symmetry

MDPI

Article

Efficient Application of the Factorized Form of the Unitary Coupled-Cluster Ansatz for the Variational Quantum Eigensolver Algorithm by Using Linear Combination of Unitaries

Luogen Xu * and James K. Freericks

Department of Physics, Georgetown University, 37th and O Sts. NW, Washington, DC 20057, USA; james.freericks@georgetown.edu

* Correspondence: lx63@georgetown.edu

Abstract: The variational quantum eigensolver is one of the most promising algorithms for near-term quantum computers. It has the potential to solve quantum chemistry problems involving strongly correlated electrons with relatively low-depth circuits, which are otherwise difficult to solve on classical computers. The variational eigenstate is constructed from a number of factorized unitary coupled-cluster terms applied onto an initial (single-reference) state. Current algorithms for applying one of these operators to a quantum state require a number of operations that scale exponentially with the rank of the operator. We exploit a hidden SU(2) symmetry to allow us to employ the linear combination of unitaries approach, Our PREPARE subroutine uses $n+2$ ancilla qubits for a rank-n operator. Our SELECT($\hat{U}$) scheme uses $\mathcal{O}(n)$ CNOT gates. This results in a full algorithm that scales like the cube of the rank of the operator n^3, a significant reduction in complexity for rank five or higher operators. This approach, when combined with other algorithms for lower-rank operators (when compared to the standard implementation), will make the factorized form of the unitary coupled-cluster approach much more efficient to implement on all types of quantum computers.

Keywords: factorized unitary coupled-cluster ansatz; quantum chemistry; linear combination of unitaries

Citation: Xu, L.; Freericks, J.K. Efficient Application of the Factorized Form of the Unitary Coupled-Cluster Ansatz for the Variational Quantum Eigensolver Algorithm by Using Linear Combination of Unitaries. *Symmetry* **2023**, *15*, 1429. https://doi.org/10.3390/sym15071429

Academic Editor: Michel Planat

Received: 7 June 2023
Revised: 5 July 2023
Accepted: 6 July 2023
Published: 16 July 2023

1. Introduction

One of the important motivations for developing quantum computers is their potential to simulate strongly correlated many-body systems efficiently [1,2]. Algorithms that exactly diagonalize the electronic Hamiltonian, known as the full configuration interaction approach, scale exponentially with the size of the Hilbert space, making it applicable to very few cases [3] on classical computers. The configuration interaction (CI) method offers an approximate solution by truncating the Hilbert space to only include the most important basis states. However, the energy calculated by the CI method does not scale properly with the size of the system when used on molecules with varying sizes, nor does it predict the dissociation energy correctly because it cannot produce factorized atomic states. The coupled-cluster (CC) method addresses these issues by being both size consistent and size extensive. Size consistency means that the method would yield the same energy of two particles separated by an infinite distance as the sum of the energies calculated individually. Size extensivity means the energy scales linearly with the number of particles for a homogeneous system [4]. The CC method is also memory efficient because it does not explicitly construct the energy eigenstate. Instead, the set of amplitudes for the CC ansatz is calculated iteratively by the so-called amplitude equations [4–6], which correspond to zeroing out the row (or column) of the similarity-transformed Hamiltonian matrix that corresponds to the initial single-reference state. The CC method with single, double and (perturbative) triple excitations is regarded as the "gold standard" for computational chemistry [7].

Quantum computers have been proposed as being capable of solving a set of quantum chemistry problems that are otherwise difficult or very challenging on classical machines: namely, molecules that contain both weakly and strongly correlated electrons. One of the most promising algorithms for the noisy intermediate-scale quantum (NISQ) era is the variational quantum eigensolver (VQE), where the trial wave function is prepared on the quantum hardware and the expectation value of the Hamiltonian is measured there as well; the parameters in the eigenstate are optimized variationally on classical machines [8,9]. The conventional coupled-cluster ansatz is given as $|\psi_{CC}\rangle = e^{\hat{T}}\left|\psi_{ref}\right\rangle$, where $\left|\psi_{ref}\right\rangle$ is a trial wave function (often chosen to be the single-reference Hartree–Fock state), and $\hat{T} = \sum_{k=1}^{n} \hat{T}_k$ is the cluster operator consisting of up to rank-n excitations (n electrons are removed from the Hartree–Fock state and replaced by n electrons in virtual orbitals). The excitation operator is given as

$$\hat{T}_k = \frac{1}{(k!)^2} \sum_{ij\cdots}^{occ} \sum_{ab\cdots}^{vir} t_{ij\cdots}^{ab\cdots} \hat{A}_{ij\cdots}^{ab\cdots}, \tag{1}$$

and $\hat{A}_{ij\cdots}^{ab\cdots} = \hat{a}_a^\dagger \hat{a}_b^\dagger \cdots \hat{a}_j \hat{a}_i$, where $\hat{a}_a^\dagger$ is the creation operator acting on virtual orbital a and $\hat{a}_i$ is the annihilation operator acting on occupied orbital i. Traditionally, the CC method employs a similarity-transformed Hamiltonian to obtain a set of equations to determine the amplitudes t:

$$\left\langle \psi_{ref}\left|e^{-\hat{T}}\hat{H}e^{\hat{T}}\right|\psi_{ref}\right\rangle = E \tag{2}$$

$$\left\langle \psi_{\mu}\left|e^{-\hat{T}}\hat{H}e^{\hat{T}}\right|\psi_{ref}\right\rangle = 0 \tag{3}$$

where $\langle\psi_\mu| = \left\langle \psi_{ref}\right|\hat{A}_\mu$, and $\{\langle\psi_\mu|\}$ is a set of states that covers the entire space generated by $\hat{T}_{CCSD}$ acting on the reference state [5,10]. In practice, this set of amplitude equations is solved iteratively, which yields the energy without needing to construct the energy eigenstate. The total number of amplitude equations is given by the number of amplitudes in the expansion of the $\hat{T}$ operator, which is much smaller than the total number of Slater determinants in the $|\psi_{CC}\rangle$ (which is typically exponentially larger). The properties of size consistency and size extensivity for the CC ansatz stem from the facts that the similarity-transformed Hamiltonian $e^{-\hat{T}}\hat{H}e^{\hat{T}}$ is additively separable and the term e^T is multiplicatively separable. Notice that the electronic Hamiltonian for the molecule (in second quantization) is given by

$$H = \sum_{ij} h_{ij} \hat{a}_i^\dagger \hat{a}_j + \frac{1}{2} \sum_{ijkl} g_{ijkl} \hat{a}_i^\dagger \hat{a}_j^\dagger \hat{a}_k \hat{a}_l, \tag{4}$$

where h_{ij} are the one-electron integrals and g_{ijkl} are the two-electron integrals:

$$h_{ij} = \int dr_1 \phi_i^*(r_1) \left(-\frac{1}{2}\nabla_{r_1}^2 - \sum_{I=1}^{M} \frac{Z_I}{R_{1I}} \right) \phi_j(r_1) \tag{5}$$

$$g_{ijkl} = \int dr_1 dr_2 \phi_i^*(r_1) \phi_j^*(r_2) \frac{1}{r_{12}} \phi_k(r_1) \phi_l(r_2). \tag{6}$$

Here, M is the number of atoms in the system, Z_I are their atomic numbers, $R_{1I} = |r_1 - R_I|$, $r_{12} = |r_1 - r_2|$, and $\phi(r)$ are the single-particle optimized orbitals from the HF solution [11,12]. In order to solve the amplitude Equations (2) and (3), we need to explicitly compute the similarity-transformed Hamiltonian. Using the Hadamard lemma (also called the Baker–Campbell–Hausdorff expansion or the Baker–Campbell formula), we can rewrite the transformed Hamiltonian as

$$e^{-\hat{T}}\hat{H}e^{\hat{T}} = \hat{H} + [\hat{H},\hat{T}] + \frac{1}{2!}[[\hat{H},\hat{T}],\hat{T}] + \frac{1}{3!}[[[\hat{H},\hat{T}],\hat{T}],\hat{T}] + \frac{1}{4!}[[[[\hat{H},\hat{T}],\hat{T}],\hat{T}],\hat{T}] + \cdots \tag{7}$$

Conveniently, the series truncates at the fourth order due to the Hamiltonian having only one- and two-body interaction terms [5,13] and the excitations always being from real to virtual orbitals. Traditionally, this projective method to determine the CC amplitudes is preferred over variational methods due to the non-unitarity of the $e^{\hat{T}}$ operator [5,7].

Despite its success, the lack of unitarity prevents the CC operators from being implemented on quantum computers. This suggests using the unitary coupled-cluster ansatz (UCC), whose cluster operator now includes the excitation minus the de-excitation operator $\hat{T} - \hat{T}^\dagger$ [14,15]. Similar to the CC approximation, only the low-rank cluster operators such as singles and doubles are usually selected for the variational eigenstate ansatz, but for more strongly correlated systems, one expects that higher-rank factors will also be needed. In practice, a projective method like the one used in the CC calculation does not work with the UCC ansatz because the similarity-transformed Hamiltonian no longer truncates after the fourth term. Common strategies for carrying it out on classical computers include truncating the Hadamard lemma series at a fixed order [14], expanding the exponential operator in a power series and then truncating it when the higher-rank terms no longer change the eigenfunction [16], and using an exact operator identity of the factorized form of the UCC to allow the wavefunction to be constructed in a tree structure [17]. But, there exists no simple method to work directly with the UCC ansatz in its original form. Since we are working with non-commuting fermionic operators $\hat{a}_a^\dagger \hat{a}_b^\dagger \cdots \hat{a}_j \hat{a}_i - \hat{a}_i^\dagger \hat{a}_j^\dagger \cdots \hat{a}_b \hat{a}_a$ in the exponent, one common way to decompose such a function is to adopt a Trotter product formula:

$$\hat{U}_{UCC} = e^{\sum_k \theta_k (\hat{T}_k - \hat{T}_k^\dagger)} = \lim_{N \to \infty} \left(\prod_k e^{\frac{\theta_k}{N} (\hat{T}_k - \hat{T}_k^\dagger)} \right)^N. \tag{8}$$

Another useful method is to express the ansatz in a factorized form, given by

$$\hat{U}'_{UCC} = \prod_k e^{\theta_k (\hat{T}_k - \hat{T}_k^\dagger)}, \tag{9}$$

which corresponds to the first-order approximation of the Trotter product formula in Equation (8). The benefit of only using the $N = 1$ extreme case is two-fold: the quantum resources required to prepare the factorized UCC ansatz are much smaller than higher-order approximations and the Trotter errors of the first-order approximation can be ameliorated by the fact that the calculation is variational [18,19]. Within the classical computational chemistry framework, work by Chen et al. [17] created an algorithm using the factorized form of the UCC that produces significantly better results for strongly correlated systems and comparable results in terms of accuracy for weakly correlated systems. In this work, we show how one can create efficient implementation of these UCC factors using the linear combination of unitaries approach. For high-rank factors, this approach is preferable.

To implement the factorized UCC ansatz on quantum computers, one needs to transform the cluster operators $\hat{T} - \hat{T}^\dagger$ expressed in the fermionic language into a spin language (via the Jordan–Wigner transformation, or other fermionic encodings). A common realization of this approach is to exactly simulate the individual exponentials of Pauli strings found after the JW transformation of $e^{\hat{T} - \hat{T}^\dagger}$ [18,19]. For example, a Jordan-Wigner transformed rank-2 UCC factor is given as

$$\begin{aligned} \exp\left(\frac{\theta_{ijkl}}{2}\left(\hat{a}_i^\dagger \hat{a}_j^\dagger \hat{a}_k \hat{a}_l - \hat{a}_l^\dagger \hat{a}_k^\dagger \hat{a}_j \hat{a}_i\right)\right) = \exp\Bigg(\frac{i\theta_{ijkl}}{16} \bigotimes_{a=l+1}^{k-1} Z_a \\ \bigotimes_{b=j+1}^{i-1} Z_b \times \Big(X_l X_k Y_j X_i + Y_l X_k Y_j Y_i + X_l Y_k Y_j Y_i + X_l X_k X_j Y_i \\ - Y_l X_k X_j X_i - X_l Y_k X_j X_i - Y_l Y_k Y_j X_i - Y_l Y_k X_j Y_i\Big)\Bigg). \end{aligned} \tag{10}$$

Such a UCC factor can be rewritten as a product of exponentials of Pauli strings because the Pauli strings in the exponentials all commute [18,20]. A common strategy for creating circuits of the form $\exp\{-i\frac{\theta}{2}Z_1Z_2\ldots Z_n\}$ is to use basis transformations where one starts with the circuit for evaluating $\exp\{-i\frac{\theta}{2}Z_1Z_1\ldots Z_n\}$ and then apply basis transformations to evaluate the exponential of any Pauli string [21]. In order to evaluate a generic Pauli string, a basis transformation can be applied before the CNOT cascades such that the effective Pauli string is that of only Z's. If the ith gate in the Pauli string is an X, a Hadamard gate is sandwiched around the CNOT cascade on the ith qubit. This leads to the effective exponential containing a Z since $HXH = Z$. Figure 1 shows an example circuit to apply $\exp\{-i\frac{\theta}{2}Z_1Z_2Z_3X_4\}$. Similarly, if an exponentiated Y gate is applied, a $R_x(-\frac{\pi}{2})$ gate is sandwiched around the CNOT cascade. Figure 2 shows an example circuit to apply Equation (10). In applying the UCC ansatz, circuits such as Figure 2 must be re-run 2^{2n-1} times after applying all of the 2^{2n-1} different basis transformations [18]. A general factorized doubles UCC operator can be rewritten as Equation (10) and implemented exactly by the circuit shown in Figure 2.

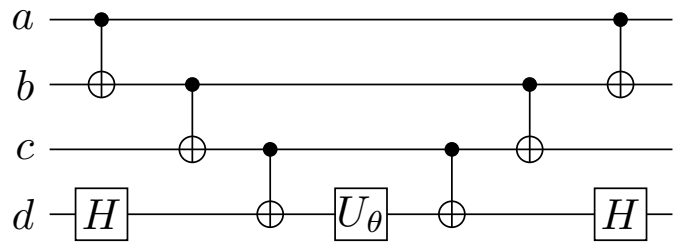

Figure 1. Example of a circuit implementing $\exp\{-i\frac{\theta}{2}Z_a \otimes Z_b \otimes Z_c \otimes X_d\}$ for four qubits. To apply the X on a different qubit, Hadamard gates can be sandwiched around the respective qubits.

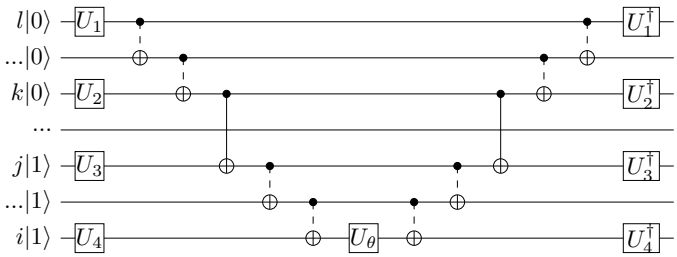

Figure 2. Doubles UCC circuit as discussed in Refs. [18,19]. For a general doubles operator, the circuit must be applied eight times, with different combinations of U gates each time. The U-gate choices are summarized in Table 1. The dashed CNOT gates are part of a CNOT cascade.

Table 1. Eight different subcircuits run sequentially to apply a UCC doubles factor to a wavefunction. Since the relevant operators all commute, the subcircuits can be run in any order, but all eight need to appear exactly once to complete the full circuit.

Subcircuit	U_1	U_2	U_3	U_4
1	H	H	$R_x(-\frac{\pi}{2})$	H
2	$R_x(-\frac{\pi}{2})$	H	$R_x(-\frac{\pi}{2})$	$R_x(-\frac{\pi}{2})$
3	H	$R_x(-\frac{\pi}{2})$	$R_x(-\frac{\pi}{2})$	$R_x(-\frac{\pi}{2})$
4	H	H	H	$R_x(-\frac{\pi}{2})$
5	$R_x(-\frac{\pi}{2})$	H	H	H
6	H	$R_x(-\frac{\pi}{2})$	H	H
7	$R_x(-\frac{\pi}{2})$	$R_x(-\frac{\pi}{2})$	$R_x(-\frac{\pi}{2})$	H
8	$R_x(-\frac{\pi}{2})$	$R_x(-\frac{\pi}{2})$	H	$R_x(-\frac{\pi}{2})$

This is possible because the different 2^{2n-2} Pauli strings (for a rank-n UCC factor) commute with each other. In our previous work, we found a way of reducing the number of control-NOT (CNOT) gates in quantum circuits for the factorized UCC ansatz by

introducing extra ancilla qubits [20], with the largest reductions for the higher-rank factors. Alternatively, a factorization method introduced in Ref. [22] uses a two-step low-rank factorization to approximate the UCC operator. Circuits that implement the SELECT($\hat{U}$) subroutine for more general Jordan–Wigner strings with linear scaling have also been developed [23]. Another framework developed in Ref. [24] optimizes the two-qubit gates by bootstrapping the VQE iterations towards the convergence of the systems ground state energy. A quantum software developed in Ref. [25] optimizes the number of two-qubit gates systematically by compressing adjacent *CNOT* and Pauli operators using a set of rules from the *ZX*-calculus. Other works have proposed different efficient methods to simulate the UCC factors[26], whose main idea is to directly implement the SU(2) identity of the UCC factors presented in Equation (22) by exchanging coefficients between the two active states. In this work, we introduce a method to directly simulate the sum of terms obtained from a hidden SU(2) symmetry of the first-order Trotter product that greatly reduces the number of multi-qubit entanglement gates of factorized UCC circuits.

2. Background

2.1. SU(2) Identity for Individual UCC Factors

Recall the rank-n cluster operator is defined as

$$\hat{T}_k = \frac{1}{(k!)^2} \sum_{ij\cdots}^{occ} \sum_{ab\cdots}^{vir} \theta_{ij\cdots}^{ab\cdots} \left(\hat{A}_{ij\cdots}^{ab\cdots} - \hat{A}_{ab\cdots}^{ij\cdots} \right). \tag{11}$$

The first two ranks are

$$\hat{T}_1 = \sum_{ia} \theta_i^a \left(\hat{a}_a^\dagger \hat{a}_i - \hat{a}_i^\dagger \hat{a}_a \right) = \sum_{ia} \theta_i^a \left(\hat{A}_i^a - \hat{A}_a^i \right) \tag{12}$$

$$\begin{aligned} \hat{T}_2 &= \frac{1}{4} \sum_{ijab} \theta_{ij}^{ab} \left(\hat{a}_a^\dagger \hat{a}_b^\dagger \hat{a}_j \hat{a}_i - \hat{a}_i^\dagger \hat{a}_j^\dagger \hat{a}_b \hat{a}_a \right) \\ &= \frac{1}{4} \sum_{ijab} \theta_{ij}^{ab} \left(\hat{A}_{ij}^{ab} - \hat{A}_{ab}^{ij} \right), \end{aligned} \tag{13}$$

where $\hat{a}_a^\dagger$ is the fermionic creation operator on the virtual orbital a and $\hat{a}_i$ is the fermionic annihilation operator on the occupied orbital i, and they obey the standard anticommutation relations given by

$$\{\hat{a}_i, \hat{a}_j\} = 0; \{\hat{a}_i^\dagger, \hat{a}_j^\dagger\} = 0; \{\hat{a}_i, \hat{a}_j^\dagger\} = \delta_{ij} \tag{14}$$

where $\{A, B\} = AB + BA$ and δ_{ij} is the Kronecker delta function. First, we note that because $\{i, j, k, \cdots\}$ and $\{a, b, c, \cdots\}$ are disjoint sets, $\hat{A}^2 = 0 = \hat{A}^{\dagger 2}$, so the squared term becomes

$$\begin{aligned} (\hat{A} - \hat{A}^\dagger)^2 = -\hat{A}\hat{A}^\dagger - \hat{A}^\dagger \hat{A} = &-\hat{n}_{a_1} \hat{n}_{a_2} \cdots \hat{n}_{a_n} (1 - \hat{n}_{i_1})(1 - \hat{n}_{i_2}) \cdots (1 - \hat{n}_{i_n}) \\ &- (1 - \hat{n}_{a_1})(1 - \hat{n}_{a_2}) \cdots (1 - \hat{n}_{a_n}) \hat{n}_{i_1} \hat{n}_{i_2} \cdots \hat{n}_{i_n}, \end{aligned} \tag{15}$$

where $\hat{n}_\alpha = \hat{a}_\alpha^\dagger \hat{a}_\alpha$ is the number operator for spin-orbital α. The cubed term then becomes

$$(\hat{A} - \hat{A}^\dagger)^3 = \hat{A}\hat{A}^\dagger \hat{A} - \hat{A}^\dagger \hat{A}\hat{A}^\dagger = \hat{A} - \hat{A}^\dagger, \tag{16}$$

because the projection operators $\hat{n}$ and $1 - \hat{n}$ evaluate to one when they act on the corresponding fermionic operators. For any UCC factor, the power series expansion is given as

$$e^{\theta(\hat{A} - \hat{A}^\dagger)} = \sum_{n=0}^{\infty} \frac{\theta^n}{n!} (\hat{A} - \hat{A}^\dagger)^n. \tag{17}$$

Combining with Equations (15) and (16), we can then exactly write the sum as

$$e^{\theta(\hat{A}-\hat{A}^\dagger)} = \hat{I} + \sin\theta(\hat{A}-\hat{A}^\dagger) + (\cos\theta - 1)(\hat{n}_{a_1}\hat{n}_{a_2}\cdots\hat{n}_{a_n}(1-\hat{n}_{i_1})(1-\hat{n}_{i_2})\cdots(1-\hat{n}_{i_n}) + (1-\hat{n}_{a_1})(1-\hat{n}_{a_2})\cdots(1-\hat{n}_{a_n})\hat{n}_{i_1}\hat{n}_{i_2}\cdots\hat{n}_{i_n}), \quad (18)$$

for any given set of occupied orbitals $\{i_1 \cdots i_n\}$ and virtual orbitals $\{a_1 \cdots a_n\}$ of rank n [17,27,28]. This identity gives a clear picture of what is happening after a UCC factor is applied to a state. If the state is neither excited by $\hat{A}$ nor de-excited by $\hat{A}^\dagger$, the state is unchanged by the UCC factor. Otherwise, the UCC factor acting on the state is equivalent to a cosine multiplied by the original state plus a sine multiplied by the excited (or de-excited) state, just as we would expect from a rotation in the many-body configuration space.

2.2. Jordan–Wigner Transformation of the SU(2) Identity

Hamiltonians written in fermionic terms need to be re-expressed in terms of spin operators in order to be implemented by quantum computers. In this work, we choose to work with the JW transformation for the fermionic encoding. This transformation is given by

$$\hat{a}_k = \frac{1}{2}(X_k + iY_k) \otimes Z_{k+1} \otimes Z_{k+2} \otimes \cdots \otimes Z_N \quad (19)$$

$$\hat{a}_k^\dagger = \frac{1}{2}(X_k - iY_k) \otimes Z_{k+1} \otimes Z_{k+2} \otimes \cdots \otimes Z_N \quad (20)$$

$$\hat{n}_k = \hat{a}_k^\dagger \hat{a}_k = \frac{1}{2}(1 - Z_k), \quad (21)$$

where X, Y, and Z are the standard Pauli matrices, and $0 \leq k \leq N-1$, for the N qubits that describe the molecule. The qubit state $|0\rangle$ has no electrons and $|1\rangle$ has one electron. The SU(2) identity for a UCC factor, as shown in Equation (18), can be re-expressed in terms of the Pauli operators using Equations (19)–(21). For a factorized UCC double (UCCD) operator, the transformation is as follows:

$$\begin{aligned}
\hat{U}(\theta) =& \exp\left(\theta(\hat{a}_i^\dagger \hat{a}_j^\dagger \hat{a}_k \hat{a}_l - \hat{a}_l^\dagger \hat{a}_k^\dagger \hat{a}_j \hat{a}_i)\right) \\
=& \hat{I} + \sin\theta(\hat{a}_i^\dagger \hat{a}_j^\dagger \hat{a}_k \hat{a}_l - \hat{a}_l^\dagger \hat{a}_k^\dagger \hat{a}_j \hat{a}_i) + (\cos\theta - 1) \\
& \times (\hat{n}_l \hat{n}_k (1-\hat{n}_i)(1-\hat{n}_j) + (1-\hat{n}_l)(1-\hat{n}_k)\hat{n}_i \hat{n}_j) \\
=& \hat{I} + \frac{i\sin\theta}{8} \bigotimes_{a=l+1}^{k-1} Z_a \bigotimes_{b=j+1}^{i-1} Z_b \times \\
& \Big(X_l X_k Y_j X_i + Y_l X_k Y_j Y_i + X_l Y_k Y_j Y_i + X_l X_k X_j Y_i \\
& - Y_l X_k X_j X_i - X_l Y_k X_j X_i - Y_l Y_k Y_j X_i - Y_l Y_k X_j Y_i \Big) \\
& + \frac{1}{8}(\cos\theta - 1)(\hat{I} + Z_i Z_j + Z_l Z_k - Z_j Z_l - Z_j Z_k \\
& - Z_i Z_l - Z_i Z_k + Z_i Z_j Z_k Z_l).
\end{aligned} \quad (22)$$

Note that the JW strings simplify because $Z_k^2 = \mathbb{1}$ for all cases where two strings overlap. This expression is a unitary operator, but it is also here expressed as a sum over unitary operators because Pauli strings are both Hermitian and unitary.

2.3. Linear Combination of Unitaries

To simulate the sum in Equation (22) on a quantum computer, we use the linear combination of unitaries (LCU) query model [29,30]. Given an input operator $\hat{U}$ represented by a sum of unitaries $\hat{U} = \sum_n \alpha_n \hat{U}_n$, with coefficients $\alpha_n > 0$ for each unitary operator $\hat{U}_n$,

the LCU technique will create a circuit to evaluate this operator acting on a state. It first prepares an ancilla bank with coefficients based on the coefficients in the linear combination:

$$\hat{B}|0\rangle = \frac{1}{\sqrt{s}} \sum_n \sqrt{\alpha_n}|n\rangle. \tag{23}$$

Here, $\frac{1}{\sqrt{s}}$ is a normalization factor, $|0\rangle$ is the initial state of the ancilla bank, and $|n\rangle$ is the product state that will later encode the unitaries in the LCU procedure. The operator SELECT($\hat{U}$) is then used to create entanglement between the ancilla bank and system states

$$\text{SELECT}(U)|n\rangle \otimes |\psi\rangle = |n\rangle \otimes U_n|\psi\rangle. \tag{24}$$

One of the hallmarks of the LCU approach is that if the original operator $\hat{U}$ is unitary and $s \leq \sqrt{2}$, then a single step of oblivious amplitude amplification is able to exactly apply the $\hat{U}$ to the state [29]. Note that in our case the UCC factor, given in Equation (22) is unitary and $s = \cos\theta + \sin\theta \leq 2$ for all θ, so it always satisfies this criteria. Hence, the LCU treatment of the sum is exact. The oblivious amplitude amplification is given by

$$-\hat{W}\hat{R}\hat{W}^\dagger\hat{R}\hat{W}|0\rangle \otimes |\psi\rangle = |0\rangle \otimes \hat{U}|\psi\rangle, \tag{25}$$

where the $\hat{W}$ and $\hat{R}$ operators are defined as

$$\begin{aligned} \hat{W} :=& (\hat{B}^\dagger \otimes \mathbb{1})\text{SELECT}(\hat{U})(\hat{B} \otimes \mathbb{1}), \\ \hat{R} :=& \mathbb{1} - 2(|0\rangle\langle 0| \otimes \mathbb{1}). \end{aligned} \tag{26}$$

The main source of circuit complexity of the LCU query model comes from the unitary transformation $\hat{W}$ because it involves applying SELECT($\hat{U}$), which itself can contain a substantial number of multi-qubit gates and quickly outgrows the capability of near-term hardware. One efficient circuit implementation of the SELECT($\hat{U}$) subroutine for a generic fermionic Hamiltonian uses $\mathcal{O}(\eta)$ Clifford and T gates, with Clifford gates running in $\mathcal{O}(\log^2 \eta)$ layers and T gates in $\mathcal{O}(\eta)$ layers. Here, η is the number of spin orbitals [23]. The ancilla preparation operator $\hat{B}$ is often implemented by rotations and controlled rotations on the target qubits, followed by Hadamard$^{\otimes\eta}$ gates that create the required entanglement state for the ancilla bank.

3. Circuit Construction

We begin by illustrating the circuit implementation of the PREPARE and SELECT($\hat{U}$) subroutines present in the LCU adaptation of the UCC factors for doubles. The doubles are the most ubiquitous terms in the low-rank representation of a UCC ansatz. Later in this section, we will show that UCC factors of arbitrary rank-n can be implemented via a similar algorithm. High-rank factors are necessary to generate an accurate correlation energy in strongly correlated systems.

3.1. PREPARE *Subroutine*

The unitary transformation $\hat{B}$ is used to generate the required entangled state in the ancilla bank, shown in Equation (23). The operator in Equation (22) lends itself to a binary encoding, where we create the linear combination of states multiplied by amplitudes, that is, the sum of $\alpha_1|0000\rangle + \alpha_2|0001\rangle + \cdots + \alpha_8|1111\rangle$. Because there are only three distinct coefficients present in the UCC factor regardless of the rank, the binary encoding allows us to reduce the size of the ancilla bank logarithmically so that it grows with the rank not with the exponential of the rank.

A PREPARE circuit for the doubles factor is shown in the Figure 3. The H gates are Hadamard operators, and the R_{X_i} and R_{Y_i} gates are rotations by an angle Θ_i along the X and Y axis, respectively. For a UCC doubles operator, $n = 2$, and there are three distinct

coefficients: eight terms with $\frac{i\sin\theta}{8}$, seven terms with $\frac{\cos\theta-1}{8}$, and one term with $\frac{\cos\theta+7}{8}$. The four angles used in the circuit shown in Figure 3 can be found analytically:

$$\Theta_1 = \arcsin\left(-\frac{\sqrt{2}}{4}\sin\theta\right) \tag{27}$$

$$\Theta_2 = \arcsin\left(\frac{\cos\theta-1}{\sqrt{14+2\cos^2\theta}}\right) \tag{28}$$

$$\Theta_3 = \arcsin\left(\frac{\sqrt{2}}{2}\frac{\cos\theta-1}{\sqrt{13+\cos^2\theta+2\cos\theta}}\right) \tag{29}$$

$$\Theta_4 = \arcsin\left(\frac{\sqrt{2}}{2}\frac{\cos\theta-1}{\sqrt{25+\cos^2\theta+6\cos\theta}}\right) \tag{30}$$

And the magnitudes of these four angles are shown in Figure 4. The PREPARE subroutine is implemented by encoding the amplitudes $1+\frac{\cos\theta-1}{2^{2n-1}}$ and $\frac{\cos\theta-1}{2^{2n-1}}$ in 2^{2n-1} states that always have one different binary digit than those encoding the amplitude $\frac{i\sin\theta}{2^{2n-1}}$. For example, in the doubles circuit, we encode the amplitudes $1+\frac{\cos\theta-1}{2^3}$ and $\frac{\cos\theta-1}{2^3}$ in states $|0000\rangle$ and $|0001\rangle\cdots|0111\rangle$ and encode the amplitude $\frac{i\sin\theta}{2^3}$ in states $|1000\rangle\cdots|1111\rangle$.

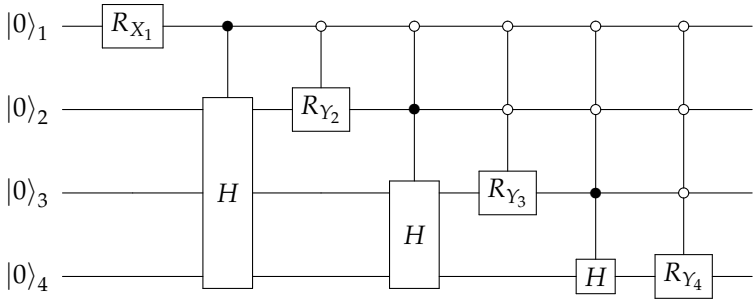

Figure 3. Quantum circuit for preparing the ancilla bank of the LCU query for a rank-2 UCC factor (so-called doubles).

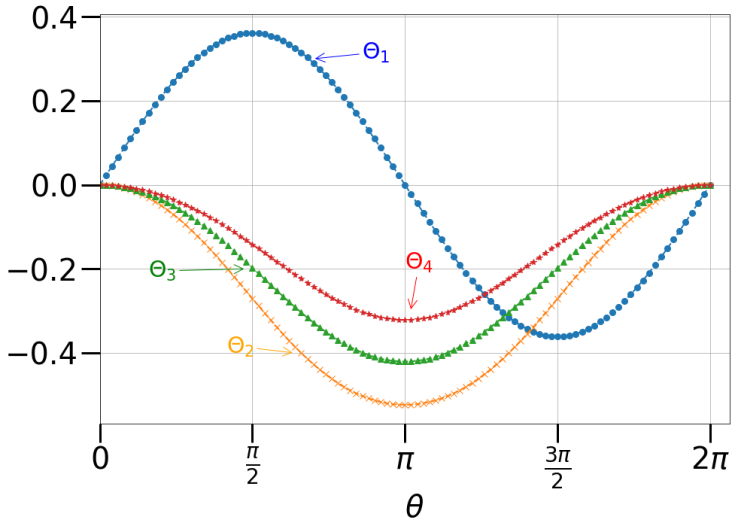

Figure 4. The four angles used in the PREPARE subroutine for a UCC doubles factor with an amplitude given by θ.

The PREPARE circuit for a doubles UCC factor can be straightforwardly generalized to one for a rank-n operator. In this case, we require $2n$ ancilla qubits, where there are 2^{2n-1} subterms with coefficients $\frac{i\sin\theta}{2^{2n-1}}$, $2^{2n-1}-1$ subterms with coefficients $\frac{\cos\theta-1}{2^{2n-1}}$, and

one term with coefficient $1 + \frac{\cos\theta - 1}{2^{2n-1}}$. The hierarchical structure of the circuit is a simple generalization of the doubles circuit to higher rank, where the first element, which sets the sin terms is the same, while the remaining factors are created by extending the hierarchy with multiple controlled Hadamards followed by multiply controlled rotations. The angles for each rotation in the algorithm are

$$\Theta_1 = \arcsin\left(-\frac{1}{\sqrt{2^{2n-1}}}\sin\theta\right) \tag{31}$$

$$\Theta_k = \arcsin\left(\frac{\cos\theta - 1}{\sqrt{2^{2n-2+k} - 2^k + 2 + 2\cos^2\theta + (2^k - 4)\cos\theta}}\right) \tag{32}$$

for $2 \leq k \leq 2n$.

Most of the quantum computing costs on near-term quantum computers come from the CNOT circuit elements; thus, we focus on counting the number of these gates to estimate the total cost of the circuits. The controlled-H gate, CH, is implemented by one CNOT sandwiched by one $R_Y(\pi/4)$ gate and one $R_Y(-\pi/4)$ gate. The controlled-$R_Y(\theta)$ operator is employed by two CNOT gates and two half rotations. We use the linear-depth method proposed in Ref. [31] to decompose the multi-qubit control operators into standard CNOT and one-qubit gates. The CNOT cost of an n-qubit controlled operator is $8n - 12$ for all $n \geq 2$. The circuit needs $2(2n - k + 1)(8k - 12)$ CNOT gates for each k-qubit controlled operation. For $k = 1$, the CNOT cost is $2n$. To implement the circuit for an arbitrary rank-n, we need to employ the modular sub-circuit shown in Figures 5 and 6 on $2n - 1$ ancilla qubits. The upper bound for the number of control qubits used in any module is $2n - 1$. The total cost of running the PREPARE subroutine is then $2n + 2\sum_{k=2}^{2n-1}(8k - 12)(2n + 1 - k)$, which can be further simplified to $\frac{8}{3}(8n^3 - 6n^2 - \frac{41n}{4} + 9)$. Hence, the cost scales like the cube of the rank.

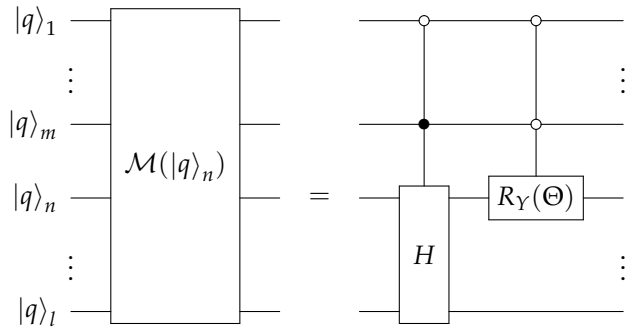

Figure 5. Quantum circuit for the modules of qubit-$|q\rangle_n$ used in the PREPARE subroutine. The Hadamard operator is anticontrolled by the qubits between $|q\rangle_1$ and $|q\rangle_m$ and controlled by the qubit $|q\rangle_m$. The $R_Y(\Theta)$ operator acting on the n-th input is anticontrolled by qubits $|q\rangle_1$ to $|q\rangle_m$. The circuit on the right defines the circuit block $\mathcal{M}$ on the left.

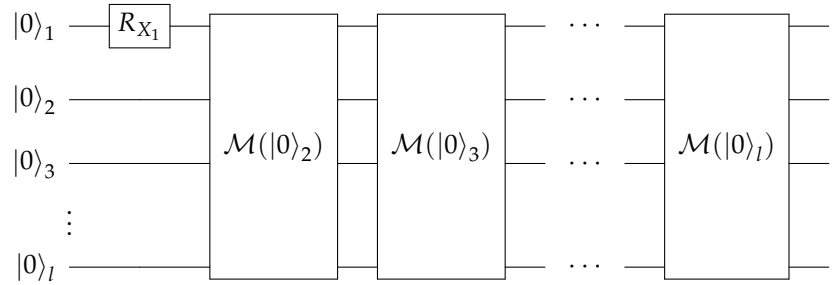

Figure 6. Quantum circuit for the PREPARE subroutine of arbitrary rank with modules introduced in Figure 5.

3.2. SELECT($\hat{U}$) Subroutine for Rank-2 Factors

In this section, we introduce a quantum circuit for implementing the SELECT($\hat{U}$) operation, illustrated in Equation (24), for the UCCD factors, where the unitary $\hat{U}$ is the operator given in Equation (22). The first step of the SELECT($\hat{U}$) circuit is to create one of the Pauli strings from the pool of the excitation operators (XY strings) and one of the Pauli strings from the projection pool (IZ strings). In this example, we opt to create the $Y_l X_k X_j X_i$ and $I_l Z_k Z_j I_i$ strings as shown in Figure 7; however, any arbitrary Pauli string can be the candidate for this step. It is important to note that we use control operations for the $Y_l X_k X_j X_i$ terms, whereas we use anticontrol operations for the $I_l Z_k Z_j I_i$ terms. This is because we are partitioning the ancilla bank into two sectors, one part for the XY strings, and one part for the IZ strings. In this case, ancilla-bank states $|1000\rangle$ to $|1111\rangle$ are used to create the XY strings, and ancilla-bank states $|0000\rangle$ to $|0111\rangle$ are used to create the IZ strings. The resulting state, omitting the corresponding coefficients, which are prepared in the previous step, becomes

$$\begin{aligned}&(|0000\rangle + \cdots + |0111\rangle)|Z_k Z_j\rangle \\ &+(|1000\rangle + \cdots + |1111\rangle) \bigotimes_{a=l+1}^{k-1} Z_a \bigotimes_{b=j+1}^{i-1} Z_b |Y_l X_k X_j X_i\rangle\end{aligned} \tag{33}$$

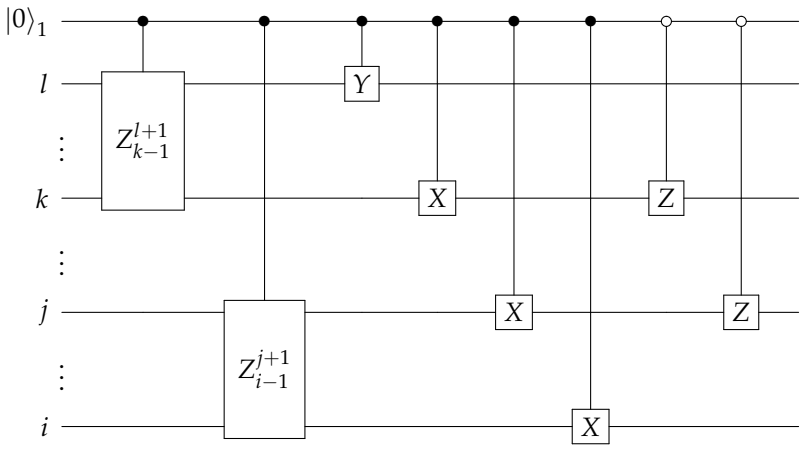

Figure 7. Circuit to create $Y_l X_k X_j X_i$ and $I_l Z_k Z_j I_i$. $|0\rangle_1$ denotes the first qubit of the ancilla bank. The first two circuit components are the controlled Pauli Z gates applied on qubits between (exclusively) indices l and k, and between indices j and i.

The construction of the controlled Pauli Z gates shown in Figure 7 is described next. With the starting reference states prepared, we can then create the entire state exactly. The first step is to apply a single-qubit controlled Pauli Z operator on qubits l and k, where the control is to be conditioned on the last qubit of the ancilla bank. The second step is similar, in that a single-qubit controlled Pauli Z operator is applied on qubits j and i. The control qubit of the second step is the second to last qubit of the ancilla bank. The final step is to apply the single-qubit controlled Pauli Z on qubits k and j, with the control qubit being the second qubit of the ancilla bank. The circuit diagram is illustrated in Figure 8. The Pauli strings and their corresponding states in the ancilla bank are shown in Table 2. The qubits on which the control operations are conditioned are chosen specifically for this table. In practice, when applying this algorithm, one needs to predetermine a table similar to Table 2 for all the binary encodings and their corresponding Pauli substrings, and choose accordingly the starting reference states and the control qubits to be used in the approach illustrated shown in Figures 7–9.

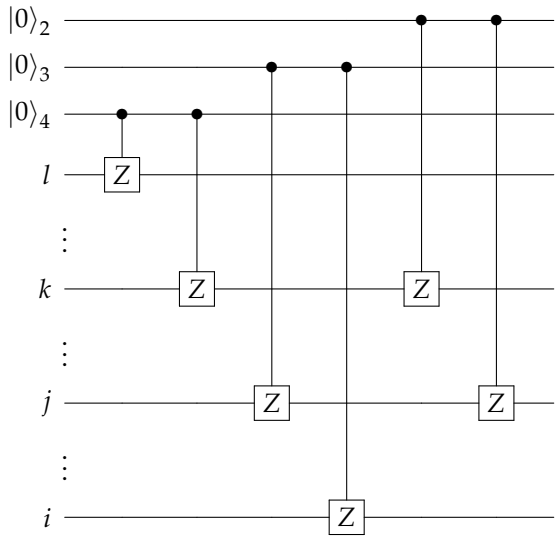

Figure 8. Circuit to create the state shown in Equation (22). $|0\rangle_i$ denotes the ith qubit of the ancilla bank.

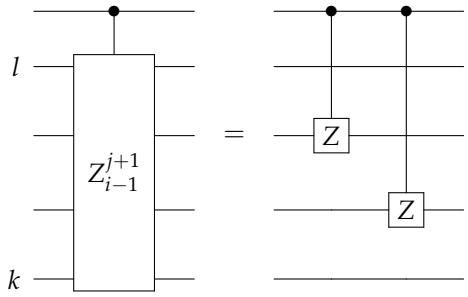

Figure 9. The circuit for implementing the boxed controlled Pauli Z operators shown in Figure 7. The Pauli Z operators are being applied to the qubits $l+1$ to $k-1$.

Table 2. The 16 Pauli strings created by scheme shown in Figure 7 and then by the circuit shown in Figure 8 and their associated binary encodings in the ancilla bank.

$\|000\rangle_{2,3,4}$	$\|0\rangle_1 = 1$	$\|0\rangle_1 = 0$
$\|000\rangle$	$YXXX$	$IZZI$
$\|001\rangle$	$XYXX$	$ZIZI$
$\|010\rangle$	$YXYY$	$IZIZ$
$\|011\rangle$	$XYYY$	$ZIIZ$
$\|100\rangle$	$YYYX$	$IIII$
$\|101\rangle$	$XXYX$	$ZZII$
$\|110\rangle$	$YYXY$	$IIZZ$
$\|111\rangle$	$XXXY$	$ZZZZ$

3.3. SELECT$(\hat{U})$ *for Arbitrary Rank-n*

In this section, we demonstrate the algorithm for the rank-n UCC factor buy generalizing the algorithms shown in Sections 3.1 and 3.2. First, let us re-examine the case for rank-2; that is, the doubles. Define groups $\mathbf{G_1} = \{G_{11}, G_{12}\}$ and $\mathbf{G_2} = \{G_{21}, G_{22}\}$, with elements $G = \sigma_i \otimes \sigma_j$, where σ_i and σ_j are two different Pauli operators acting on different qubits. Additionally, the group elements have the following identities

$$G_{11} \cdot (\sigma_z \otimes \sigma_z) = G_{12}, G_{12} \cdot (\sigma_z \otimes \sigma_z) = G_{11} \tag{34}$$

$$G_{11} \cdot (\mathbb{1} \otimes \sigma_z) = G_{22}, G_{11} \cdot (\sigma_z \otimes \mathbb{1}) = G_{21} \tag{35}$$

$$G_{21} \cdot (\sigma_z \otimes \sigma_z) = G_{22}, G_{22} \cdot (\sigma_z \otimes \sigma_z) = G_{21} \tag{36}$$

$$G_{21} \cdot (\mathbb{1} \otimes \sigma_z) = G_{12}, G_{21} \cdot (\sigma_z \otimes \mathbb{1}) = G_{11}, \tag{37}$$

where σ_z is the Pauli Z operator being applied on qubit a_1. It should be clear that the expression in Equation (22) is of the schematic form $\sum_{p\neq q}\sum_r G_{pr} \otimes G_{qr} + G'_{rp} \otimes G'_{rq}$, omitting the coefficients, where the **G** groups contain XY subterms and the **G**′ contain IZ subterms. In the example shown in the Sections 3.1 and 3.2, the corresponding groups are $\mathbf{G_1} = \{G_{11} = YX, G_{12} = XY\}$, $\mathbf{G_2} = \{G_{21} = XX, G_{22} = YY\}$, $\mathbf{G'_1} = \{G'_{11} = IZ, G'_{12} = ZI\}$, and $\mathbf{G'_2} = \{G'_{21} = II, G'_{22} = ZZ\}$. There are in total three steps needed to create the eigenfunction; hence, we opted to use three digits for the binary encoding in the ancilla bank. The SELECT($\hat{U}$) subroutines are summarized in Tables 3 and 4.

Table 3. Matrix that shows the scheme for the XY subterms presented in Section 3.2. The entry colored in blue, $G_{11}G_{21}$, is the initial reference state. After the first step, the yellow entry $G_{12}G_{21}$ is created. The pink entries are created after the second step. The first two steps use the identities in Equations (34) and (36). The final step is to use the identities in Equations (35) and (37) to create the green block.

$\otimes$	G_{11}	G_{12}	G_{21}	G_{22}
G_{11}	0	0	$G_{11}G_{21}$	$G_{11}G_{22}$
G_{12}	0	0	$G_{12}G_{21}$	$G_{12}G_{22}$
G_{21}	$G_{21}G_{11}$	$G_{21}G_{12}$	0	0
G_{22}	$G_{22}G_{11}$	$G_{22}G_{12}$	0	0

Table 4. Matrix that shows the scheme for the IZ subterms presented in Section 3.2. The entry colored in blue, $G'_{11}G'_{12}$, is chosen to be the starting reference state. After the first step, the yellow entry $G'_{12}G'_{12}$ is created. The pink entries result after the second step. The first two steps use the identities in Equations (34) and (36). The final step is to use the identities in Equations (35) and (37) to create the entire green block.

$\otimes$	G'_{11}	G'_{12}	G'_{21}	G'_{22}
G'_{11}	$G'_{11}G'_{11}$	$G'_{11}G'_{12}$	0	0
G'_{12}	$G'_{12}G'_{11}$	$G'_{12}G'_{12}$	0	0
G'_{21}	0	0	$G'_{21}G'_{21}$	$G'_{21}G'_{22}$
G'_{22}	0	0	$G'_{22}G'_{21}$	$G'_{22}G'_{22}$

Similar to the way we define for the rank-2 factors, for rank-3, we define groups $\mathbf{G_1} = \{G_{11}, G_{12}, G_{13}, G_{14}\}$ and $\mathbf{G_2} = \{G_{21}, G_{22}, G_{23}, G_{24}\}$; therefore, the wavefunction for the UCC triples factors after the JW transformation take the form $\sum_{p\neq q}\sum_r G_{pr} \otimes G_{qr} + G'_{rp} \otimes G'_{rq}$. The set of identities for rank-3 is

$$G_{11}(\sigma_z \otimes \sigma_z \otimes \mathbb{1}) = G_{14}, G_{14}(\sigma_z \otimes \sigma_z \otimes \mathbb{1}) = G_{11} \tag{38}$$
$$G_{12}(\sigma_z \otimes \sigma_z \otimes \mathbb{1}) = G_{13}, G_{13}(\sigma_z \otimes \sigma_z \otimes \mathbb{1}) = G_{12} \tag{39}$$
$$G_{11}(\mathbb{1} \otimes \sigma_z \otimes \sigma_z) = G_{12}, G_{12}(\mathbb{1} \otimes \sigma_z \otimes \sigma_z) = G_{11} \tag{40}$$
$$G_{13}(\mathbb{1} \otimes \sigma_z \otimes \sigma_z) = G_{14}, G_{14}(\mathbb{1} \otimes \sigma_z \otimes \sigma_z) = G_{13} \tag{41}$$
$$G_{21}(\sigma_z \otimes \sigma_z \otimes \mathbb{1}) = G_{24}, G_{24}(\sigma_z \otimes \sigma_z \otimes \mathbb{1}) = G_{21} \tag{42}$$
$$G_{22}(\sigma_z \otimes \sigma_z \otimes \mathbb{1}) = G_{23}, G_{23}(\sigma_z \otimes \sigma_z \otimes \mathbb{1}) = G_{24} \tag{43}$$
$$G_{21}(\mathbb{1} \otimes \sigma_z \otimes \sigma_z) = G_{22}, G_{22}(\mathbb{1} \otimes \sigma_z \otimes \sigma_z) = G_{21} \tag{44}$$
$$G_{23}(\mathbb{1} \otimes \sigma_z \otimes \sigma_z) = G_{24}, G_{24}(\mathbb{1} \otimes \sigma_z \otimes \sigma_z) = G_{22} \tag{45}$$
$$G_{11}(\mathbb{1} \otimes \mathbb{1} \otimes \sigma_z) = G_{21}, G_{12}(\mathbb{1} \otimes \mathbb{1} \otimes \sigma_z) = G_{22} \tag{46}$$
$$G_{13}(\mathbb{1} \otimes \mathbb{1} \otimes \sigma_z) = G_{23}, G_{14}(\mathbb{1} \otimes \mathbb{1} \otimes \sigma_z) = G_{24} \tag{47}$$
$$G_{21}(\sigma_z \otimes \mathbb{1} \otimes \mathbb{1}) = G_{13}, G_{22}(\sigma_z \otimes \mathbb{1} \otimes \mathbb{1}) = G_{14} \tag{48}$$
$$G_{23}(\sigma_z \otimes \mathbb{1} \otimes \mathbb{1}) = G_{11}, G_{24}(\sigma_z \otimes \mathbb{1} \otimes \mathbb{1}) = G_{12}. \tag{49}$$

As shown in the Tables 5 and 6, there are in total five steps needed to create the exact JW-transformed unitary, and thus we use five out of six digits for the binary encoding in the ancilla bank. Table 7 shows that the total number of steps needed to fully create the unitary is seven.

Table 5. Matrix illustrating the scheme for constructing XY substrings in the JW-transformed unitary of the UCCT factors. The blue entry is chosen to be the starting reference state. After the first step using identity (38), grey entry $G_{14}G_{21}$ is created. Using the identities in Equations (39) and (40), the yellow entries are created after the second step. The orange and pink columns are created after the next two steps via the identities in Equations (41) through (44). The final step is to apply the identities in Equations (46) through (49) to create the entire green block.

$\otimes$	G_{11}	G_{12}	G_{13}	G_{14}	G_{21}	G_{22}	G_{23}	G_{24}
G_{11}	0	0	0	0	$G_{11}G_{21}$	$G_{11}G_{22}$	$G_{11}G_{23}$	$G_{11}G_{24}$
G_{12}	0	0	0	0	$G_{12}G_{21}$	$G_{12}G_{22}$	$G_{12}G_{23}$	$G_{12}G_{24}$
G_{13}	0	0	0	0	$G_{13}G_{21}$	$G_{13}G_{22}$	$G_{13}G_{23}$	$G_{13}G_{24}$
G_{14}	0	0	0	0	$G_{14}G_{21}$	$G_{14}G_{22}$	$G_{14}G_{23}$	$G_{14}G_{24}$
G_{21}	$G_{21}G_{11}$	$G_{21}G_{12}$	$G_{21}G_{13}$	$G_{21}G_{14}$	0	0	0	0
G_{22}	$G_{22}G_{11}$	$G_{22}G_{12}$	$G_{22}G_{13}$	$G_{22}G_{14}$	0	0	0	0
G_{23}	$G_{23}G_{11}$	$G_{23}G_{12}$	$G_{23}G_{13}$	$G_{23}G_{14}$	0	0	0	0
G_{24}	$G_{24}G_{11}$	$G_{24}G_{12}$	$G_{24}G_{13}$	$G_{24}G_{14}$	0	0	0	0

Table 6. Matrix illustrating the scheme for constructing IZ substrings in the JW-transformed unitary of the UCCT factors. The blue entry is the starting reference state. After the first step using the identity in Equation (38), the grey entry $G_{14}G_{14}$ is created. Using the identities in Equations (39) and (40), the yellow entries are created after the second step. The orange and pink columns are created after the next two steps via the identities in Equations (41) through (44). The final step is to apply the identities in Equations (46) through (49) to create the entire green block.

$\otimes$	G'_{11}	G'_{12}	G'_{13}	G'_{14}	G'_{21}	G'_{22}	G'_{23}	G'_{24}
G'_{11}	$G'_{11}G'_{11}$	$G'_{11}G'_{12}$	$G'_{11}G'_{13}$	$G'_{11}G'_{14}$	0	0	0	0
G'_{12}	$G'_{12}G'_{11}$	$G'_{12}G'_{12}$	$G'_{12}G'_{13}$	$G'_{12}G'_{14}$	0	0	0	0
G'_{13}	$G'_{13}G'_{11}$	$G'_{13}G'_{23}$	$G'_{13}G'_{13}$	$G'_{13}G'_{14}$	0	0	0	0
G'_{14}	$G'_{14}G'_{11}$	$G'_{14}G'_{12}$	$G'_{14}G'_{13}$	$G'_{14}G'_{14}$	0	0	0	0
G'_{21}	0	0	0	0	$G'_{21}G'_{21}$	$G'_{21}G'_{22}$	$G'_{21}G'_{23}$	$G'_{21}G'_{24}$
G'_{22}	0	0	0	0	$G'_{22}G'_{21}$	$G'_{22}G'_{22}$	$G'_{22}G'_{23}$	$G'_{22}G'_{24}$
G'_{23}	0	0	0	0	$G'_{23}G'_{21}$	$G'_{23}G'_{22}$	$G'_{23}G'_{23}$	$G'_{23}G'_{24}$
G'_{24}	0	0	0	0	$G'_{24}G'_{21}$	$G'_{24}G'_{22}$	$G'_{24}G'_{23}$	$G'_{24}G'_{24}$

Table 7. Off-diagonal block of a matrix illustrating the scheme for constructing XY substrings in the JW-transformed unitary of the UCCQ factors. The blue entry of the first column is chosen to be the starting reference state. The grey entry is created after the first step, followed by the green entries in the second step and the yellow entries in the third step. The pink entries of the last column are created after the fourth step, followed by the orange columns in the fifth step and purple columns in the sixth step.

$\otimes$	G_{21}	G_{22}	G_{23}	G_{24}	G_{25}	G_{26}	G_{27}	G_{28}
G_{11}	$G_{11}G_{21}$	$G_{11}G_{22}$	$G_{11}G_{23}$	$G_{11}G_{24}$	$G_{11}G_{25}$	$G_{11}G_{26}$	$G_{11}G_{27}$	$G_{11}G_{28}$
G_{12}	$G_{12}G_{21}$	$G_{12}G_{22}$	$G_{12}G_{23}$	$G_{12}G_{24}$	$G_{12}G_{25}$	$G_{12}G_{26}$	$G_{12}G_{27}$	$G_{12}G_{28}$
G_{13}	$G_{13}G_{21}$	$G_{13}G_{22}$	$G_{13}G_{23}$	$G_{13}G_{24}$	$G_{13}G_{25}$	$G_{13}G_{26}$	$G_{13}G_{27}$	$G_{13}G_{28}$
G_{14}	$G_{14}G_{21}$	$G_{14}G_{22}$	$G_{14}G_{23}$	$G_{14}G_{24}$	$G_{14}G_{25}$	$G_{14}G_{26}$	$G_{14}G_{27}$	$G_{14}G_{28}$
G_{15}	$G_{15}G_{21}$	$G_{15}G_{22}$	$G_{15}G_{23}$	$G_{15}G_{24}$	$G_{15}G_{25}$	$G_{15}G_{26}$	$G_{15}G_{27}$	$G_{15}G_{28}$
G_{16}	$G_{16}G_{21}$	$G_{16}G_{22}$	$G_{16}G_{23}$	$G_{16}G_{24}$	$G_{16}G_{25}$	$G_{16}G_{26}$	$G_{16}G_{27}$	$G_{16}G_{28}$
G_{17}	$G_{17}G_{21}$	$G_{17}G_{22}$	$G_{17}G_{23}$	$G_{17}G_{24}$	$G_{17}G_{25}$	$G_{17}G_{26}$	$G_{17}G_{27}$	$G_{17}G_{28}$
G_{18}	$G_{18}G_{21}$	$G_{18}G_{22}$	$G_{18}G_{23}$	$G_{18}G_{24}$	$G_{18}G_{25}$	$G_{18}G_{26}$	$G_{18}G_{27}$	$G_{18}G_{28}$

For a UCC factor with arbitrary rank n, a total number of $n-1$ transformations are required to complete the first column of the matrix. An additional $n-1$ transformations

are then required to complete the rest of the diagonal or off-diagonal block, depending on whether the subterms are XY strings or IZ strings. The final step is to perform the 'flip' transformation to make the entire matrix. Therefore, for any UCC factor of rank n, a total number of $2n-1$ steps are needed to construct the JW-transformed unitary operator.

3.4. Gate Counts

The LCU framework for rank-n UCC factors is given by a simple circuit implementation of the JW-transformed unitary in the form of

$$\sum_{p\neq q}\sum_{r} G_{pr}\otimes G_{qr} + G'_{rp}\otimes G'_{rq} \tag{50}$$

where each group $\mathbf{G_p}$ contains 2^{n-1} elements that commute with every other element in the same group but anticommute with elements in the other group. The total number of steps in the SELECT($\hat{U}$) subroutine is $2n-1$ for rank-n factors. Within each step of the SELECT($\hat{U}$) subroutine, two single-qubit controlled Pauli Z operators are needed, making the total number of CNOT gates $4n-2$. In addition, $4n+\sum_i^{2n-2}\rho_i$ CNOT gates are needed for initializing the reference Jordan–Wigner strings. Here, ρ_i is the number of qubits between the ith pair of the active orbitals. In the case of a UCCD factor discussed in Sections 3.1 and 3.2, ρ_1 is the number of qubits between qubits k and l and ρ_2 is the number of qubits between qubits j and i. The circuit for preparing the ancilla bank hosts the majority of the complexity, where the total CNOT cost is $\frac{8}{3}(8n^3-6n^2-\frac{41n}{4}+9)$. The total number of CNOT gates used in the LCU circuit for preparing the JW-transformed unitary is then $6\cdot(\frac{8}{3}(8n^3-6n^2-\frac{41n}{4}+9))+3\cdot(8n-2+\sum_i^{2n-2}\rho_i)$, which can be further simplified to $128n^3-96n^2-140n+138+3\sum_i^{2n-2}\rho_i$. Note that this has a large prefactor in the scaling with the rank. The total number of ancilla qubits required for this framework is $2n$. For a UCC factor with an arbitrary set of active orbitals, the CNOT cost of the circuits proposed in our work eventually becomes favorable compared with other existing methods when the rank becomes large ($n\geq 9$), including the one proposed in the authors' previous work [20], although for low-rank factors, the fermionic-excitation-based (FEB) algorithm proposed by Ref. [26] is more efficient, which is shown in Figure 10.

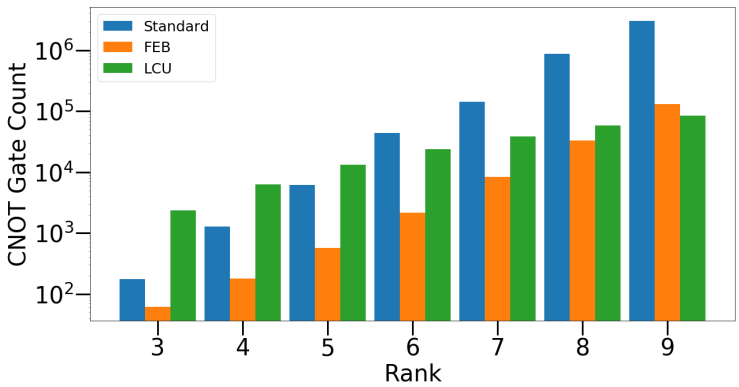

Figure 10. CNOT gate counts for three different algorithms, the standard CNOT cascading circuits, the FEB circuits [26], and the linear-combination-of-unitary query circuits introduced in this work.

4. Discussion

In summary, we have introduced an application of the LCU query model that efficiently simulates the factorized UCC ansatze with a scaling that goes like the cube of the rank. We have demonstrated the quantum circuits for the PREPARE and the SELECT($\hat{U}$) subroutines, whose CNOT counts scale linearly with the rank of the UCC ansatze and the number of active spin-orbitals. The PREPARE operator employs a quantum circuit that scales as the cube of the rank of the UCC factor and requires a linear number of ancilla. The proposed LCU framework greatly reduces the number of two-qubit gates for high-rank

UCC factors, which are needed for simulating strongly correlated systems on NISQ devices and future fault-tolerant quantum computers [28]. Although the UCC ansatzes that contain only singles and doubles with some triples treated perturbatively are considered the golden standard in quantum chemistry calculations, Refs. [32,33] demonstrated that rank-17 excitations are needed to obtain accurate results for systems such as the chromium dimer. Further, it is expected that strongly correlated systems will require higher rank terms and these are precisely the problems proposed for quantum computers.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/sym15071429/s1.

Author Contributions: Conceptualization, L.X. and J.K.F.; methodology, L.X.; validation, L.X. and J.K.F.; formal analysis, L.X. and J.K.F.; investigation, L.X. and J.K.F.; resources, J.K.F.; data curation, L.X.; writing—original draft preparation, L.X. and J.K.F.; writing—review and editing, L.X. and J.K.F.; visualization, L.X.; supervision, J.K.F.; project administration, J.K.F.; funding acquisition, J.K.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the U. S. Department of Energy grant number ERKJ347.

Data Availability Statement: The data presented in this study are openly available in Supplementary Materials.

Acknowledgments: We acknowledge helpful discussions with Joseph T. Lee, L. Xu, and J. K. Freericks were supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research (ASCR), Quantum Computing Application Teams (QCATS) program, under field work proposal number ERKJ347. J. K. Freericks was also supported by the McDevitt bequest at Georgetown University.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Aspuru-Guzik, A. Simulated Quantum Computation of Molecular Energies. *Science* **2005**, *309*, 1704–1707. [CrossRef]
2. Whitfield, J.D.; Biamonte, J.; Aspuru-Guzik, A. Simulation of electronic structure Hamiltonians using Quantum Computers. *Mol. Phys.* **2011**, *109*, 735–750. [CrossRef]
3. Sherrill, C.D.; Schaefer, H.F. The Configuration Interaction Method: Advances in Highly Correlated Approaches. *Adv. Quantum Chem.* **1999**, *34*, 143–269. [CrossRef]
4. Bartlett, R.J.; Purvis, G.D. Many-body perturbation theory, coupled-pair many-electron theory, and the importance of quadruple excitations for the correlation problem. *Int. J. Quantum Chem.* **1978**, *14*, 561–581. [CrossRef]
5. Bartlett, R.J.; Musiał, M. Coupled-cluster theory in quantum chemistry. *Rev. Mod. Phys.* **2007**, *79*, 291–352. [CrossRef]
6. Purvis, G.D.; Bartlett, R.J. A full coupled-cluster singles and doubles model: The inclusion of disconnected triples. *J. Chem. Phys.* **1982**, *76*, 1910–1918. [CrossRef]
7. Anand, A.; Schleich, P.; Alperin-Lea, S.; Jensen, P.W.; Sim, S.; Díaz-Tinoco, M.; Kottmann, J.S.; Degroote, M.; Izmaylov, A.F.; Aspuru-Guzik, A.; et al. A quantum computing view on unitary coupled cluster theory. *Chem. Soc. Rev.* **2022**, *51*, 1659–1684. [CrossRef]
8. Peruzzo, A.; Mcclean, J.; Shadbolt, P.; Yung, M.H.; Zhou, X.Q.; Love, P.J.; Aspuru-Guzik, A.; O'Brien, J.L. A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **2014**, *5*, 4213. [CrossRef]
9. Preskill, J. Quantum Computing in the NISQ era and beyond. *Quantum* **2018**, *2*, 79. [CrossRef]
10. Helgaker, T.J. *Molecular Electronic-Structure Theory*; Wiley: Hoboken, NJ, USA, 2014.
11. Szabo, A.; Ostlund, N.S. *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory*; Dover Publications: Mineola, NY, USA, 2006.
12. Taketa, H.; Huzinaga, S.; O-Ohata, K. Gaussian-Expansion Methods for Molecular Integrals. *J. Phys. Soc. Jpn.* **1966**, *21*, 2313–2324. [CrossRef]
13. Shavitt, I.; Bartlett, R.J. *Many-Body Methods in Chemistry and Physics: MBPT and Coupled-Cluster Theory*; Cambridge University Press: Cambridge, MA, USA, 2009. [CrossRef]
14. Bartlett, R.J.; Kucharski, S.A.; Noga, J. Alternative coupled-cluster ansätze II. The unitary coupled-cluster method. *Chem. Phys. Lett.* **1989**, *155*, 133–140. [CrossRef]
15. Schaefer, H.F. *Methods of Electronic Structure Theory*; Springer Science Business Media, LLC: New York, NY, USA, 2013. [CrossRef]
16. Cooper, B.; Knowles, P.J. Benchmark studies of variational, unitary and extended coupled cluster methods. *J. Chem. Phys.* **2010**, *133*, 234102. [CrossRef] [PubMed]
17. Chen, J.; Cheng, H.P.; Freericks, J.K. Quantum-Inspired Algorithm for the Factorized Form of Unitary Coupled Cluster Theory. *J. Chem. Theory Comput.* **2021**, *17*, 841–847. [CrossRef]

18. Romero, J.; Babbush, R.; Mcclean, J.R.; Hempel, C.; Love, P.J.; Aspuru-Guzik, A. Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz. *Quantum Sci. Technol.* **2018**, *4*, 014008. [CrossRef]
19. Barkoutsos, P.K.; Gonthier, J.F.; Sokolov, I.; Moll, N.; Salis, G.; Fuhrer, A.; Ganzhorn, M.; Egger, D.J.; Troyer, M.; Mezzacapo, A.; et al. Quantum algorithms for electronic structure calculations: Particle-hole Hamiltonian and optimized wave-function expansions. *Phys. Rev. A* **2018**, *98*, 022322. [CrossRef]
20. Xu, L.; Lee, J.T.; Freericks, J.K. Decomposition of high-rank factorized unitary coupled-cluster operators using ancilla and multiqubit controlled low-rank counterparts. *Phys. Rev. A* **2022**, *105*. [CrossRef]
21. Nielsen, M.A.; Chuang, I.L. *Quantum Computation and Quantum Information*; Cambridge University Press: Cambridge, MA, USA, 2019.
22. Motta, M.; Ye, E.; McClean, J.R.; Li, Z.; Minnich, A.J.; Babbush, R.; Chan, G.K.L. Low rank representations for quantum simulation of electronic structure. *NPJ Quantum Inf.* **2021**, *7*, 83. [CrossRef]
23. Wan, K. Exponentially faster implementations of select(h) for Fermionic Hamiltonians. *Quantum* **2021**, *5*, 380. [CrossRef]
24. Wang, Q.; Li, M.; Monroe, C.; Nam, Y. Resource-optimized fermionic local-Hamiltonian simulation on a quantum computer for Quantum Chemistry. *Quantum* **2021**, *5*, 509. [CrossRef]
25. Cowtan, A.; Dilkes, S.; Duncan, R.; Simmons, W.; Sivarajah, S. Phase gadget synthesis for shallow circuits. *Electron. Proc. Theor. Comput. Sci.* **2020**, *318*, 213–228. [CrossRef]
26. Magoulas, I.; Evangelista, F.A. CNOT-efficient circuits for arbitrary rank many-body fermionic and qubit excitations. *J. Chem. Theory Comput.* **2023**, *19*, 822–836. [CrossRef]
27. Xu, L.; Lee, J.T.; Freericks, J.K. Test of the unitary coupled-cluster variational quantum eigensolver for a simple strongly correlated condensed-matter system. *Mod. Phys. Lett. B* **2020**, *34*, 2040049. [CrossRef]
28. Freericks, J.K. Operator relationship between conventional coupled cluster and unitary coupled cluster. *Symmetry* **2022**, *14*, 494. [CrossRef]
29. Childs, A.M.; Wiebe, N. Hamiltonian simulation using linear combinations of unitary operations. *Quantum Inf. Comput.* **2012**, *12*, 901–924. [CrossRef]
30. Berry, D.W.; Childs, A.M.; Cleve, R.; Kothari, R.; Somma, R.D. Simulating Hamiltonian dynamics with a truncated Taylor series. *Phys. Rev. Lett.* **2015**, *114*, 090502. [CrossRef] [PubMed]
31. da Silva, A.J.; Park, D.K. Linear-depth quantum circuits for multiqubit controlled gates. *Phys. Rev. A* **2022**, *106*, 042602. [CrossRef]
32. Li, J.; Yao, Y.; Holmes, A.A.; Otten, M.; Sun, Q.; Sharma, S.; Umrigar, C.J. Accurate many-body electronic structure near the basis set limit: Application to the chromium dimer. *Phys. Rev. Res.* **2020**, *2*, 012015. [CrossRef]
33. Larsson, H.R.; Zhai, H.; Umrigar, C.J.; Chan, G.K.L. The chromium dimer: Closing a chapter of Quantum Chemistry. *J. Am. Chem. Soc.* **2022**, *144*, 15932–15937. [CrossRef]