Optimization of Simultaneous Measurement for Variational Quantum Eigensolver Applications

Pranav Gokhale∗, Olivia Angiuli†, Yongshan Ding∗, Kaïwen Gui∗, Teague Tomesh‡§, Martin Suchara∗§, Margaret Martonosi‡, and Frederic T. Chong∗
∗ University of Chicago, † University of California, Berkeley, ‡ Princeton University, § Argonne National Laboratory
Corresponding author email: pranav@super.tech

Abstract—Variational quantum eigensolver (VQE) is a promising algorithm suitable for near-term quantum computers. VQE aims to approximate solutions to exponentially-sized optimization problems by executing a polynomial number of quantum subproblems. However, the number of subproblems scales as $N^4$ for typical problems of interest—a daunting growth rate that poses a serious limitation for emerging applications such as quantum computational chemistry.

We mitigate this issue by exploiting the simultaneous measurability of subproblems corresponding to commuting terms. Our technique transpiles VQE instances into a format optimized for simultaneous measurement, ultimately yielding 8-30x lower cost. Our work also encompasses a synthesis tool for compiling simultaneous measurement circuits with minimal overhead. We demonstrate experimental validation of our techniques by estimating the ground state energy of deuteron with a quantum computer. We also investigate the underlying statistics of simultaneous measurement and devise an adaptive strategy for mitigating harmful covariance terms.

Keywords—quantum computing, Variational Quantum Eigensolver (VQE)

I. INTRODUCTION

The present Noisy Intermediate-Scale Quantum (NISQ) era [44] is distinguished by the advent of quantum computers comprising tens of qubits, with hundreds of qubits expected in the next five years. Although millions of physical qubits are needed to realize the originally-envisioned quantum applications such as factoring [51] and database search [15], a new generation of variational algorithms have been recently introduced to match the constraints of NISQ hardware. However, these variational algorithms are very expensive in terms of the number of subproblems, or measurements, needed.

Variational Quantum Eigensolver (VQE) [42] is one such algorithm that is widely considered a top contender, if not the top contender, for demonstrating a useful quantum speedup. VQE is used to approximate solutions to exponentially-sized optimization problems in polynomial time. It has a wide class of applications such as molecular ground state estimation [42]; maximum 3-sat, market split, traveling salesperson [39]; and maximum cut [36]. In this paper, we focus on the molecular ground state estimation problem though we underscore that the full range of VQE applications is very broad.

VQE solves a similar problem as Quantum Phase Estimation (QPE) [27], [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 co-processor. As a result, VQE circuits have low

This work is funded in part by EPiQC, an NSF Expedition in Computing, under grants CCF-1730449/1730082, and in part by STAQ, under grant NSF Phy-1818914. P. G. is supported by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program. O. A. is supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE 1752814. The work of K. G. and M. S. is supported by the U.S. Department of Energy, Office of Science, under contract number DE-AC02-06CH11357.

Fig. 1. Estimating the energy of $H_2$ naively requires 14 separate measurements (left). Our technique (right) transpiles the VQE instance to only require 2 simultaneous measurements. We can act either at the qubit representation, or in linear time at a higher abstraction (molecular Hamiltonian) level. Measurement circuit synthesis allows us to perform the necessary simultaneous measurements and classically post-process to obtain all 14 targeted measurements. Covariance mitigation prevents harmful covariances within measurement groups.
gate counts and are error resilient, but at the cost of splitting computation into \(O(N^4)\) subproblems, each performing measurement of a single term known as a Pauli string. In other words, VQE exchanges having too many gates with having too many measurements—this poses practical limitations.

It was observed that the \(N^4\) scaling could be partly mitigated by performing simultaneous measurement [33]: when the measurement terms for two subproblems commute, they can be measured simultaneously. Our work, depicted in the Figure 1 overview, exploits this property in order to minimize the number of measurements needed for VQE.

Our specific contributions include:

- Techniques for transpiling Na"ive VQE instances to ones partitioned into commuting families. Section [VI] operates at the qubit representation layer; Section [V] operates at a higher molecular abstraction layer, ensuring that transpilation cost is not a bottleneck.
- A circuit synthesis tool for simultaneous measurement (Section [VI].
- Validation of our techniques through benchmark simulation (Section [VII]) and experiment (Section [VIII].
- Statistical analysis of simultaneous measurement, including a technique that resolves a previously-open question regarding the impact of covariances (Section [IX].

We begin in Section [II] by presenting relevant background, followed by an analysis of commutativity in Section [III].

II. BACKGROUND

A. Quantum Measurement

Quantum bits (qubits) are differentiated from classical bits by their ability to exist in superposition. Rather than being either \(|0\rangle\) or \(|1\rangle\), the internal state of a qubit can be a coherent superposition of both. This superposition is ‘analog’ and is represented as a point on the surface of the Bloch sphere shown in Figure 2 along with \(|0\rangle\) on the north pole and \(|1\rangle\) on the south pole. Although qubits can have a superposition internally, when they are measured in hardware, they collapse to either \(|0\rangle\) or \(|1\rangle\), with a probability dependent on the state’s latitude. This measurement/readout behavior is indicated by the Standard Z Measurement label in Figure 2.

While the Z-axis or computational basis measurement is the only measurement that hardware performs, we can effectively measure a qubit on any other axis, such as the X or Y axes. Mathematically, measurement of an operator means that the qubit’s state collapses to an eigenvector of that operator. For VQE, the relevant operators are defined by the Pauli matrices: \(X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\), \(Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}\), and \(Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}\).

Visually, the eigenvectors of \(X\) (\(Y\)) are the antipodal points along the \(X\) (\(Y\)) -axis of the Bloch sphere. Since hardware cannot directly measure along these axes, measurements of \(X\) (\(Y\)) are performed by first rotating the Bloch sphere with a quantum gate so that the \(X\) (\(Y\)) -axis becomes aligned with the Z-axis, as shown in Figure 3. Subsequently, a standard Z-basis measurement can be performed, whose outcome can then be mapped to an effective \(X\) (\(Y\)) measurement. The quantum gates that implement the X-to-Z and Y-to-Z axis rotations are called \(H\) and \(HS^{-1}\) respectively [35]. Written as quantum circuits (which are ‘timeline’ views from left to right), these rotations look like \(-H\) and \(-S^{-1}H\). The same general measurement principle applies towards measuring operators across multiple qubits: measurement is accomplished by rotating the eigenvectors of the target operator to align with the standard Z-basis vectors. Thereafter, subsequent Z-basis measurement collapses the qubit state onto an eigenvector of the target operator, as desired. The quantum circuit for the necessary eigenvector rotation has a matrix representation whose columns are the eigenvectors of the target operator. In this work, we are interested in measuring Pauli strings, which are concatenated Pauli matrices across multiple qubits—for example, \(X_3I_2Z_1Y_0\), often abbreviated without subscripts as \(XIZY\).

B. Simultaneous Measurement and Commutativity

Per the preceding discussion, two operators can be measured simultaneously if they share a full set of eigenvectors. In such a case, they can be measured simultaneously by applying the quantum circuit that rotates their shared eigenvectors onto the Z-basis. In the case of Pauli strings, two operators share a set of eigenvectors if and only if they commute [50, Chapter 1], i.e. the order of their product is interchangeable. Thus, a family of operators can be measured simultaneously if they pairwise commute [14, Theorem 1.3.21].

This result is foundational in quantum mechanics. For instance, the Heisenberg Uncertainty Principle states that
position and momentum cannot be known simultaneously—this is because the position and momentum operators do not commute. In this paper, we will exploit commutativity relationships to measure many Pauli strings with a single measurement. Informally, we aim to get “many birds with one stone.” Notice that we have not yet discussed how to actually perform the necessary eigenvector rotations, or how to translate one simultaneous measurement into the actual measurement outcomes for the multiple Pauli strings. Both of these questions are addressed in Section VI, with a specific example in VI-B.

C. Variational Quantum Eigensolver (VQE)

VQE can be applied to a wide class of optimization problems that are solvable as minimum-eigenvalue estimation [39], [36]. In this paper, we focus on the application that has received the most commercial and experimental interest: estimating molecular ground state energy. We use VQE to approximate the lowest eigenvalue of an exponentially-sized matrix called the Hamiltonian that captures the molecule’s energy configuration. The lowest eigenvalue of the Hamiltonian is the molecule’s ground state energy which has important implications in chemistry such as determining reaction rates [12] and molecular geometry [41].

The Hamiltonian matrix, \( H \), can be used as an operator to calculate the energy of any given quantum state. Specifically, for an input state \( |\psi\rangle \), the expected value \( \langle H \rangle_{|\psi\rangle} \) equals the energy of that state. Thus, our goal is to find the input state that yields the lowest possible \( \langle H \rangle_{|\psi\rangle} \), which is the ground state energy. On a classical computer, measuring \( \langle H \rangle \) would require multiplication of exponentially-sized matrices. The potential quantum speedup of VQE arises from its approach of measuring \( \langle H \rangle_{|\psi\rangle} \) indirectly but efficiently by decomposing into \( O(N^4) \) subproblems. In particular, VQE employs linearity of expectation to decompose \( \langle H \rangle_{|\psi\rangle} \) into a sum of \( O(N^4) \) expectations of Pauli strings, which can each be computed efficiently. In the standard formulation of VQE, each of these Pauli strings is measured via a separate measurement [42].

At its core, VQE can be described as a guess-check-repeat algorithm. Initially, the algorithm guesses the ground state of the Hamiltonian. Then, it checks the actual energy for the guessed state by summing expected values over the \( O(N^4) \) directly measurable Pauli strings, as described above. Finally, it repeats by trying a new guess for the ground state, with the assistance of a classical optimizer that guides the next guess based on past results. Repetition continues until the classical optimizer reaches its termination condition—usually when a good minimum has been found. Algorithm [1] presents the pseudocode for VQE, under the standard ‘Naïve’ formulation where each Pauli string is measured separately.

Since the number of possible state vectors spans the exponentially large and continuous ‘Hilbert space’ of quantum states, VQE operates with a restricted family of candidate ground states. Such a restricted family is called an ansatz, and the ansatz state \( |\psi(\vec{\theta})\rangle \) is parametrized by a vector \( \vec{\theta} \). While our work in this paper is applicable to any ansatz, we focus our attention to the Unitary Coupled Cluster Single Double (UCCSD) ansatz, which is the leading contender for molecular ground state estimation. In terms of the number of qubits \( N \), the total gate count of the UCCSD circuit is \( O(N^4) \) [17], [29].

III. ANALYSIS OF COMMUTATIVITY

We analyze the commutativity of Pauli strings (concatenations of the four Pauli matrices), which are the terms present in Hamiltonian decompositions.

A. Single Qubit Case

For a single qubit, the four Pauli strings of length \( N = 1 \) are simply the four Pauli matrices. Figure 4 depicts the commutation graph for the four matrices, with edges between commuting pairs. \( I \) (Identity) commutes with everything else, and all matrices commute with themselves, as indicated by the self-loops.

![Fig. 4. Commutation graph for the four Pauli matrices.](image)

B. Qubit-Wise Commutativity (QWC)

Now we consider the general case of \( N \)-qubit Pauli strings. The simplest type of commutativity is Qubit-Wise 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 satisfy commutation.

QWC has been leveraged in past experimental work for small molecules [24], [38], [17], [28] by ad hoc inspection.

---

**Algorithm 1:** Variational Quantum Eigensolver (VQE)

Result: Approximate ground state energy, 
\[
\min_{\vec{\theta}} \langle H \rangle_{\psi(\vec{\theta})}
\]

\[
i \leftarrow 1;
\]

\[
\vec{\theta}_i \leftarrow \text{random angles};
\]

while (not classical optimizer termination condition) do

\[
\text{for } j \in [O(N^4) \text{ Pauli terms}] \text{ do}
\]

\[
\text{Prepare } \psi(\vec{\theta}_i);
\]

\[
\text{Measure } \langle H_j \rangle \text{ under state } \psi(\vec{\theta}_i);
\]

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

\[
\text{Record } (\theta, \langle H \rangle_{\psi(\vec{\theta}_i)});
\]

\[
i++;
\]

Pick new \( \vec{\theta}_i \) via classical optimizer;

end

---

of the Hamiltonian terms; however, these techniques do not scale.

C. General Commutativity (GC)

QWC captures only a small subset of possible commutativity relationships. The most General Commutativity (GC) rule is that two Pauli strings commute if and only if they do not commute on an even number of indices. For example, \{XX, YY, ZZ\} is a GC family: for any pair of strings, two (both) indices fail to commute.

Figure 5 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 (e.g. between XI, IX, and XX) and a red edge indicates that the pair is GC but not QWC (e.g. between XX, YY, and ZZ).

![Fig. 5. Commutation graph for all 16 2-qubit Pauli strings. Blue edges indicate Pauli strings that commute under QWC (which is a subset of GC). Red edges commute under GC-but-not-QWC.](image)

IV. TRANSPILATION VIA MIN-CLIQUE-COVER

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. The list of partitions amounts to a transpiled instance of VQE, indicating that the Pauli strings in each partition should be measured simultaneously. The ‘Transpilation’ box on the right side of Figure 1 depicts our goal.

Instead of solving MIN-COMMUTING-PARTITION exactly, we first map it to a graph problem, as suggestively expressed by the graph representation in Figure 5 and then use approximation methods to obtain a solution. 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 simultaneously. Therefore, we seek the MIN-CLIQUE-COVER, i.e. the smallest possible set of cliques whose union spans all vertices. As an example, Figure 6 shows the commutation graph for H₂’s 4-qubit Hamiltonian and its MIN-CLIQUE-COVERs using QWC edges and using GC edges.

MIN-CLIQUE-COVER is NP-Hard [25] and approximating a guaranteed ‘good’ clique cover is also NP-Hard for general graphs [55]. However, molecular Hamiltonian graphs are highly structured owing both to features of the Pauli commutation graph [43] and to patterns in the Pauli strings that arise in molecular Hamiltonians (we explicitly address and exploit the latter in Section V). This suggests that MIN-CLIQUE-COVER approximation algorithms may still yield reasonably good results.

A. Approximation Algorithms Tested

In our benchmarking, we performed MIN-CLIQUE-COVERs using the Boppana-Halldórsson algorithm [6] included in the NetworkX Python package [16], as well as the Bron-Kerbosch algorithm [7] 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 [33]—this approximation is a non-graph-based randomized algorithm that only finds QWC partitions. Section VII presents results across a range of molecules and Hamiltonian sizes. All of our code and results notebooks are available online [2].

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 lists all maximal cliques with a worst case exponential runtime. Therefore, it should be interpreted as a soft upper bound on how well polynomial-time approximation algorithms can approximate a MIN-CLIQUE-COVER.
Boppana-Halldórsson algorithm’s runtime is polynomial but is not well studied. Our benchmarks and theoretical analysis indicate roughly quadratic scaling in the number of vertices. Some polynomial benchmarks considered in other work [20], [21], [54], [52] scale 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 transpilation/precomputation time. Beyond simply implying impractical scaling rates, these runtimes may exceed the runtime of Naïve VQE—in which case, we’d be better off just running VQE in the Naïve fashion! The exact runtime of VQE is dependent on the classical optimizer and optimization landscape, but estimates range from \( O(N^8) \) under Scipy optimization settings [22] 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 the VQE transpilation cost from expensive MIN-CLIQUE-COVER approximations would exceed the naïve Pauli measurement cost.

In the next section, we remedy this concern by presenting a linear-time transpilation technique that exploits our knowledge of the structure of molecular Hamiltonians and their encodings into qubits. The resulting transpilation procedure runs in \( O(N^4) \) time (linear in the number of Pauli strings), which is safely below the quantum invocation cost of VQE.

V. LINEAR-TIME TRANSPILATION

As discussed in the previous section, the VQE transpilation methods based on MIN-CLIQUE-COVER approximation may have classical costs greatly exceeding the quantum cost of running VQE naïvely. This motivates us to find faster transpilation procedures by exploiting the structure of molecular Hamiltonians, which is ignored by the MIN-CLIQUE-COVER approximations.

Thus, we now address the simultaneous measurement goal at a higher abstraction level. This is depicted by the arrow from ‘Molecular Model (Hamiltonian)’ to ‘Simultaneous Measurement’ in Figure 1. Previously, we worked with Hamiltonians specified as a sum over \( O(N^4) \) Pauli strings. However, this Pauli string summation is derived from the ‘true’ molecular form of \( H \), which is expressed in fermionic language as:

\[
H = \sum_{p}^{N} \sum_{q}^{N} h_{pq} a_{p}^{\dagger} a_{q} + \sum_{p}^{N} \sum_{q}^{N} \sum_{r}^{N} \sum_{s}^{N} h_{pqrs} a_{p}^{\dagger} a_{q}^{\dagger} a_{r} a_{s} \tag{1}
\]

\( a \) and \( a^{\dagger} \) are fermionic operators that are converted to Pauli strings via a fermion-to-qubit encoding. We focus on Jordan-Wigner [23], which is the most popular encoding [19], though our results also apply to the Parity encoding [47]. Notice that the \( N^4 \) scaling of the number of terms in the Hamiltonian is clear from the quadruple summation in Equations (1). These terms are asymptotically dominant [53]. At the scale of smaller molecules, the \( O(N) \) terms of form \( pp \) and the \( O(N^2) \) terms of form \( pqpq \) are frequent. We treat both the asymptotically-dominant terms and the frequent-for-small-molecules terms.

A. \( pqrs \) Terms

For each of the asymptotically dominant \( N^4 \) terms of form \( a_{p}^{\dagger} a_{q}^{\dagger} a_{r} a_{s} \), the Jordan-Wigner encoding yields a sum over the 16 Pauli strings matching the regular expression:

\[
(X_p|Y_p)Z_{p-q}(X_q|Y_q)(X_r|Y_r)Z_{r-s}(X_s|Y_s)
\]

where \( Z_{p-q} \) denotes repeating \( Z \)’s from indices \( p \) to \( q \), excluding endpoints.

Thus, the \( \sim 1N^4 \) fermionic terms expand into \( \sim 16N^4 \) Pauli strings. Our key observation is that within each set of 16 Pauli strings, the MIN-CLIQUE-COVER is just 2. Figure 7 illustrates this, showing the 16 Pauli strings matching the regular expression above (it can be shown that the \( Z \)’s don’t affect commutativity since they apply to the same indices). Using GC partitions, we can perform all 16 measurements using only two measurement circuits. Thus, we reduce the measurement cost from \( \sim 16N^4 \) to \( \sim 2N^4 \). Moreover, this transpilation is fast (linear time in number of terms), occurring directly alongside the fermion-to-qubit encoding stage.

B. \( pp \) and \( ppqq \) Terms

While the 8-fold reduction via partitioning the \( pqrs \) terms is the asymptotic bottleneck, we also note a useful reduction for the smaller terms which are significant for smaller molecules. For the \( O(N) \) operators of form \( a_{p}^{\dagger} a_{p} \), multiplying out the Jordan-Wigner encoding yields the Pauli string \( Z_p \). For the \( O(N^2) \) operators of form \( a_{p}^{\dagger} a_{q}^{\dagger} a_{p} a_{q} \), the Jordan-Wigner encoding yields the Pauli string \( Z_p Z_q \). All of these Pauli strings commute and therefore can be simultaneously measured. We use this technique to yield reduced measurement costs even for small molecules.

VI. MEASUREMENT CIRCUIT SYNTHESIS

Once a MIN-COMMUTING-PARTITION has been approximated, we have effectively transpiled the VQE instance into one in which terms within a partition should be measured simultaneously. Naturally, the question arises of how to actually perform the necessary simultaneous measurements. In the case of Naïve partitions where each Pauli string is measured separately, the measurement circuit is trivial. In particular, recall from Section 1 that we simply perform the \( H \) and
$HS^{-1}$ 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.

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 $X_{II}Y_{IZ}$ and $I_{XI}Y_{IZ}$. We simply measure in the $X$ basis on the left two qubits, the $Y$ basis on the middle two qubits, and the $Z$ basis on the right two qubits. In terms of circuit cost, QWC measurement is essentially equivalent to the 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. As depicted in the ‘Measurement Circuit Synthesis’ box in Figure 1, we provide a procedure that enables these measurements, and we analyze its classical and quantum costs. 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 and validated it across a wide range of molecular Hamiltonians.

A. Background

As discussed in Section 11 performing a simultaneous measurement amounts to rotating the axes of the shared eigenvectors to align with the standard $Z$-basis axes. One attempt to construct such a rotation would be to explicitly compute the shared eigenvectors and then feed them to one of many possible gate decomposition techniques [10], [26], [45], [50], [10]. However, this approach is problematic for two reasons. First, decomposition techniques trade off 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, which are each represented by a $2^N$-sized column vector. The exponential effort of computing these eigenvectors would erase any potential quantum advantage.

Fortunately, the stabilizer formalism—typically applied to quantum error correction—provides us an alternative. Our work is built upon the language of stabilizers introduced in [14] and expanded upon in [3] as well as [49], [48]. While these two papers were applied to error correction, quantum simulation, and Mutually Unbiased Bases, the core techniques also apply to our use case. Our circuit construction procedure is inspired by these papers, but stems from a different context and end goal.

B. An 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. This task is equivalent to another task in quantum computing that has a well-known solution: Bell basis measurement. Figure 8 presents the circuit for such a measurement. We now explore why this circuit performs simultaneous measurement of $XX$, $YY$, and $ZZ$.

First observe that since $YY = -(XX)(ZZ)$, it is sufficient to target a measurement of $[XX, ZZ]$. Our goal is to transform this target measurement into $[ZI, IZ]$, which captures the top- and bottom-qubit outcomes we measure directly via standard $Z$-basis measurement, as discussed in Section 11. An important background fact is that after applying a quantum gate $U$, a target measurement of $M$ on the original state has become equivalent to a measurement of $UMU^{-1}$ [14], [40] on the new state. This is known as unitary conjugation.

In the Bell basis measurement circuit, we first apply a gate called $U = CNOT$. By computing $UMU^{-1}$ we can see that target measurements of $[XX, ZZ]$ are transformed under conjugation to measurements of $[XX, ZZ] \xrightarrow{UMU^{-1}} [UXXU^{-1}, UZZU^{-1}] = [XI, IZ]$.

Finally, after applying the ‘Hadamard’ gate on the top qubit $(U = H \otimes I)$, the measurements are transformed to $[XI, IZ] \xrightarrow{UMU^{-1}} [UXIU^{-1}, U1IZU^{-1}] = [ZI, IZ]$.

Thus, this $CNOT$, $H \otimes I$ gate sequence performs the desired transformation of rotating a measurement of $[XX, ZZ]$ into the $Z$-basis, $[ZI, IZ]$. 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 $-(XX)(ZZ)$. Thus, after applying the Bell basis measurement circuit and reading out the results, we simultaneously know the outcomes of $XX$, $YY$, and $ZZ$.

C. Circuit Synthesis Procedure

Algorithm 2 describes our circuit synthesis procedure for the general case. For brevity, we omit technical details about the construction of stabilizer matrices, $Z$-matrices, and $X$-matrices. However, we refer readers to [14], [3], [48] for technical background. Moreover, our software repository includes generate_measurement_circuit.py, an implementation of Algorithm 2 with detailed comments along the way.

D. Circuit Synthesis Complexity and Circuit Cost

The efficiency of Algorithm 2 stems from its use of linearly sized stabilizer matrices. This averts the exponential cost
that manipulating simultaneous eigenvectors would entail via standard gate decomposition techniques. In terms of classical complexity, the synthesis tool is fast because its slowest step is the Gaussian elimination in Algorithm 2, which has time complexity of $O(N^3)$ and small constant factors [13].

The actual quantum circuit produced by the synthesis procedure requires only $O(N^2)$ gates in the worst case. This follows because the Gaussian elimination can require $O(N^2)$ elementary row operations, which entails $O(N^2)$ CNOT gates. In practice, the gate count scaling is only quadratic if the partition sizes are linear. In cases such as Section V’s linear time transpilation, which has constant sized partitions (of size 4), the gate count is still $O(N)$.

While the $O(N^2)$ worst-case gate count for GC measurement is worse than the $O(N)$ gate count for Naïve 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 [17]. [29]. We base our studies on UCCSD because it is the gold standard for quantum computational chemistry [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 [32]. [31]. Even in the case of other non-hardware-driven ansatzes, gate counts and depths generally scale at least as $N^3$ in order to achieve high accuracy. Thus, the circuit cost of GC measurement appears to be benign, even in the worst case quadratic-cost scenario.

For demonstration, we show in Figure 9 the simultaneous measurement circuit for the 4 GC-but-not-QWC qubits in the Pauli partition for the Jordan-Wigner transformation discussed in Section V. Specifically, this measurement circuit is used to measure the green 8-clique in Figure 7.

![Fig. 9. Simultaneous measurement circuit generated by our software for the green 8-clique in Figure 7.](image)

VII. BENCHMARK RESULTS

We tested the performance of our simultaneous measurement strategies in Section V on multiple molecular benchmarks, whose Hamiltonians we obtained via OpenFermion [33]. Our benchmark results encompass both the reduction in number of partitions relative to Naïve, as well as the classical transpilation runtime required to produce the partitioning.

Figure 10 indicates the reduction in commuting partitions (cliques) found using both QWC and GC edges, in comparison to the Naïve VQE implementation in which each Pauli string is in a singleton partition. The MIN-CLIQUE-COVERs were approximated by the exponential Bron-Kerbosch algorithm, and thus should be regarded as soft upper bounds on the partitioning advantage that is practically achievable. The improvement from Naïve to QWC is consistently about 4-5x—a significant reduction especially considering that QWC measurement is cheap. The improvement from Naïve to GC ranges from 7x to 12x from $H_2$ to $CH_4$ (methane). This suggests that the advantage for GC partitioning improves for larger molecules.

![Fig. 10. 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 each molecular Hamiltonian.](image)
relatively large number of active spaces are needed to attain high accuracy [4].

![Bron-Kerbosch Simultaneous Measurement for H₂](image)

Fig. 11. Number of QWC and GC partitions generated by Bron-Kerbosch for the H₂ molecule, across an increasing number of active spaces.

We also benchmarked the performance of three polynomial time transpilation procedures, Boppana-Halldórsson (on QWC- and GC-edge graphs) [6], OpenFermion’s partitioning heuristic (on QWC-edge graphs only) [33], and our Linear-Time Transpilation on problem sizes ranging from 4 to 5237 terms in the molecular Hamiltonian. We generated a variety of Hamiltonians describing the H₂, LiH, H₂O, and CH₄ molecules, and recorded both the number of partitions generated and the runtime for each algorithm-Hamiltonian pair. Figures 12 and 13 show the reduction factor and wall clock runtimes for a subset of the benchmarks up to 630 Pauli strings. Figure 12 extends the general trend shown in Figure 10 to different partitioning algorithms: GC leads to much more optimal partitioning than QWC, because GC captures a denser edge set. Bron-Kerbosch GC achieves the fewest number of partitions (however, benchmarks past 275 terms were unable to be run due to exponential runtimes), and Boppana-Halldórsson GC achieves comparable optimality with quadratic runtime (see Figure 13). Our Linear-Time Transpilation consistently has an ~8x advantage. Among the QWC methods, we consistently see 3-4x reductions in number of partitions over Naïve separate measurements. Our Linear-Time Transpilation is by far the fastest—it is not plotted in Figure 13, since it always took < 0.01s. OpenFermion’s function is also fast, but is restricted to QWC-edge graphs only and has the lowest partitioning advantage.

VIII. EXPERIMENTAL RESULTS

We validated our techniques with a proof of concept demonstration by experimentally replicating a recent result [11]: 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 [1], which is cloud accessible.

Following [11], deuteron can be modeled with a 2-qubit Hamiltonian spanning 4 Pauli strings: IZ, ZI, XX, and YY. Under Naïve measurement, each Pauli string is measured in a separate partition. Under GC, we can partition into just two commuting families: \{IZ, IZ\} and \{XX, YY\}. Recall that the \{IZ, IZ\} partition is QWC and can be measured with simple Z-basis measurements. The \{XX, YY\} partition can be measured by the Bell basis measurement circuit in Figure 8.

To establish a fair comparison between Naïve measurement and simultaneous measurement we performed experiments in which both settings were allocated an equal budget of 100 total shots (trials). This corresponds to 25 shots per partition in Naïve measurement and 50 shots per partition in GC simultaneous measurement. We chose a small shot budget to emphasize the harmful effect of statistical variance with low shots. Figure 14 plots our results using a simplified Unitary Coupled Cluster ansatz with a single parameter and just three gates, as described in [11].

The results indicate reasonable agreement between Naïve measurement, GC measurement, and the true (Theory) values. The deviation from Theory stems both from statistical variance...
Fig. 14. Deuteron energy estimation under Naive and GC partitions, as executed on IBM Q20 with a total shot budget of 100. The energies are in MeV. Average error is 11% lower with GC simultaneous measurement than with Naive separate measurements.

due to the low shot budget, as well as systematic noise in the quantum processes. On average, the GC measurements had an error of 835 keV—11% less than the average error of 940 keV for Naive measurement.

These results are presented as proof-of-concept that simultaneous measurement can achieve higher accuracy than separate measurements. For several reasons, these experimental results underestimate the potential of simultaneous measurement, especially as more sophisticated quantum devices emerge. In particular:

- the Unitary Coupled Cluster ansatz of [11] 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.
- For a small Hamiltonian like the one considered here, the partitioning gain from GC is only 2x. As indicated in the benchmark results in Section VII we expect up to 30x gains for larger Hamiltonians and possibly a gain factor that continues to linearly increase for larger molecules, based on extrapolation of the benchmark results.
- The number of jobs (separate measurement circuits) is far more costly than the number of shots for practical purposes, since (a) it is far more expensive to switch circuits between jobs than to repeat shots within a job and (b) executions are scheduled at the granularity of jobs. In our experiments, 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.

IX. STATISTICS OF SIMULTANEOUS MEASUREMENT

We have shown both how to approximate a MIN-COMMUTING-PARTITION and how to actually construct the requisite simultaneous measurement circuits. Finally, we address an important question regarding the statistics of simultaneous measurement. This question was first raised by [34] Section IV B2] which proved that simultaneous measurement can actually underperform separate measurements due to the presence of covariance terms within partitions. In particular, the MIN-COMMUTING-PARTITION can require more total measurements than Naive measurements, in certain situations.

Our key result, depicted by Figure 1’s ‘Covariance Mitigation’ box, is to resolve this open issue in two ways: (1) we demonstrate that such a situation is atypical and (2) we demonstrate a strategy for detecting and course-correcting if we are dealing with such a situation. Before expanding on these points, we provide a pathological example for reference.

A. A Pathological Example

Consider the Hamiltonian, $H = IZ + ZI - XX - YY + ZZ$, following the example of [34]. The commutation graph has a bowtie shape. Figure 15 depicts two possible clique covers with $k = 2$ and $k = 3$ commuting-family partitions respectively.

For each ansatz state checked by VQE, we must perform enough repetitions to determine the expected value of the Hamiltonian to a target accuracy level $\epsilon$. The expected number of repetitions, $n_{\text{expect}}$, needed to achieve this accuracy for a $k$-way partitioning is [34]:

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

(2)

The variance from each partition can be computed from the formula for the variance of the sum of terms that make up the partition:

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

(3)
where \( \text{Cov}(M_1, M_2) = \langle M_1 M_2 \rangle - \langle M_1 \rangle \langle M_2 \rangle \).

In the state \( |\psi\rangle = |01\rangle \), all covariances evaluate to 0 except for \( \text{Cov}(-XX, -YY) = 1 \). Thus the \( k = 2 \) partitioning picks up this additional +1 penalty from the \(-XX, -YY\) covariance, whereas the \( k = 3 \) partitioning avoids this penalty by splitting the \(-XX\) and \(-YY\) measurements into separate partitions. Computing \( n_{\text{expect}} \) explicitly, we find \( 8/\epsilon^2 \) and \( 6/\epsilon^2 \) respectively for \( k = 2 \) and \( k = 3 \); thus, the seemingly suboptimal \( k = 3 \) partition is actually better under the \( |01\rangle \) ansatz state! This phenomenon motivates us to pay close attention to covariances within each partitioning.

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 show that when we have no prior attention to covariances within each partitioning.

Theorem 1. Given \( M_1, M_2 \), 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 \([56], [46]\)).

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 which 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 \([40]\) Chapter 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 
\[
\text{Cov}(M_1, M_2)|\psi\rangle = \langle M_1 M_2 |\psi\rangle - \langle M_1 \rangle \langle M_2 \rangle = (a^2 - b|2|^2 + c|2|^2 + d|2|^2) - (-a^2 - b|2|^2 + c|2|^2 + d|2|^2).
\]

Now consider the matching state:
\[
|\psi'\rangle = b |\psi_{-1,-1}\rangle + a |\psi_{-1,+,1}\rangle + d |\psi_{+,1,-1}\rangle + c |\psi_{+,1,+,1}\rangle
\]

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

Thus, 
\[
\text{Cov}(M_1, M_2)|\psi\rangle = -\text{Cov}(M_1, M_2)|\psi'\rangle.
\]

C. Mitigating Covariances: Partition Splitting

While we have now secured the top level goal of initially performing measurements under the MIN-COMMUTING-PARTITION approximation, we demonstrate an adaptive strategy for detecting and correcting course in the atypical case when a simultaneous measurement should be split into separate measurements.

Let us consider two partitionings: a baseline partitioning with \( k \) partitions, and a candidate clique-splitting partitioning with \( k' > k \) partitions. The \( k' \)-partition clique-splitting partition should be favored if it has a lower \( n_{\text{expect}} \). Following from equation 2, this occurs if \( k' \sum_{i=1}^{k'} \text{Var}(\text{Partition } i) \leq k \sum_{i=1}^{k} \text{Var}(\text{Partition } i) \).

Returning to the example of Section [X-A], this means that the \( k' = 3 \) partition should be favored over the \( k = 2 \) partition if:
\[
3[\text{Var}(-XX) + \text{Var}(-YY, ZZ)] + \text{Var}(IZ, ZI) \leq 2[\text{Var}(-XX, -YY, ZZ)] + \text{Var}(IZ, ZI)]
\]

(4)

Using equation 3 to expand these variances into their component terms, we note that the terms present on the left-hand side but not the right-hand side of Equation 4 are those that are “broken” by the clique-splitting \( k' = 3 \) setting:

broken terms = \{2\text{Cov}(-XX, -YY), 2\text{Cov}(-XX, ZZ)\}

And the terms that are found on both the left- and right-hand sides of equation 4 which we call the “unbroken terms” are:

unbroken terms = \{\text{Var}(-XX), \text{Var}(-YY), \text{Var}(ZZ), \text{Var}(IZ), \text{Var}(IZ), 2\text{Cov}(-YY, ZZ), 2\text{Cov}(ZI, IZ)\}

This generalizes into the rule that fewer measurements are needed in a clique-splitting partitioning with \( k' > k \) partitions, as compared with the \( k \)-partition setting if:
\[
k \sum \text{broken terms} > (k' - k) \sum \text{unbroken terms} \quad (5)
\]

D. Strategies for covariance estimation

In practice, the true theoretical values of these covariances 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 covariances of Pauli strings in the same partition.

With each additional measurement, we calculate the sample covariance \( \text{Cov}(M_1, M_2) = \frac{1}{n-1} \sum_{i=1}^{n}(m_{i1} - \bar{m}_1)(m_{i2} - \bar{m}_2) \), where \( \{m_{11}, ..., m_{1n}\} \) and \( \{m_{21}, ..., m_{2n}\} \) are the \( n \) observed measurements of \( M_1 \) and \( M_2 \) respectively, and check whether the criterion in (5) is satisfied.
X. Conclusion

Simultaneous measurement substantially reduces the cost of Variational Quantum Eigensolver by allowing measurements to cover several Pauli strings simultaneously. We demonstrate transpilation procedures that achieve up to 30x reductions in the number of requisite measurements. We also raise practical concerns about the classical transpilation costs and identify an alternate strategy that exploits properties of molecular Hamiltonians to achieve an 8x reduction in quantum cost in linear time with respect to the number of Pauli strings. Our systems emphasis also includes explicit attention to the overhead of simultaneous measurement circuits. Accordingly, we develop a circuit synthesis procedure, implemented and tested in software. 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, via ground state estimation of deuteron. All of our software and results are available online [2].

Acknowledgements

We would like to thank Kenneth Brown, Peter Love, and Will Kirby for helpful discussions about VQE. We are also grateful to Olivia Di Matteo for introducing us to MUBs and clarifying the construction of simultaneous measurement circuits. Additionally, we acknowledge Henry Hoffmann for advising us on the time-to-solution for classical optimizers.

This work is funded in part by EPIQC, an NSF Expedition in Computing, under grants CCF-1730449/1730082; in part by STAO, under grant NSF Phy-1818914; and in part by DOE grants DE-SC0020289 and DE-SC0020331. P. G. is supported by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate Fellowship Program. O. A. is supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE 1752814. The work of K. G. and M. S. is supported by the U.S. Department of Energy, Office of Science, under contract number DE-AC02-06CH11357.

References


