9780511810817).
- [15] D. L. Strout and G. E. Scuseria, “A quantitative study of the scaling properties of the Hartree-Fock method,” *J. Chem. Phys.*, vol. 102, no. 21, pp. 8448–8452, 1995, doi: [10.1063/1.468836](https://doi.org/10.1063/1.468836).
- [16] G. Kresse and J. Furthmller, “Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set,” *Comput. Mater. Sci.*, vol. 6, no. 1, pp. 15–50, 1996, doi: [10.1016/0927-0256%2896%2900008-0](https://doi.org/10.1016/0927-0256%2896%2900008-0).
- [17] G. E. Scuseria and T. J. Lee, “Comparison of coupledcluster methods which include the effects of connected triple excitations,” *J. Chem. Phys.*, vol. 93, no. 8, pp. 5851–5855, 1990, doi: [10.1063/1.459684](https://doi.org/10.1063/1.459684).
- [18] C. D. Sherrill, “Computational scaling of the configuration interaction method with system size,” Quantum Chemistry Lecture Notes, School of Chem. and Biochem., Georgia Inst. of Technol., Atlanta, GA, USA, 1996. [Online]. Available: <http://vergil.chemistry.gatech.edu/notes/ciscale/ciscale.html>
- [19] K. Sugasaki, S. Nakazawa, K. Toyota, K. Sato, D. Shiomi, and T. Takui, “Quantum chemistry on quantum computers: A method for preparation of multiconfigurational wave functions on quantum computers without performing post-Hartree–Fock calculations,” *ACS Central Sci.*, vol. 5, no. 1, pp. 167–175, 2018, doi: [10.1021/acscentsci.8b00788](https://doi.org/10.1021/acscentsci.8b00788).
- [20] H. Eyring, “The activated complex in chemical reactions,” *J. Chem. Phys.*, vol. 3, no. 2, pp. 107–115, 1935, doi: [10.1063/1.1749604](https://doi.org/10.1063/1.1749604).
- [21] H. Paulsen and A. X. Trautwein, “Density functional theory calculations for spin crossover complexes,” in *Spin Crossover in Transition Metal Compounds III*. New York, NY, USA: Springer, 2004, pp. 197–219, doi: [10.1007/BF95428](https://doi.org/10.1007/BF95428).
- [22] S. McArdle, S. Endo, A. Aspuru-Guzik, S. Benjamin, and X. Yuan, “Quantum computational chemistry,” *Rev. Mod. Phys.*, vol. 92, 2020, Art. no. 015003, doi: [10.1103/RevModPhys.92.015003](https://doi.org/10.1103/RevModPhys.92.015003).
- [23] R. M. Parrish *et al.*, “Psi4 1.1: An open-source electronic structure program emphasizing automation, advanced libraries, and interoperability,” *J. Chem. Theory Comput.*, vol. 13, no. 7, pp. 3185–3197, 2017, doi: [10.1021/acs.jctc.7b00174](https://doi.org/10.1021/acs.jctc.7b00174).
- [24] Q. Sun *et al.*, “PYSCF: The python-based simulations of chemistry framework,” *Wiley Interdiscipl. Rev., Comput. Mol. Sci.*, vol. 8, no. 1, 2018, Art. no. e1340, doi: [10.1002/wcms.1340](https://doi.org/10.1002/wcms.1340).
- [25] J. R. McClean *et al.*, “OpenFermion: The electronic structure package for quantum computers,” *Quantum Sci. Technol.*, vol. 5, 2020, Art. no. 034014, doi: [10.1088/2058-9565%2Fab8ebc](https://doi.org/10.1088/2058-9565%2Fab8ebc).
- [26] D. Wecker, B. Bauer, B. K. Clark, M. B. Hastings, and M. Troyer, “Gate-count estimates for performing quantum chemistry on small quantum computers,” *Phys. Rev. A*, vol. 90, no. 2, 2014, Art. no. 022305, doi: [10.1103/PhysRevA.90.022305](https://doi.org/10.1103/PhysRevA.90.022305).
- [27] M. B. Hastings, D. Wecker, B. Bauer, and M. Troyer, “Improving quantum algorithms for quantum chemistry,” *Quantum Inform. Comput.*, vol. 15, no. 1/2, pp. 1–21, 2015.
- [28] P. Jordan and E. P. Wigner, “About the Pauli exclusion principle,” *Z. Phys.*, vol. 47, pp. 631–651, 1928, doi: [10.1007/BF01331938](https://doi.org/10.1007/BF01331938).
- [29] J. T. Seeley, M. J. Richard, and P. J. Love, “The Bravyi–Kitaev transformation for quantum computation of electronic structure,” *J. Chem. Phys.*, vol. 137, no. 22, 2012, Art. no. 224109, doi: [10.1063/1.4768229](https://doi.org/10.1063/1.4768229).
- [30] S. B. Bravyi and A. Y. Kitaev, “Fermionic quantum computation,” *Ann. Phys.*, vol. 298, no. 1, pp. 210–226, 2002, doi: [10.1006/aphy.2002.6254](https://doi.org/10.1006/aphy.2002.6254).
- [31] R. J. Bartlett and M. Musiał, “Coupled-cluster theory in quantum chemistry,” *Rev. Mod. Phys.*, vol. 79, no. 1, pp. 291–352, 2007, doi: [10.1103/REVMODPHYS.79.291](https://doi.org/10.1103/REVMODPHYS.79.291).
- [32] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, “Barren plateaus in quantum neural network training landscapes,” *Nat. Commun.*, vol. 9, no. 1, 2018, Art. no. 4812, doi: [10.1038/s41467-018-07090-4](https://doi.org/10.1038/s41467-018-07090-4).
- [33] A. J. McCaskey *et al.*, “Quantum chemistry as a benchmark for near-term quantum computers,” *npj Quantum Inform.*, vol. 5, 2019, Art. no. 99, doi: [10.1038/s41534-019-0209-0](https://doi.org/10.1038/s41534-019-0209-0).
- [34] C. Hempel *et al.*, “Quantum chemistry calculations on a trapped-ion quantum simulator,” *Phys. Rev. X*, vol. 8, no. 3, 2018, Art. no. 031022, doi: [10.1103/PhysRevX.8.031022](https://doi.org/10.1103/PhysRevX.8.031022).
- [35] J. Lee, W. J. Huggins, M. Head-Gordon, and K. B. Whaley, “Generalized unitary coupled cluster wave functions for quantum computation,” *J. Chem. Theory Comput.*, vol. 15, no. 1, pp. 311–324, 2018, doi: [10.1021/acs.jctc.8b01004](https://doi.org/10.1021/acs.jctc.8b01004).
- [36] J. Schwinger, “Unitary operator bases,” *Proc. Nat. Acad. Sci. USA*, vol. 46, no. 4, pp. 570–579, 1960, doi: [10.1073/PNAS.46.4.570](https://doi.org/10.1073/PNAS.46.4.570).
- [37] A. Klappenecker and M. Rötteler, “Constructions of mutually unbiased bases,” in *Proc. Int. Conf. Finite Fields Appl.*, 2003, pp. 137–144, doi: [10.1007/978-3-540-24633-6_10](https://doi.org/10.1007/978-3-540-24633-6_10).
- [38] A. Jena, S. Genin, and M. Mosca, “Pauli partitioning with respect to gate sets,” 2019, *arXiv:1907.07859*.
- [39] V. Verteletskyi, T.-C. Yen, and A. F. Izmaylov, “Measurement optimization in the variational quantum eigensolver using a minimum clique cover,” *J. Chem. Phys.*, vol. 152, 2020, Art. no. 124114, doi: [10.1063/1.5141458](https://doi.org/10.1063/1.5141458).
- [40] T.-C. Yen, V. Verteletskyi, and A. F. Izmaylov, “Measuring all compatible operators in one series of single-qubit measurements using unitary transformations,” *J. Chem. Theory Comput.*, vol. 16, pp. 2400–2409, 2020, doi: [10.1021/acs.jctc.9b00008](https://doi.org/10.1021/acs.jctc.9b00008).
- [41] A. F. Izmaylov, T.-C. Yen, R. A. Lang, and V. Verteletskyi, “Unitary partitioning approach to the measurement problem in the variational quantum eigensolver method,” *J. Chem. Theory Comput.*, vol. 16, pp. 190–195, 2020, doi: [10.1021/acs.jctc.9b00791](https://doi.org/10.1021/acs.jctc.9b00791).
- [42] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, “The theory of variational hybrid quantum-classical algorithms,” *New J. Phys.*, vol. 18, no. 2, 2016, Art. no. 023023, doi: [10.1088/1367-2630/18/2/023023](https://doi.org/10.1088/1367-2630/18/2/023023).

- [43] A. Kandala *et al.*, “Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets,” *Nature*, vol. 549, no. 7671, pp. 242–246, 2017, doi: [10.1038/nature23879](https://doi.org/10.1038/nature23879).
- [44] Y. Nam *et al.*, “Ground-state energy estimation of the water molecule on a trapped-ion quantum computer,” *npj Quantum Inform.*, vol. 6, 2020, Art. no. 33, doi: [10.1038/s41534-020-0259-3](https://doi.org/10.1038/s41534-020-0259-3).
- [45] C. Kokail *et al.*, “Self-verifying variational quantum simulation of the lattice Schwinger model,” *Nature*, vol. 569, 2019, Art. no. 355, doi: [10.1038/s41586-019-1177-4](https://doi.org/10.1038/s41586-019-1177-4).
- [46] R. S. Smith, M. J. Curtis, and W. J. Zeng, “A practical quantum instruction set architecture,” 2016, *arXiv:1608.03355*.
- [47] S. Bravyi, J. M. Gambetta, A. Mezzacapo, and K. Temme, “Tapering off qubits to simulate fermionic Hamiltonians,” 2017, *arXiv:1701.08213*.
- [48] A. F. Izmaylov, T.-C. Yen, and I. G. Ryabinkin, “Revising the measurement process in the variational quantum eigensolver: Is it possible to reduce the number of separately measured operators?,” *Chem. Sci.*, vol. 10, 2019, Art. no. 3746, doi: [10.1039/c8sc05592k](https://doi.org/10.1039/c8sc05592k).
- [49] B. Moseley, M. Osborne, and S. Benjamin, “Bayesian optimisation for variational quantum eigensolvers,” *Quantum*, vol. 3, pp. 4–5, 2018.
- [50] Y. Shi *et al.*, “Optimized compilation of aggregated instructions for realistic quantum computers,” in *Proc. 24th Int. Conf. Archit. Support Program. Lang. Oper. Syst.*, 2019, pp. 1031–1044, doi: [10.1145/3297858.3304018](https://doi.org/10.1145/3297858.3304018).
- [51] P. Gokhale *et al.*, “Partial compilation of variational algorithms for noisy intermediate-scale quantum machines,” in *Proc. 52nd Annu. IEEE/ACM Int. Symp. Microarchit.*, 2019, pp. 266–278, doi: [10.1145/3352460.3358313](https://doi.org/10.1145/3352460.3358313).
- [52] P.-L. Dallaire-Demers, J. Romero, L. Veis, S. Sim, and A. Aspuru-Guzik, “Low-depth circuit ansatz for preparing correlated fermionic states on a quantum computer,” *Quantum Sci. Technol.*, vol. 4, 2019, Art. no. 045005, doi: [10.1088/2058-9565/2FAB3951](https://doi.org/10.1088/2058-9565/2FAB3951).
- [53] M. Otten and S. K. Gray, “Accounting for errors in quantum algorithms via individual error reduction,” *npj Quantum Inform.*, vol. 5, no. 1, 2019, Art. no. 11, doi: [10.1038/s41534-019-0125-3](https://doi.org/10.1038/s41534-019-0125-3).
- [54] W. M. Kirby and P. J. Love, “Contextuality test of the in the variational quantum eigensolver,” *Phys. Rev. Lett.*, vol. 123, 2019, Art. no. 200501.
- [55] R. M. Karp, “Reducibility among combinatorial problems,” in *Complexity of Computer Computations*. New York, NY, USA: Springer, 1972, pp. 85–103, doi: [10.1007/978-3-540-68279-0_8](https://doi.org/10.1007/978-3-540-68279-0_8).
- [56] D. Zuckerman, “Linear degree extractors and the inapproximability of max clique and chromatic number,” in *Proc. 38th Annu. ACM Symp. Theory Comput.*, 2006, pp. 681–690, doi: [10.1145/1132516.1132612](https://doi.org/10.1145/1132516.1132612).
- [57] M. Planat and M. Saniga, “On the Pauli graphs of N-qudits,” 2017, *arXiv:quant-ph/0701211*.
- [58] R. Boppana and M. M. Halldórsson, “Approximating maximum independent sets by excluding subgraphs,” *BIT Numer. Math.*, vol. 32, no. 2, pp. 180–196, 1992, doi: [10.1007/BF035946-6_74](https://doi.org/10.1007/BF035946-6_74).
- [59] A. Hagberg, P. Swart, and D. S Chult, “Exploring network structure, dynamics, and function using networkX,” in *Proc. 7th Python Sci. Conf.*, 2008, pp. 11–15.
- [60] C. Bron and J. Kerbosch, “Algorithm 457: Finding all cliques of an undirected graph,” *Commun. ACM*, vol. 16, no. 9, pp. 575–577, 1973, doi: [10.1145/362342.362367](https://doi.org/10.1145/362342.362367).
- [61] E. Jones *et al.*, “SciPy: Open source scientific tools for Python,” 2001, [Online]. Available: <http://www.scipy.org/>
- [62] J. D. Whitfield, J. Biamonte, and A. Aspuru-Guzik, “Simulation of electronic structure Hamiltonians using quantum computers,” *Mol. Phys.*, vol. 109, no. 5, pp. 735–750, 2011, doi: [10.1080/00268976.2011.552441](https://doi.org/10.1080/00268976.2011.552441).
- [63] Z. Baranyai, “On the factorization of the complete uniform hypergraphs,” in *Infinite and Finite Sets*, A. Hanaj, R. Rado, and V. T. Sós, Eds., *Colloq. Math. Soc. Janos Bolyai*, 1974, pp. 91–108.
- [64] R. F. Bailey and B. Stevens, “Hamiltonian decompositions of complete k-uniform hypergraphs,” *Discr. Math.*, vol. 310, no. 22, pp. 3088–3095, 2010, doi: [10.1016/j.disc.2009.03.047](https://doi.org/10.1016/j.disc.2009.03.047).
- [65] U. Tamm, “Applications of Baranyai’s theorem in information theory.”
- [66] N. Deo and P. Mickevicius, “On one-factorization of complete 3-uniform hypergraphs,” *Congressus Numerantium*, vol. 158, pp. 153–162, 2002.
- [67] A. Brouwer and A. Schrijver, “Uniform hypergraphs,” in *Packing and Covering in Combinatorics*, A. Schrijver, Ed. Amsterdam, The Netherlands: Stichting Mathematisch Centrum, 1979, pp. 39–93.
- [68] E. W. Weisstein, “Baranyai’s theorem. From MathWorld—A Wolfram Web Resource.”
- [69] P. Bartlett, “Lecture 2: The max-flow min-cut theorem,” Apr. 2015.
- [70] D. Eisenstat, “Find unique compositions of k-distinct-element subsets for a set of n elements,” Stack Overflow (version: 2019-08-06). [Online]. Available: <https://stackoverflow.com/a/25902878>
- [71] L. Ford and D. Fulkerson, “Maximal flow through a network,” in *Classic Papers in Combinatorics*. New York, NY, USA: Springer, 1954, pp. 243–248, doi: [10.1007/978-0-8176-4842-8](https://doi.org/10.1007/978-0-8176-4842-8).
- [72] E. Cohen, “Approximate max-flow on small depth networks,” *SIAM J. Comput.*, vol. 24, no. 3, pp. 579–597, 1995, doi: [10.1137/S0097539792236717](https://doi.org/10.1137/S0097539792236717).
- [73] A. Madry, “Navigating central path with electrical flows: From flows to matchings, and back,” in *Proc. IEEE 54th Annu. Symp. Foundations Comput. Sci.*, 2013, pp. 253–262, doi: [10.1109/FOCS.2013.35](https://doi.org/10.1109/FOCS.2013.35).
- [74] Y. T. Lee, S. Rao, and N. Srivastava, “A new approach to computing maximum flows using electrical flows,” in *Proc. 40th Annu. ACM Symp. Theory Comput.*, 2013, pp. 755–764, doi: [10.1145/2488608.2488704](https://doi.org/10.1145/2488608.2488704).
- [75] D. Kang and J. Payor, “Flow rounding,” 2015, *arXiv:1507.08139*.
- [76] K. Gui *et al.*, “Term grouping and travelling salesperson for digital quantum simulation,” 2020, *arXiv:2001.05983*.
- [77] “Github repository: VQE-term-grouping,” 2020. [Online]. Available: <https://github.com/teaguetomesh/vqe-term-grouping/>
- [78] C. M. Dawson and M. A. Nielsen, “The Solovay-Kitaev algorithm,” 2005, *arXiv:quant-ph/0505030*.
- [79] N. Khaneja and S. Glaser, “Cartan decomposition of SU(2”), constructive controllability of spin systems and universal quantum computing,” *Chem. Phys.*, vol. 267, no. 13, pp. 11–23, Jun. 2001, doi: [10.1016/S0301-0104/2801%2900318-4](https://doi.org/10.1016/S0301-0104/2801%2900318-4).
- [80] N. J. Ross, “Optimal ancilla-free Pauli+ v approximation of z-rotations,” 2014, *arXiv:1409.4355*.
- [81] C.-K. Li, R. Roberts, and X. Yin, “Decomposition of unitary matrices and quantum gates,” *Int. J. Quantum Inform.*, vol. 11, no. 1, 2013, Art. no. 1350015, doi: [10.1142/S0219749913500159](https://doi.org/10.1142/S0219749913500159).
- [82] A. Daskin and S. Kais, “Decomposition of unitary matrices for finding quantum circuits: application to molecular Hamiltonians,” *J. Chem. Phys.*, vol. 134, no. 14, 2011, Art. no. 144112, doi: [10.1063/1.3575402](https://doi.org/10.1063/1.3575402).
- [83] Y. Nakajima, Y. Kawano, and H. Sekigawa, “A new algorithm for producing quantum circuits using KAK decompositions,” 2005, *arXiv:quant-ph/0509196*.
- [84] D. Gottesman, “Stabilizer codes and quantum error correction,” Ph.D. dissertation, Phys., Math. Astron., California Inst. Technol., Pasadena, CA, USA, 1997, doi: [10.7907/RZR7-DT72](https://doi.org/10.7907/RZR7-DT72).
- [85] S. Aaronson and D. Gottesman, “Improved simulation of stabilizer circuits,” *Phys. Rev. A*, vol. 70, no. 5, 2004, Art. no. 052328, doi: [10.1103/PhysRevA.70.052328](https://doi.org/10.1103/PhysRevA.70.052328).
- [86] U. Seyfarth and K. S. Ranade, “Construction of mutually unbiased bases with cyclic symmetry for qubit systems,” *Phys. Rev. A*, vol. 84, no. 4, 2011, Art. no. 042327, doi: [10.1103/PhysRevA.84.042327](https://doi.org/10.1103/PhysRevA.84.042327).
- [87] U. Seyfarth, “Cyclic mutually unbiased bases and quantum public-key encryption,” 2019, *arXiv:1907.02726*.
- [88] Y.-C. Zheng, C.-Y. Lai, T. A. Brun, and L.-C. Kwek, “Depth reduction for quantum Clifford circuits through Pauli measurements,” 2018, *arXiv:1805.12082*.
- [89] R. W. Farebrother, *Linear Least Squares Computations*. New York, NY, USA: Marcel Dekker, 1988, doi: [10.1201/9780203748923](https://doi.org/10.1201/9780203748923).
- [90] P. K. Barkoutsos *et al.*, “Quantum algorithms for electronic structure calculations: Particle-hole Hamiltonian and optimized wave-function expansions,” *Phys. Rev. A*, vol. 98, no. 2, 2018, Art. no. 022322, doi: [10.1103/PhysRevA.98.022322](https://doi.org/10.1103/PhysRevA.98.022322).
- [91] E. F. Dumitrescu *et al.*, “Cloud quantum computing of an atomic nucleus,” *Phys. Rev. Lett.*, vol. 120, no. 21, 2018, Art. no. 210501, doi: [10.1103/PhysRevLett.120.210501](https://doi.org/10.1103/PhysRevLett.120.210501).
- [92] “Quantum devices & simulators,” 2018, Accessed: Jul. 30, 2019, [Online]. Available: <https://www.research.ibm.com/ibm-q/technology/devices/#ibmq-20-tokyo>
- [93] K. Życzkowski, P. Horodecki, A. Sanpera, and M. Lewenstein, “Volume of the set of separable states,” *Phys. Rev. A*, vol. 58, no. 2, 1998, Art. no. 883, doi: [10.1103/PHYSREVA.58.883](https://doi.org/10.1103/PHYSREVA.58.883).
- [94] N. J. Russell, L. Chakhmakhchyan, J. L. OBrien, and A. Laing, “Direct dialling of haar random unitary matrices,” *New J. Phys.*, vol. 19, no. 3, 2017, Art. no. 033007, doi: [10.1088/1367-2630%2Faaf60ed](https://doi.org/10.1088/1367-2630%2Faaf60ed).