Fragment-based Excitonic Coupled-Cluster Theory for Large Chemical Systems

Yuhong Liu
University of the Pacific, yliu556@gmail.com

Follow this and additional works at: https://scholarlycommons.pacific.edu/uop_etds
Part of the Chemistry Commons, and the Pharmacy and Pharmaceutical Sciences Commons

Recommended Citation
FRAGMENT-BASED EXCITONIC COUPLED-CLUSTER THEORY FOR LARGE CHEMICAL SYSTEMS

by

Yuhong Liu

A Dissertation Submitted to the
Office of Research and Graduate Studies
In Partial Fulfillment of the
Requirements for the Degree of
DOCTOR OF PHILOSOPHY

Thomas J. Long School of Pharmacy and Health Sciences
Pharmaceutical and Chemical Sciences

University of the Pacific
Stockton, CA

2017
FRAGMENT-BASED EXCITONIC COUPLED-CLUSTER THEORY FOR LARGE CHEMICAL SYSTEMS

by

Yuhong Liu

PROVED BY:

Dissertation Advisor: Anthony D. Dutoi, Ph.D.

Committee Member: Jianhua Ren, Ph.D.

Committee Member: Charles M. McCallum, Ph.D.

Committee Member: Balint Sztaray, Ph.D.

Committee Member: Elisa Toloba, Ph.D.

Department Chair: Jianhua Ren, Ph.D.

Dean of Graduate Studies: Thomas H. Naehr, Ph.D.
FRAGMENT-BASED EXCITONIC COUPLED-CLUSTER THEORY FOR LARGE CHEMICAL SYSTEMS

Copyright 2017

by

Yuhong Liu
DEDICATION

For my parents, Ying Hong and Wei Liu.
ACKNOWLEDGMENTS

My gratitude goes to Dr. Anthony D. Dutoi for his countless hours of patience and guidance while I struggled through building the excitonic coupled-cluster code. His efforts to make me a responsible scientist hopefully will be fruitful. Dr. C. Mike McCallum, Dr. Hyun Joo and Dr. Balint Sztaray have been, and will continue to be, an inspiration to me; Dr. Andreas Franz and Ms. Susan McCann are always very helpful throughout my time at Pacific. I would like to thank my parents for their unconditional support.
Accurate energetic modeling of large molecular systems is always desired by chemists. For example, ligand-protein binding simulations and enzymatic catalysis studies all involve with a small energy difference. The energetic accuracy depends largely on a proper handling of electronic correlations. Molecular mechanics (MM) methods deliver a parameterized Newtonian treatment to these problems. They show great capability in handling large calculations but give only qualitatively good results. Quantum mechanics (QM) methods solve Schrödinger equations and exhibit much better energy accuracy, though the computational cost can be prohibitive if directly applied to very large systems.
Fragment-based methods have been developed to decompose large QM calculations into fragment calculations. However, most current schemes use a self-consistent field (SCF) method on fragments, in which no electronic correlation is accounted for. The super-system energy is computed as a sum of fragment energies plus two-body corrections and, possibly, three-body corrections (a "body" is a fragment). Higher order corrections can be added.

Nevertheless, many problems require the treatment of high order electronic correlations. The coupled-cluster (CC) theory is the state-of-the-art QM method for handling electronic correlations. The CC wavefunction contains correlated excitations up to a given truncated level and coincidental excitations for all possible electronic excitations. It is a brilliant way of including more electronic correlations while maintaining a low-order scaling. In the proposed excitonic coupled-cluster (X-CC) theory, substantial modifications have been made to allow CC algorithms to act on the collective coordinates of fragment fluctuations to obtain super-system energy.

The X-CC theory is designed to achieve accurate energetic modeling results for large chemical systems with much improved affordability and systematic improvability. The test system used in this work is a chain of beryllium atoms. A 30-fragment X-CCSD(2) calculation delivered matching accuracy with traditional CCSD method. An X-CCSD(2) calculation on a chain of 100 bonded fragments finished in 7 hours on a single 2.2 GHz CPU core. The X-CC scheme also demonstrates the ability in handling charge transfer problems. Due to the use of fluctuation basis in the test cases, the excitonic algorithms can
be easily generalized to inhomogeneous systems. This will be investigated in future work.
# TABLE OF CONTENTS

LIST OF FIGURES........................................................................................................12

LIST OF ABBREVIATIONS.............................................................................................13

CHAPTER

1. Introduction..............................................................................................................14
   Introduction to Computational Chemistry Methods..............................................14
   Recoupled-System Intuition....................................................................................30
   Detailed Exposition of Coupled-Cluster Theory..................................................34

2. Derivation of the Excitonic Coupled-Cluster Theory............................................44
   General Notation.....................................................................................................44
   Fluctuation Operators............................................................................................45
   Coupled-Cluster Ansatz..........................................................................................46
   Hamiltonian............................................................................................................47

3. Excitonic Coupled-Cluster Theory for Quantum Harmonic Oscillator Model..........................................................50
   One-Dimensional Quantum Harmonic Oscillator.................................................50
General Linearly Coupled Harmonic Oscillators and Solutions.....52

Fragment and Super-Systems.............................................55

Numerical Solutions to the Recoupled Problems.....................56

Conclusions........................................................................68

4. Excitonic Coupled-Cluster Theory in Electronic Systems........69

Background......................................................................69

Conventions and Notations................................................74

The One-Electron Basis....................................................75

Many-Electron Basis........................................................76

Single-Fragment Fluctuation Operators...............................81

The Hamiltonian in terms of Fragment Fluctuation Operators....84

Practical Considerations....................................................90

Conclusions......................................................................99

5. Excitonic Coupled-Cluster Theory Initial Tests: Beryllium Atoms....101

Beryllium Bonding Problem..............................................101

Fragmentation and Fragment Eigenstates............................103

Decomposition of Dimer States into Monomer States............104
Dimer Interaction Terms..................................................................................110

Optimal Monomer States..............................................................................112

Results and Discussion................................................................................113

6. Outlook............................................................................................................118

REFERENCES........................................................................................................120

APPENDICES

A. X-CCSD(2) AMPLITUDE EQUATIONS..........................................................131

B. GENERAL DISCUSSION OF AUTOMATIC CODE GENERATION............143

C. DIRECT INVERSION OF ITERATIVE SUBSPACE ACCELERATION.............149

D. PYTHON AND C INTERFACE........................................................................151

E. PARALLEL COMPUTING WITH PYTHON AND OPENMP.........................160
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>31</td>
</tr>
<tr>
<td>2.</td>
<td>51</td>
</tr>
<tr>
<td>3.</td>
<td>62</td>
</tr>
<tr>
<td>4.</td>
<td>63</td>
</tr>
<tr>
<td>5.</td>
<td>66</td>
</tr>
<tr>
<td>6.</td>
<td>102</td>
</tr>
<tr>
<td>7.</td>
<td>106</td>
</tr>
<tr>
<td>8.</td>
<td>106</td>
</tr>
<tr>
<td>9.</td>
<td>107</td>
</tr>
<tr>
<td>10.</td>
<td>107</td>
</tr>
<tr>
<td>11.</td>
<td>114</td>
</tr>
<tr>
<td>12.</td>
<td>114</td>
</tr>
<tr>
<td>13.</td>
<td>115</td>
</tr>
<tr>
<td>14.</td>
<td>116</td>
</tr>
</tbody>
</table>

1. Primitive and collective interactions
2. Wavefunction of a quantum harmonic oscillator
3. Excitonic FCI method accuracy
4. Excitonic FCI method onto different oscillator molecules
5. Excitonic CCSD method tested on groups of harmonic oscillators
6. The Be dimer dissociation potential modeled by FCI method with different basis sets
7. Unitary matrix of Be atom HF orbitals
8. $C_{\text{Be in Be}_2}$ matrix
9. Unitary matrix of Be$_2$ HF orbitals
10. Molecular orbital overlap matrix for beryllium atom and beryllium dimer
11. Be dimer dissociation potential modeled by X-CCSD(2), FCI, CCSD and CCSD(T) methods
12. The Be dimer dissociation potential modeled by X-CCSD(2) method without charge transfer
13. The Be trimer dissociation potential modeled by X-CCSD(2) and other methods
14. The scaling comparisons between X-CCSD(2) method and traditional CCSD and CCSD(T) methods
LIST OF ABBREVIATIONS

Abbreviations

1. MM: Molecular mechanics
2. MD: Molecular dynamics
3. QM: Quantum mechanics
4. SCF: Self-consistent field
5. HF: Hartree-Fock
6. DFT: Density functional theory
7. PT: Perturbation theory
8. CI: Configuration interaction
9. FCI: Full configuration interaction
10. CC: Coupled-cluster theory
11. CCSD: Coupled-cluster theory with single and double excitations
12. CCSD(T): Coupled-cluster theory with single, double and perturbative triple excitations
13. X-CC: Excitonic coupled-cluster theory
14. X-CCSD: Excitonic coupled-cluster theory with single and double excitations
Chapter 1: Introduction

Introduction to Computational Chemistry Methods

Chemical reactions are central to our understanding of the chemical world. One important aspect of the understanding is to predict the likelihood for a reaction to occur. Thermodynamically such likelihood can be described by using the reaction equilibrium constant, $K$. The equilibrium constant, from the Arrhenius equation, is known to be exponentially dependent on the energy difference of the systems before and after the chemical reaction. However, this energy difference is relatively small compared to the system total energy and requires accurate molecular modeling methods to be applied. Furthermore, the bond-breaking process, which is common in chemical reactions, involves with electrons being moved from bonding orbitals into anti-bonding orbitals. Other uninvolved electrons would also adjust their distributions based on the change of electronic repulsions. This correlated process is called electronic correlation. It describes how electrons would react to the changes created by other electrons. Electronic correlation is crucial to properly describe bond-breaking process and given the fact that electrons are fundamentally quantum mechanical, one needs an electronically correlated quantum mechanical method to recover electronic correlations. Many reactions in biological systems rely on transferring charges (electrons) between proteins or complex molecules, such as in the respiration chain or in the photosynthesis process. [1] Being able to properly handle electronic correlation inside a large molecular system is desired by chemists.

Currently, there are two major categories of molecular modeling methods. They are molecular mechanics (MM) methods and quantum mechanics (QM)
methods. The MM methods treat atoms as the most basic units in simulations, which lack a proper handling of electronic correlations. The quantum mechanics methods, on the other hand, start with electrons and orbitals. By solving the Schrödinger equation one can build wavefunctions based on these orbitals. In the following section, brief introductions will be given for various QM methods about their pros and cons.

**Molecular mechanics methods.** The MM methods or the molecular dynamics (MD) methods [2, 3] treat atoms as unbreakable fundamental particles. Atoms carry (partial) charges and have assigned velocities. By using the Newtonian laws of classical physics and parameterizations of well-established interactions, MM methods are able to deliver energy simulations for molecular systems. These well-established interactions include bonded interactions such as bond vibrations, bending of bond angle, dihedral torsions, hydrogen bonding and so on. For non-bonded interactions, it includes Van der Waal forces, dispersion coupling, solvent effects and other interactions. The system total energy is a summation of individual atom’s kinetic energy and potential energy plus all interaction energies. Geometry optimizations can be carried out for discovering local minimum. Time dynamics simulations are also available to analyze how atoms propagate in chemical processes.

One big advantage of MM methods is the very low computational cost or, equivalently, the ability to handle millions of atoms within one simulation but still being able to finish in practical time frames. However, the results computed by MM methods are in general only qualitatively good in terms of energy accuracy. Due to the lack of electronic correlations and the fact that being unable to handle fundamentally quantum mechanical processes such as the charge transfer process, the MM method is not an ideal approach for recovering accurate energies in large molecular systems. This can be understood from many aspects. Firstly,
parameterized equations may not work for all types of systems and the calculated energies are only as good as they are calibrated. Secondly, if not well-established interactions should exist in certain cases, for example, the halogen bonding interaction \[4, 5, 6, 7, 8\], the current MM methods cannot account for unknown interactions. Last but not the least, if one MM force-field failed to describe the modeling problem, no higher-level method can be turned to, that is, the MM method lacks proper systematic improvability. For accurate energetic modeling, one should consider the quantum mechanics methods.

**Quantum mechanics methods.** The quantum mechanics methods are also called *ab initio* methods, which means in QM calculations every interaction is automatically included and treated "from the beginning". The equations used in QM methods are derived from fundamental laws of physics. QM methods use wavefunctions to represent electron density in space and these wavefunctions are built upon orbitals. There are several most basic assumptions in QM methods. First of all, a non-relativistic treatment of the molecular systems is applied for simplicity. Then it added the Born-Oppenheimer approximation in which all nuclear motions are ignored given the fact that electrons have much higher velocities than nuclei do. The next fundamental approximation is the basis set approximation. A basis set is a set of Gaussian orbitals used in a QM calculations. Each orbital within a set is constructed by one more Gaussian functions to model electron densities at bonding ranges. If a basis set is small, then the ability to flexibly describe the shapes of molecular orbitals is limited. These three approximations are used in all QM methods.

The QM methods automatically include, to certain method limit, all known and unknown electronic interactions from the beginning, without the need for any parameterization. This can be easily understood since all sort of interaction inside a molecule are indeed electrostatic interactions at the most fundamental level.
For QM calculations, it is essentially solving the time-independent Schrödinger equation for electronic systems. A Hamiltonian is an energy operator. The electronic Hamiltonian would return the electronic energy of a molecular system if it acts on the wavefunction of that system. And it is defined as a summation of individual electron’s kinetic and nuclear attraction energy terms plus all pairs of electronic repulsion energy terms, as shown in eq. (1.1), in which all quantities are defined in atomic units for simplicity.

\[
\hat{H} = \sum_{i=1}^{n} -\frac{1}{2} \nabla_i^2 + \sum_{i=1}^{n} \sum_{A=1}^{m} -\frac{Z_A}{r_{iA}} + \sum_{i=1}^{n} \sum_{j>i}^{n} \frac{1}{r_{ij}}
\]  

(1.1)

The first term is the sum of all kinetic energy operators. The summation index \( i \) goes all the way to the total number of electrons \( n \) and the \( \nabla_i^2 \) is the Laplacian for the \( i \)-th electron. The second term contains all nuclear attraction energy operators. The summation over all nuclei is indexed by \( A \) and goes up to the total number of nuclei \( m \). Here the nuclear attraction potential is an attractive potential between electron \( i \) and nucleus \( A \) at the distance of \( r_{iA} \) affected by the number of protons \( Z_A \) of nucleus \( A \). The last term includes all electronic repulsion energy operators. It is summing over all pairs, denoted by \( i \) and \( j \), of electronic repulsions at a distance of \( r_{ij} \). The positive term indicates a repulsive potential.

\[
\varepsilon = \langle \Psi | \hat{H} | \Psi \rangle
\]

(1.2)

The expectation value \( \varepsilon \), eq. (1.2), from solving the Schrödinger equation, gives the electronic energy of a molecular system in state \( \Psi \). The \( \Psi \) here can represent the ground state or any excited states.

The first two terms of eq. (1.1) are very simple to solve since they deal with one electron at a time and the system total kinetic or nuclear attraction energy is additive in terms of electrons. However, there is no analytical solution to the electronic repulsion energy terms. Different QM methods handle electronic repulsion
(electronic correlation) in different ways but it is important to have the correlations properly described in order to get accurate energetic modeling results.

Generally speaking, QM methods display much better accuracy than MM methods in energy modeling. MM methods disregard electrons and electronic correlations, it cannot handle bond-breaking, transition states, charge transfer processes, multi-reference problems and many other cases that require proper treatment of electronic correlations.

QM methods are typically much more expensive than MM methods in terms of computational cost. The scaling of common MM methods is formally quadratic, but it could be reduced into linear range if special techniques are applied. The least expensive QM method, namely the self-consistent field (SCF) method, has a formal scaling of $O(N^4)$ where the $N$ is the number of electrons. This is due to the four-index quantity coming from the two-electron repulsion operator. The SCF method has been largely optimized with thresholding, fittings and other techniques so that the practical scaling is almost linear. But QM methods still execute substantially slower than MM methods. This is due to the nature of different schemes. MM equations require one-time computation using analytical formulas dealing with atoms. However, the SCF and some high-level electronically correlated QM methods perform iterative optimizations on electronic systems, and they carry out much more complicated procedures within each iteration. The efficiency of MM methods is out of reach for any QM schemes if the same problem is given. High computational cost prevents QM methods from being widely applied to large molecular systems in both industrial and academic applications. Many endeavors have been made over decades to tackle this problem. The fragment-based methods are among the most successful ones. In the following sections, brief introductions to the common QM methods will be given and several important state-of-the-art fragment-based methods will be visited.
**QM/MM methods.** The quantum mechanics/molecular mechanics (QM/MM) [9, 10, 11] method is an important method for modeling large molecular systems with a moderately sized site of interests. QM/MM is an embedding scheme in which the site of interests (e.g., the enzyme active site or ligand-protein binding pocket) is treated with QM theory and is embedded in, most often, the electrostatic potential generated by MM calculations on the rest of the molecular system. Due to the high computational cost of QM methods, the size of QM region is still limited. More atoms included in the QM region would improve the modeling results but also cause the total computation time to rise. Not surprisingly, the SCF methods are typically used in the QM region to cut back computational cost. The excitonic CC theory proposed by this work is not orthogonal to the QM/MM method. The excitonic CC theory can be applied to the QM region within the scheme. Our job is to make the QM region perform better in accuracy and/or getting larger by fragment-based decomposition while retaining systematic improvability.

**Self-consistent field methods.** The SCF methods include the Hartree-Fock (HF) method and the density functional theory (DFT) [12] methods. The Hartree-Fock method iterates over all occupied orbitals and each orbital is optimized in the electrostatic mean field generated by other occupied orbitals. This procedure yields a set of new molecular orbitals to be used in the next iteration. Orbitals are repeatedly optimized for energy minimizations until convergence, that is, the set of new orbitals are self-consistent with the mean field potential which generated the these new molecular orbitals.

\[
\hat{f}\psi_a(1) = \hat{h}(1)\psi_a(1) + \sum_{b \neq a} \left[ \int dx_2 |\psi_b(2)|^2 \frac{1}{r_{12}} \right] \psi_a(1) - \sum_{b \neq a} \left[ \int dx_2 \psi_b(2)\psi_a(2) \frac{1}{r_{12}} \right] \psi_b(1)
\]

(1.3)

The Fock operator [13] is given by eq. (1.3) in a form that the operator itself is acting on the occupied molecular orbital \(a\), denoted as \(\psi_a\). The \(\psi_b\) represents any other occupied molecular orbitals except for \(a\). The \(\hat{h}(1)\) indicates the core
Hamiltonian that acts on electron 1. The core Hamiltonian is defined as the sum of kinetic energy operator and nuclear attraction energy operator. The last two terms in eq. (1.3) are called the Coulomb operator and the exchange operator, respectively. The Coulomb operator describes the electrostatic repulsive behaviors of occupied orbital \( a \) inside the mean field generated by any other occupied orbital \( b \). The exchange operator solely arose from the anti-symmetry property of fermions, which can be explained as that electrons are indistinguishable and exchanges of electrons are necessary to account all possible arrangements of electrons within occupied orbitals. There is no electronic correlation accounted from HF scheme due to the mean field treatments. Nevertheless, the result of a HF calculation serves as the starting point of electronically correlated QM methods.

The DFT methods use the same SCF iterative machinery as the HF method but with different energy formulas to account for electronic correlations. The Hohenberg-Kohn theorem states that the system total energy is a unique function of electronic density. If one is able to find the exact electronic density of a molecule, then the exact energy can be calculated. It is unclear how to get the exact density analytically, but the semi-empirical functionals performed poorly until Kohn-Sham theory came out. The Kohn-Sham equations state that the exact ground state electronic density can be written as the ground state density of a fictitious system of non-interacting particles. This gives us a set of independent particle equations (orbitals) that can be solved numerically. Semi-empirical Kohn-Sham DFT functionals perform reasonably well most of the time in modeling chemical problems.

$$E_{\text{B3LYP}} = 0.2E_{\text{HF}}^X + 0.72E_{\text{B88}}^X + 0.08E_{\text{Slater}}^X + 0.81E_{\text{LYP}}^C + 0.19E_{\text{VWN}}^C \quad (1.4)$$

In DFT Hamiltonians, the mean field and the exchange terms are replaced with some semi-empirical electronic correlation terms, e.g., eq. (1.4) is the well-known B3LYP functional [14, 15]. The B3LYP energy is a parameterized summation of HF
energy, exchange energy and several functional energies. The parameters in front of all the terms are adjusted by a best fit to certain sets of molecules. [16]

The DFT methods are the most frequently used QM methods today. They are widely applied to small to large (thousands of atoms) molecular systems in many different types of simulations, such as single point energy, geometry optimization, transition state calculations, thermal chemistry calculations, property calculations and so on. Different DFT functionals were designed for different sets of molecules by fitting to their experimental data. Therefore working experiences of different functionals can play an important role when getting the best modeling results for certain chemical species. This makes it cumbersome for beginners.

On encountering of failure cases, due to the parameterization taken in the functionals, DFT methods have no higher-level of theory that can be turned to. Therefore, the DFT methods are not systematically improvable. It is also true that no current functionals can properly handle bond-breaking or charge transfer problems as opposed to wavefunction-based correlation methods, which will be discussed shortly.

The DFT methods have the same formal scaling, $O(N^4)$, as the HF method because the functionals posses two-electron integral terms, which give rise to four-index quantities and fourth-order scaling. But they have also been extensively optimized and have achieved a linear scaling in practice. In most scenarios, a DFT method is a much better choice than a HF calculation given the comparable computational cost.

**Many-body perturbation theory.** The many-body perturbation theory (MBPT or PT) is well-known under the method name Møller-Plesset second-order (MP2) correction method among chemists. The perturbation theory rewrites the electronic Hamiltonian into a sum of the Fock operator and the fluctuation
operator, \( \lambda \hat{V} \), which works like a perturbation to the Fock operator [13], as shown in eq. (1.5).

\[
\hat{H} = \hat{H}_0 + \lambda \hat{V}
\]  

(1.5)

\[
|\Psi_i\rangle = |\psi_i^{(0)}\rangle + \lambda |\psi_i^{(1)}\rangle + \lambda^2 |\psi_i^{(2)}\rangle + \lambda^3 |\psi_i^{(3)}\rangle + \cdots
\]  

(1.6)

\[
\mathcal{E}_i = E_i^{(0)} + \lambda E_i^{(1)} + \lambda^2 E_i^{(2)} + \lambda^3 E_i^{(3)} + \cdots
\]  

(1.7)

\[
E_i^{(1)} = \langle \psi_i^{(0)} | \hat{V} | \psi_i^{(0)} \rangle
\]  

(1.8)

\[
E_i^{(2)} = \sum_{j \neq i} \frac{|\langle \psi_i^{(0)} | \hat{V} | \psi_j^{(0)} \rangle|^2}{E_i^{(0)} - E_j^{(0)}}
\]  

(1.9)

Here the \( \lambda \) takes the value of 1. The exact electronic energy of state \( i \), \( \mathcal{E}_i \), can be expanded by Taylor series as shown in eq. (1.7). The MBPT method generates energy corrections to the zero-th order Hartree-Fock energy, \( E_i^{(0)} \). The first-order and second-order corrections shown in eq. (1.8) and eq. (1.9), will result in the HF energy and the MP2 energy. These energy corrections are non-iterative computations that account for some electronic correlations. The \( \psi_i^{(0)} \) is the \( i \)-th determinant wavefunction. The eq. (1.9) is summing over all states other than \( i \). For the usual ground state cases we are only interested in \( \mathcal{E}_0 \).

The PT method generally performs well when the ground state of the chemical system is dominated by a single electronic configuration, which means no degenerate (same energy) orbitals are present to allow electrons to stay partially in both orbitals. Degenerate states cause the denominator of eq. (1.9) to be zero and diverge the corrections.

The MP2 method delivers the lowest level of correlation correction based on HF results. Higher-order corrections are available such as the MP3, MP4 and so on. They are mostly obsolete due to the less optimal performance-to-cost ratio compared to coupled-cluster theory. The MP2 method is still widely used for a
relatively quick, though formally scaling as $O(N^5)$ but in practice very fast to compute for the sake of non-iterative procedures, correlation correction when DFT methods are not working well or the system size is too large for an attempt with computationally intense methods like coupled-cluster theory.

**Configuration interaction method.** The configuration interaction method (CI) takes care of electronic correlations by doing computations using explicit excited state configurations. The electronic configuration describes in which orbitals the electrons are staying. The CI wavefunction, eq. (1.10), is a superposition of chosen electronic configurations [17], however with different weights assigned to each configuration.

$$|\Psi_0\rangle = c_0 |\psi_0\rangle + \sum_{ia} c_i^a |\psi_i^a\rangle + \sum_{i<j} \sum_{a<b} c_{ij}^{ab} |\psi_{ij}^{ab}\rangle + \sum_{i<j<k} \sum_{a<b<c} c_{ijk}^{abc} |\psi_{ijk}^{abc}\rangle + \cdots$$  \hspace{1cm} (1.10)

The $|\psi_0\rangle$ is the configuration that all electrons stay in occupied orbitals. The $|\psi_i^a\rangle$ denotes any single excited state configuration and $|\psi_{ij}^{ab}\rangle$ for double excited state configuration. These weights, $c$, are solved by CI procedures. The CI method can lead to the exact solution of a chemical system within the chosen basis if one includes all possible electronic configurations. This is called the full CI (FCI) method.

$$H_{\mu\nu} = \langle \psi_\mu | \hat{H} | \psi_\nu \rangle$$  \hspace{1cm} (1.11)

$$E = U^+ H U$$  \hspace{1cm} (1.12)

To solve the CI wavefunction, one can simply build the CI Hamiltonian matrix using the reference configuration, single excited states and double excited states, etc. By looping over all pairs of configurations and acquiring the matrix element values between any state $|\psi_\mu\rangle$ and $|\psi_\nu\rangle$, eq. (1.11), one can construct the CI Hamiltonian matrix. By diagonalizing the Hamiltonian matrix, eq. (1.12), a set of CI eigenstates are obtained in matrix $U$. So are the eigenvalues which form the diagonal of matrix $E$. The lowest eigenvalue is the energy of the CI ground state. The
corresponding vector in the unitary matrix $U$ contains coefficients to construct the
ground state as a linear combination of CI configurations.

Let us do a simple estimation for a beryllium atom and beryllium dimer if we choose a FCI method with the 6-31G basis set. For the Be atom, there are 4 electrons in 18 spin orbitals. This gives rise to 3060 possible electronic configurations and for the Be dimer, it would have 30260340 configurations, that is, 30 million possible electronic configurations. The 6-31G basis set is considered as a fairly small basis set and for an example of moderately sized basis set, the aug-cc-pVTZ basis, which contains 110 spin orbitals for Be atom. The total number of electronic configurations for Be atom in this case would be 5773185 or 58 million states for one atom. By summing up all 58 million configurations one can resolve the FCI ground state. This is indeed an computational disaster. It is obvious that quickly the complexity of the FCI method will explode beyond the capability of any super-computers in both hardware resources and computation time. The FCI method is still useful today with the sole purposes of benchmarking other lower-cost methods.

One must truncate FCI method to a certain level as an approximation to the exact solutions. The well-known scheme is the CI with single and double excitations (CISD) model. The CISD wavefunction only contains the first three terms in eq. (1.10) and it is only a barely acceptable approximation to the real chemical systems. Many processes occur at higher than double excitation levels, e.g., the breaking of a multiple bond. The CISD method will not include any electronic correlation above the double excitation level. By adding triple (CISDT) and even quadruple excitations (CISDTQ) in the CI wavefunction, the formal scaling goes to $O(N^7)$ and $O(N^8)$, respectively. They are only feasible for really small molecules due to the high demand of computer resources and computation time. The CI methods are indeed very computationally intense.
The major reason why CI methods are not favored is that the truncated CI methods are not size-consistent. A size-consistent method is defined as the following. In a calculation of two molecules that are placed infinitely far apart, the total energy of the dimer calculation should be equal to the sum of the energies from monomer calculations, in which the monomer energy is computed without the knowledge of any other molecule. This concept makes physical sense since particles will not interact at an infinite distance. The dimer total energy should be simply additive. Artificial correlation is effectively introduced if a method is not size-consistent. For these reasons, the truncated CI methods are mostly obsolete.

**Coupled-cluster theory.** The coupled-cluster theory is major working tool in this work and it will be discussed in great detail in later sections. The real power of CC theory originates from the special form of CC wavefunctions. Similar to CI wavefunctions, the CC wavefunctions contain connected (correlated) excitations up to truncation level. For example, the coupled-cluster theory with single and double excitations (CCSD) has up to correlated double excitation terms in the wavefunctions. However, CC wavefunctions also include disconnected (coincidental) excitations for all possible electronic excitations. This is a very brilliant way of adding more electronic correlation information into the CC wavefunction but it still manages to maintain a low-order scaling. The CC theory is the state-of-the-art method to recover electronic correlation and remarkably it is applicable to moderately sized systems. The CCSD(T) method, which is the CC theory with single excitations, double excitations and perturbative triple excitations, is the so-called "gold standard" of quantum chemistry, having the ability to achieve less than 1 kcal/mol accuracy against experimental data at a formal scaling of $O(N^7)$. The systematic improbability can be obtained through allowing higher-order excitations (e.g., CCSDT adding triples or CCSDTQ adding triples and quadruples) in calculations. For this
work, substantial modifications are made to the electronic CC theory to allow CC algorithms to act on the collective coordinates of fragment fluctuations.

**Common fragment-based methods for large chemical systems.** Starting a few decades ago, and continuing apace today, enormous progress is being made in performing useful chemical simulations by decomposing large quantum mechanical calculations into fragment-based subsystems. The state of the art generally consists of some kind of embedding of a fragment into the electrostatic environment of its neighbors with various approaches to the exchange interaction,[18, 19, 20, 21, 22, 23, 24, 25, 26] or a fragment-based decomposition of a reference wavefunction as a starting point for handling electron correlation,[27, 28, 29] and all perhaps in combination with schemes for configurational sampling or techniques for handling redundancy in periodic systems.[30, 31, 32] The literature chronicling the evolution of fragment-based methods is vast, and has been reviewed several times.[33, 34, 35, 36] Here we will be introducing several important methods out of the many.

The molecular tailoring approach (MTA) [37, 38, 39] is one of the earliest fragment-based method. In MTA, the fragmentation scheme is based on the desired maximum fragment size (number of atoms per fragment) and is achieved by creating spheres on each atom and by doing recursive merging of atoms until the maximum size criterion is satisfied. These small fragments, modeled with QM methods independently, have overlapping atoms, and the super-system energy is obtained through the inclusion-exclusion principle. [36] That is, for a general property $P$, it can be described by eq. (1.13) in terms of many-body corrections. The $P_{Fi}$ is the molecular property of the $i$-th fragment, the $P_{F_i \cap F_j}$ stands for the overlap of the molecular property between $i$-th and $j$-th fragments. If we replace the $P$ with energy then we can get the super-system energy through eq. (1.13). The MTA was
mainly used for non-bonded systems and SCF methods are commonly applied to fragments.

\[ P = \sum P_{F_i} - \sum P_{F_i \cap F_j} + \ldots + (-1)^{k-1} \sum P_{F_i \cap F_j \ldots F_k} + \ldots \]  

(1.13)

Concerning with how to cap the unpaired electrons, which are resulted from breaking up bonded systems, many methods have been developed. The molecular fractionation with conjugate caps (MFCC) [40, 41] was designed for fragmentations on peptides with functional groups as caps. The introduced energy of these extra caps will be canceled out by subtracting the energy of molecules made of every two adjacent caps, the "concaps", from the super-system energy. The hydrogen caps are used in the many-overlapping body expansion (MOB or MOBE) [42, 43] method and the kernel energy method (KEM). MOBE can have high-order body terms and allow overlapping fragments to be generated. However, the KEM [44] is limited to two-body terms for non-overlapping fragments and it was designed mainly for biological polymers. HF or DFT methods are frequently used in the three schemes discussed above but electronically correlated methods are rarely applied.

Another non-overlapping method is the fragment molecular orbital (FMO) [18, 20] approach. The FMO method allows for some explicit mean field interactions of the fragments. Each fragment is first modeled in an SCF calculation to retrieve the electron density. Then the fragments are optimized again using the Coulomb fields generated by the densities of other fragments. The fragments are iterated over until the energy convergence is met. After this, the two-body (FMO2) calculation is carried out but they are solved in a non-iterative fashion. A trimer (FMO3) calculation is optional if more computer resources are available. The FMO2 energy can be obtained through eq. (1.14) and the FMO3 formula has an analogous form.
The FMO method differs from most of fragment-based methods in the procedure of allowing fragment relaxation inside the Coulomb fields of other fragments. This gives some sort of coupling between fragments however typically at a SCF level with no or parameterized electronic correlation accounted. It is worth noting that the FMO method is the most frequently used fragment-based method today.

The electrostatically embedded many-body (EE-MB) [45, 46, 47] method delivers an attempt to improve the accuracy of many-body calculations. An electrostatic field of point charges is added as the background potential while the method computes dimer (and trimer if applicable) interactions. The super-system energy is in a similar many-body correction form.

The molecules-in-molecules (MIM) method is a multi-layer partitioning method that allows multiple levels of theory to be applied. A generalized hybrid energy expression was proposed. Cap atoms are needed to close broken covalent bonds. The MIM method uses multiple layers to handle the long-range interactions instead of introducing an electrostatic embedding. The multi-layer design of MIM is very similar to the scheme used by ONIOM (our own n-layered integrated molecular orbital and molecular mechanics) method. They both use the cut-off radius \( r \) to deliver fragmentation. The MIM method has also been applied to large biological systems [48] such as proteins and DNA molecules.

The symmetry-adapted perturbation theory (SAPT) [27, 28] is a perturbative model for computing intermolecular interactions. The SAPT method has recently gained popularity among computational chemists. For typical fragment-based methods, the inter-molecular energy is acquired by subtracting dimer energy from the sum of monomer energies. But in SAPT we rewrote the dimer

\[
E^{\text{FMO2}} = \sum_{i}^{N} E_{i} + \sum_{i>j}^{N} (E_{i,j} - E_{i} - E_{j})
\]
Hamiltonian into,

\[ \hat{H} = \hat{F}^A + \hat{F}^B + \xi \hat{W}^A + \eta \hat{W}^B + \zeta \hat{V} \]  

(1.15)

where \( \hat{F}^A \) and \( \hat{F}^B \) are Fock operators for fragment A and B, respectively. \( \hat{W}^A \) and \( \hat{W}^B \) are the Møller-Plesset fluctuation operators for fragment A and B. The \( \hat{V} \) is the intermolecular Coulomb operator which takes a perturbative form. At SAPT0 level, the coefficients of \( \hat{W}^A \) and \( \hat{W}^B \) are set to zero, that is, \( \xi = \eta = 0 \). The Coulomb interaction, the first-order exchange correction, the second-order dispersion and induction corrections are chosen in the \( \hat{V} \) terms instead of using the full perturbative form which would be the exact perturbation theory for dimer. Consequently, the path for SAPT to systematic improvability is unclear. This is because SAPT deals with explicit electronic coordinates and approximations must be made to reduce computational cost. Ideally one would want to have more efficient algorithms, e.g., CC theory, to be applied among fragments to recover more electronic correlations than the MP2 level, which is frequently used in SAPT.

The motivations for proceeding beyond low-order perturbation theory are manifold. Long-range induction and other cooperative effects can be substantial and occur at relatively high orders of perturbation theory (e.g., 4th).[49, 50] In addition one expects further errors if polarizabilities from mean-field descriptions of the fragments are used, since these are known to be quite sensitive to correlation level,[51] given that excitations energies control the "stiffness" of a charge distribution (denominators in perturbation theory). There is also some dynamical screening/cooperation between charge fluctuations, collectively known as many-body dispersion (a "body" is a fragment), which are missing at low orders of perturbation theory, and this has recently been shown to be important.[52]

**Summary of computational chemistry methods.** In general, most of the fragment-based methods only apply SCF-level methods within fragments. The inter-fragment interaction is either handled at the SCF level or computed additively
using the inclusion-exclusion principle. The latter way, in theory, can approach the exactness of the selected method, for example, the SCF method. But SCF methods do not have a clear path for systematic improvability. It is desired to have an efficient algorithm to capture electronic correlations at both fragment level and super-system level, which is also capable of constructing global wavefunctions for systematic improvability.

**Recoupled-System Intuition**

One shortcoming of presently available fragment-based electronic structure methods with regard to cost is that electron correlation (if included) is computed at the level of individual electrons. Reliance on integrals describing interactions between individual electrons will render any high-order correlation scheme inefficient because the structure of the local relaxations that accompany single-electron fluctuations are effectively recomputed for each separate interaction between them.

Given the relative complexity of intra-fragment electronic correlation, as compared to the simple picture of coupled dipolar (etc.) fluctuations for inter-fragment interactions, it is reasonable to view the electronic coordinates of component fragments in terms of collective motions. By making truncations in the context of electronically correlated basis states for the fragments, high-energy local arrangements are expunged from consideration, and an effective suppression of individual degrees of freedom results. The proposed scheme is illustrated pictorially in Fig. 1, whereby the degrees of freedom at the global scope of a computation are internally correlated excitations of entire subunits. Local relaxations that accompany the primary motion of a fluctuating electron are computed once for each fragment, and the effects are permanently folded into the the effective Hamiltonian. The strong intra-fragment correlations become constant features of the low-energy part of the fragment state spaces, reducing the dimensionality of the problem.
Figure 1: Primitive and collective interactions. A pair of interacting molecules can be viewed in terms of individual electrons fluctuating between local orbitals, whereby a “primary” excitation (denoted with an asterix) is accompanied by a number of other connected nominal excitations that actually account for relaxation of the remaining charge density in the field of the shifted charge distribution. The same interaction can also be conceived of in terms of fluctuations of the fragments between their electronically correlated states, which inherently contain these relaxations. Given the large energetic scale of intra-fragment correlation, relative to the interaction between fragments, when a fragment interacts with any of its neighbors, the relaxations that accompany the local fluctuations that build the interaction should be largely similar and need only be computed once for any fragment.
The basis of internally correlated fragment states will be referred to as the excitonic basis, since this is the conceptual site basis from which Frenkel excitons\cite{53, 54, 55, 56, 57} are built. The general concept of working with fragment states is not new; it is the basis of molecular perturbation theory,\cite{58} and it features prominently in the formal development of symmetry-adapted perturbation theory\cite{27, 28} and the more recently developed molecular cluster perturbation theory.\cite{29} However, the implementations of these methods still proceed in terms of a Hamiltonian described in the one-electron basis. In low-order perturbation theory, there is indeed no computational savings in transforming the Hamiltonian because the implicit transformation may be incorporated into the non-iterative Hamiltonian action at no additional cost.

It is important to also mention the success of the general paradigm of coupling reduced spaces of correlated subsystem states as embodied in the density matrix renormalization group method and the more general class of tensor-network states.\cite{59, 60, 61} While we might anticipate lesser success when near neighbors are highly entangled with each other, a more efficient treatment of system-wide dynamical correlation should result from the exponential Ansatz, which cleanly separates connected (correlated) and disconnected (coincidental) simultaneous fluctuations.

To elucidate the potential of the proposed paradigm mechanistically, consider that dispersion forces conventionally require at least a connected double substitution, already exhausting the excitation level that is available in conventional CC with single and double substitutions (CCSD). Corrections to fragment polarizabilities due to local relaxations will show up first when higher substitutions are included, such as perturbative inclusion of connected triples [CCSD(T)], the so-called "gold-standard" of quantum chemistry. In contrast, if up to connected double substitutions of primitive electrons were to enter the fragment wavefunctions, a global excitonic CC wavefunction with up to double excitations (excitonic CCSD)
would automatically include some contributions from connected quadruple primitive substitutions. For these reasons it is not unreasonable to suggest that a practicable method might be built that can exceed the accuracy of a hypothetical conventional CCSD(T) calculation on systems that are well out of reach for even standard CCSD.

Another perspective on the justification for treating intra- and inter-fragment correlation on different footings is that the effective multipolar interactions between fragment fluctuations are orders of magnitude weaker than the bare Coulomb potential that is relevant to correlations within a fragment. The fluctuations of individual electrons within a few Ångstroms of one another are highly entangled, whereas fluctuations separated by great distances are described well as being a small perturbation away from independent.

An important aspect of the excitonic CC model going forward is that it is theoretically independent of the level of electronic structure theory used to compute the excited states of the constituent fragments. Since it is not tied to any specific level of theory, it is not inherently subject to the shortcomings of any given base method. The level of theory used for individual fragments could even be multi-reference in nature, effectively inverting the traditional paradigm of introducing dynamic correlation into multi-reference problems.\[62, 63, 64, 65, 66, 67\] Furthermore, in common with frameworks, such as ONIOM,\[68\] a site of interest may be handled in more detail than other parts of the system. For example, some small molecules undergoing reactions may be treated using multi-reference methods, whereas less expensive methods suffice for the excited states of molecules in the periphery, and all of this remains systematically improvable. Naturally, the flexibility to have then entire quantum mechanical system subject to an external potential (e.g., molecular mechanics embedding) is still available. A completely novel capability that may
open up if the method proves to be efficient enough is that the states of the fragment molecules might include vibrational excitations.

As is familiar from local-correlation methods operating on primitive electronic coordinates,[69, 70, 71, 72, 73, 29] it will also be possible to implement index restrictions that further simplify interactions on the basis of the distance between the fragments involved. Similarly, the established equation-of-motion formalism for linear response[74, 75] remains available to handle excited states in terms of fragment excitations on top of the globally correlated ground state. The ability to straightforwardly proceed from ground-state to excited-state calculations, on account of having a global wavefunction, is an important distinction in contrast to incremental methods.[30, 31, 76] which are formally exact fragment decompositions for the ground-state energy and properties (also reduced density matrices).

The formal challenge in applying the abstract machinery proposed here to realistic systems is the mapping of the *ab initio* Hamiltonian onto one written in terms of fragment fluctuations, inclusive of exact handling of exchange and even potential linear dependencies arising in the one-body basis. This is the subject of Chapter 4 that presents a general solution to the problem in detail. By way of foreshadow, the number of terms in the exactly transformed electronic Hamiltonian scales formally quartically, reflecting the scaling of the two-electron integrals tensor, but entering through a requirement to perform computations on all possible tetramers. There will admittedly be cases where this cost is prohibitive, but also a number of important cases where it is not. Inter-fragment charge-resonance terms, which could potentially even create covalent linkages, appear already in the quadratically scaling dimer part of the Hamiltonian, giving cubically scaling methods. Furthermore, to within an arbitrarily small error tolerance, the Hamiltonian asymptotically contains only a quadratic number of terms, in general. The vast majority of these
interactions also occur between systems that are not overlapping, and, for these purely Coulombic interactions, reusable single-fragment information suffices.

**Detailed Exposition of Coupled-Cluster Theory**

The excitonic coupled-cluster theory is a modification of the electronic coupled-cluster theory. In order to better explain the mechanism of X-CC methods, the electronic CC theory needs to be reviewed here in detail. As mentioned earlier, the Hartree-Fock method is a mean field theory which accounts for virtually no electronic correlation. To achieve a better energy modeling accuracy one need to choose a QM method that includes electronic correlations. The coupled-cluster theory is a very successful method for handling electronic correlations with increased efficiency. Remarkably, it is also a size-consistent approach. These are the main reasons why we chose CC theory in constructing super-system wavefunctions. However, substantial modifications were made to the traditional electronic CC theory in order to allow the CC algorithms to be applied towards the collective coordinates of fragment fluctuations.

The CC theory for electronic systems will be introduced in this section. The knowledge of the second quantization of electronic systems is essential to understand the CC theory. The core concept of the second quantization is to construct different quantum states in terms of electronic transitions (fluctuations). A set of creation and annihilation operators allow us to formally manipulate and simplify CC equations and, furthermore, to explore more efficient ways of solving the correlated problems.

**Amplitudes and wavefunctions.** The state of one electron in one orbital can be understood as the creation of such electron into this orbital from vacuum as shown in eq. (1.16),

\[ a_i^+ |i\rangle = |i\rangle \]  \hspace{1cm} (1.16)
\[
\begin{align*}
a_j |j\rangle &= |j\rangle \quad (1.17) \\
a_i^+ a_j |j\rangle &= a_i^+ |i\rangle &= |i\rangle \quad (1.18)
\end{align*}
\]

Here the \(a_i^+\) is a creation operator acting on a vacuum state \(|\rangle\) and this process generates an electron in orbital \(i\). The opposite process, eq. (1.17), of destroying an electron in orbital \(j\) can be accomplished through an annihilation operator \(a_j\) acting on an orbital \(j\). A combination of annihilation and creation operator completes the process of an electronic transition, for example, a transition from the \(j\)-th orbital to the \(i\)-th orbital is given in eq. (1.18). An annihilation onto an unoccupied orbital and a creation onto an occupied orbital will both result in zero terms.

\[
a_i^+ |i\rangle = 0 \quad (1.19)
\]

\[
a_i |k\rangle = 0 \quad \text{(if } i \neq k) \quad (1.20)
\]

When one encounters the situation in which more than one creation or annihilation operators are present in a sequence, it is necessary to know the anti-commutation relations for swapping operators. They are,

\[
\{a_p^+, a_q\} = \delta_{p,q} \quad (1.21)
\]

\[
\{a_p, a_q^+\} = \delta_{p,q} \quad (1.22)
\]

\[
\{a_p^+, a_q^+\} = 0 \quad (1.23)
\]

\[
\{a_p, a_q\} = 0 \quad (1.24)
\]

where the \(\delta\) is the Kronecker delta.

In excitonic CC theory, these creation and annihilation operators are replaced by fragment fluctuation operators that excite or de-excite fragments into different states or transfer electrons between fragments, which is discussed in detail in section 4.6.
The electronic Hamiltonian can be rewritten in terms of creation and annihilation operators,

\[ \hat{H} = \sum_{pq} h_{pq} a_p^+ a_q + \frac{1}{4} \sum_{pqrs} V_{pqrs} a_p^+ a_q^+ a_s a_r \]  \hspace{1cm} (1.25)

where,

\[ h_{pq} = \langle p | \hat{h} | q \rangle \]  \hspace{1cm} (1.26)

\[ V_{pqrs} = \langle p | \langle q | \hat{V} | r \rangle | s \rangle \]  \hspace{1cm} (1.27)

The sequence of creation and annihilation operators for two-electron operator \( \hat{V} \) in eq. (1.25) can also show up in a different order in other works at one’s convenience.

We can also rewrite the CI ground state wavefunction in eq. (1.10) in the second quantized form:

\[ |\Psi_0\rangle = \left( c_0 + \sum_{ia} c_i^a a_a^+ a_i + \sum_{i<j \atop a<b} c_{ij}^{ab} a_a^+ a_b^+ a_i a_j + \cdots \right) |\psi_0\rangle \]  \hspace{1cm} (1.28)

The truncated CI method is not size-consistent and has information only up to truncated excitation level (e.g., CISD truncated at double excitations) included in the CI wavefunctions. The truncated CI method is, in general, not an efficient algorithm of handling electronic correlations if higher-order excitations are important like breaking a multiple bond.

The CC wavefunction tackles this problem by adding more coincidental excitation terms to the CI wavefunction. Let us start with defining single, eq. (1.29), and double excitation, eq. (1.30), operators.

\[ \hat{\tau}_i^a = a_a^+ a_i \]  \hspace{1cm} (1.29)

\[ \hat{\tau}_{ij}^{ab} = a_a^+ a_i a_b^+ a_j \]  \hspace{1cm} (1.30)

Here the \( i, j, k, \cdots \) denote occupied orbitals and the \( a, b, c, \cdots \) represent the unoccupied orbitals. Let us define \( \hat{\tau}_\mu \) as a general excitation operator for any specific
excitation ($\mu$ can be any single, double or higher excitations). Then the CC ground state wavefunction based on Hartree-Fock ground state can be written in a product form,

\[ |\Psi_{CC}\rangle = \prod_{\mu} (1 + t_{\mu} \hat{\tau}_{\mu}) |\Psi_{HF}\rangle \]  

(1.31a)

\[ = \left[ (1 + t_{i}^{a} \hat{\tau}_{i}^{a}) (1 + t_{j}^{b} \hat{\tau}_{j}^{b}) \cdots (1 + t_{i}^{a} \hat{\tau}_{i}^{a}) (1 + t_{j}^{b} \hat{\tau}_{j}^{b}) \cdots (1 + t_{ij}^{ab} \hat{\tau}_{ij}^{ab}) \cdots \right] |\Psi_{HF}\rangle \]

(1.31b)

\[ = \left( 1 + \sum_{\mu'} t_{\mu} \hat{\tau}_{\mu'} \right) |\Psi_{HF}\rangle \]

(1.31c)

The $t$ here is the CC amplitude with respect to each excitation. If we combine all of the same-level excitations in eq. (1.31b) (single $t$ terms and multiple $t$ terms) into each level $\mu'$, such as, reference, singles, doubles, triple, quadruple or higher, we can then simplify eq. (1.31a) into eq. (1.31c), which takes a summation form of all possible excitations.

The rationale behind the product form of CC wavefunction is that CC wavefunction not only generates the connected (correlated) excitations, e.g. $t_{ij}^{ab} \hat{\tau}_{ij}^{ab}$ for correlated double excitations and $t_{ijk}^{abc} \hat{\tau}_{ijk}^{abc}$ for correlated triple excitations, but also gives the disconnected (coincidental) excitations, such as $t_{ij}^{ab} \hat{\tau}_{ij}^{a} \hat{\tau}_{ij}^{b}$ for coincidental double excitations and $t_{ijk}^{abc} \hat{\tau}_{ijk}^{a} \hat{\tau}_{ijk}^{b} \hat{\tau}_{ijk}^{c}$ for coincidental triple excitations. The coincidental excitations are included all the way to all possible $n$-tuple excitations even if one has the excitation level truncated at low orders. This is because the multiplying excitation terms are accumulating excitation levels and are able to generate full $n$-tuple excitations itself. These extra coincidental excitation terms allow CC theory for a better description of electronic correlations by containing more electronic correlation information, which are missing in the low-order CI method. These extra coincidental excitation makes the CC method very robust, efficient and a size-consistent scheme.
The CC theory is also systematically improvable by adding more correlated excitations into the scheme. For example, coupled-cluster theory with single, double and triple excitations (CCSDT) or coupled-cluster theory with single, double, triple and quadruple excitations (CCSDTQ) can be applied when higher-order correlated excitations are important (when CCSD method failed). By doing so systematic improvability can be achieved.

**Exponential ansatz.** There are up to $N$-tuple excitations in the full CC wavefunction. Here $N$ is the number of electrons. Naturally one wants to organize them into different excitation ranks to allow better manipulations of the equations. Therefore, we can define the cluster operator as,

$$\hat{T} = \sum_{\mu''} t_{\mu''} \hat{t}_{\mu''}$$

(1.32)

Here the $\mu''$ represents all possible excitation levels. The operator $\hat{T}$ in eq. (1.32) can be expanded in terms of excitation ranks.

$$\hat{T} = \hat{T}_1 + \hat{T}_2 + \cdots + \hat{T}_N$$

(1.33)

which has up to $N$-level excitations, and the single and double excitation cluster operators are given as,

$$\hat{T}_1 = \sum_{ia} t_i^a \hat{t}_i^a = \sum_{ia} t_i^a a_i^+ a_i$$

(1.34)

$$\hat{T}_2 = \sum_{ab} t_{ij}^a \hat{t}_{ij}^a = \frac{1}{4} \sum_{ab} t_{ij}^a a_i^+ a_j^+ a_i^+ a_j = \frac{1}{4} \sum_{ab} t_{ij}^a a_i^+ a_j^+ a_i^+ a_j$$

(1.35)

They are responsible for excitations from reference state to single or double excited states. The prefactor of $\frac{1}{4}$ comes from removing the summation restrictions on all pairs of occupied or unoccupied orbitals. And by using Taylor series,

$$e^{\hat{T}} = 1 + \hat{T} + \frac{1}{2} \hat{T}^2 + \frac{1}{6} \hat{T}^3 + \cdots$$

(1.36)

and now we define the other form of the CC wavefunction,

$$|\Psi_{\text{CC}}\rangle = e^{\hat{T}} |\Psi_{\text{HF}}\rangle$$

(1.37)
Eq. (1.37) is called the exponential ansatz of CC theory. It assumes that the CC wavefunction, $|\Psi_{CC}\rangle$, consists of the product of an excitation manifold, $e^{\hat{T}}$, with the reference wavefunction, being the HF ground state in this case. Eq. (1.31a) and eq. (1.37) are mathematically equivalent. The exponential form will also generate all possible coincidental excitations due to the use of Taylor series. This equivalence is very important for deriving practical CC solutions. By starting from the ansatz form of the CC wavefunction, one is able to solve the projected CC equations, eq. (1.42), using the well-understood Baker-Campbell-Hausdorff expansion, eq. (1.47). The expansion gives a finite set of equations and makes the CC equations trackable as opposed to dealing with all possible coincidental excitations in the CC wavefunction.

In excitonic CC scheme, the exponential ansatz still holds but the cluster operators are changed into fluctuation operators that act on fragment states rather than primitive orbitals. The fluctuation operators would result in excitations into higher-energy collective states that is unclear which electron or electrons got excited. These fluctuations are based on fragment eigenstates.

**Projected coupled-cluster equation and energy.** The variational solution to CC Schrödinger equations is very difficult to solve. This is because CC wavefunctions contain an enormous number of possible excitation terms. Simultaneous optimization on these cluster amplitudes is virtually impossible. Therefore people turned to the projected CC equations which are non-variational and much simpler to solve. Nevertheless, the CC wavefunction resolved from computing projected CC equations are no long the exact CC wavefunction but they resemble a good approximation to the variational solutions.

By using CC exponential ansatz one can obtain,

$$\hat{H}|\Psi_{CC}\rangle = E_{CC}|\Psi_{CC}\rangle = \hat{H}e^{\hat{T}}|\Psi_{HF}\rangle = E_{CC}e^{\hat{T}}|\Psi_{HF}\rangle$$

(1.38)
And for any excited state with excitation level \( \mu \), we define

\[
\langle \Psi_\mu \rangle = \langle \Psi_{\text{HF}} | \hat{\tau}_\mu^\dagger \rangle \quad (1.39)
\]

Then if we multiply another negatively signed excitation manifold to the left side of the Schrödinger equation in eq. (1.38), we have

\[
e^{-\hat{T} \hat{H} e^{\hat{T}}} | \Psi_{\text{HF}} \rangle = E_{\text{CC}} e^{-\hat{T} \hat{e}^T} | \Psi_{\text{HF}} \rangle = E_{\text{CC}} | \Psi_{\text{HF}} \rangle \quad (1.40)
\]

This product of electronic Hamiltonian with two excitation manifolds is called the non-Hermitian similarity-transformed Hamiltonian. That is,

\[
\hat{H}^T = e^{-\hat{T} \hat{H} e^{\hat{T}}} \quad (1.41)
\]

If we project any excited state \( \Psi_\mu \) to eq. (1.40) from the left side,

\[
\langle \Psi_\mu | e^{-\hat{T} \hat{H} e^{\hat{T}}} | \Psi_{\text{HF}} \rangle = E_{\text{CC}} \langle \Psi_\mu | \Psi_{\text{HF}} \rangle = 0 \quad (1.42)
\]

Eq. (1.42) is called the projected CC equation. The value of the projected CC equations will be zero when the CC wavefunction is an eigenstate of the electronic Hamiltonian. The left-hand side of eq. (1.42) is a function of amplitudes \( t \). The procedure for solving the projected CC equations will be discussed shortly.

Now we derive CC energy formula. This time we simply multiply only one excitation manifold to right side of the Hamiltonian and project the HF ground state to the left as before, we have

\[
E = \langle \Psi_{\text{HF}} | \hat{H} \left( 1 + \hat{T}_1 + \hat{T}_2 + \frac{1}{2} \hat{T}_1^2 \right) | \Psi_{\text{HF}} \rangle \quad (1.43)
\]

Here the cluster operators of higher than double excitations do not contribute to CC energy since the electronic Hamiltonian can at most de-excite a double excited state back to ground state. Any higher excitations will cancel through the projection of HF ground state from the left.

According to Brillouin theorem, the one-particle operators only contribute to second order:

\[
\langle \Psi_{\text{HF}} | \hat{H} \hat{T}_1 | \Psi_{\text{HF}} \rangle = 0 \quad (1.44)
\]
Hence, the CC energy is given by,

\[ E = \langle \Psi_{\text{HF}} | \hat{H} \left( 1 + \hat{T}_2 + \frac{1}{2} \hat{T}_1^2 \right) | \Psi_{\text{HF}} \rangle \]  \hspace{1cm} (1.45)

One thing that still needs to be addressed is how we evaluate such similarity transformed Hamiltonian. This product form is called the Baker-Campbell-Hausdorff (BCH) expansion,

\[ e^{-\hat{A}} \hat{B} e^{\hat{A}} = \hat{B} + [\hat{B}, \hat{A}] + \frac{1}{2} [[\hat{B}, \hat{A}], \hat{A}] + \frac{1}{6} [[[\hat{B}, \hat{A}], \hat{A}], \hat{A}] + \cdots \]  \hspace{1cm} (1.46)

\( \hat{A} \) and \( \hat{B} \) are arbitrary operators in this general formula. Nevertheless, the BCH expansion for amplitude equations vanishes after the fourth-order commutation, that is,

\[ e^{-\hat{T} \hat{H} e^{\hat{T}}} = \hat{H} + [\hat{H}, \hat{T}] + \frac{1}{2} [[\hat{H}, \hat{T}], \hat{T}] + \frac{1}{6} [[[\hat{H}, \hat{T}], \hat{T}], \hat{T}] + \frac{1}{24} [[[[[\hat{H}, \hat{T}], \hat{T}], \hat{T}], \hat{T}] \]  \hspace{1cm} (1.47)

Here the square brackets denote the commutation operation.

It’s worth noting that, in excitonic CC theory, this BCH expansion still holds valid and vanishes after four-order commutation in the basis of fragment fluctuations if up to two-body Hamiltonian terms are used.

**Solving the amplitude equations.** The projected CC equations have to be solved iteratively until convergence criteria, typically energy convergence and amplitude convergence, are met. In the \( n \)-th iteration, each amplitude \( t \) in eq. (1.31c) is updated as the following,

\[ t^{(n+1)} = t^{(n)} + \Delta t^{(n)} \]  \hspace{1cm} (1.48)

If we define a function \( \Omega \) as,

\[ \Omega^{(0)}_{\mu} (t^{(n)}) = \langle \Psi_{\mu} | e^{-\hat{T}^{(n)} \hat{H} e^{\hat{T}^{(n)}}} | \Psi_{\text{HF}} \rangle \]  \hspace{1cm} (1.49)

Then the change in CC amplitudes for excitation \( \mu \) can be calculated by,

\[ \Delta t^{(n)}_{\mu} = -\varepsilon_{\mu}^{-1} \Omega^{(0)}_{\mu} (t^{(n)}) \]  \hspace{1cm} (1.50)

where the \( \varepsilon_{\mu} \) is the sum of unoccupied orbitals’ energy minus the sum of occupied orbitals’ energy for this excitation \( \mu \).
This procedure is called the Quasi-Newton method which is the standard procedure for solving projected CC equations today. Within each iteration, the CC amplitudes are being updated, so is the CC energy which is recomputed and compared to the previous iteration’s value until a convergence is met. Other criteria, e.g., amplitude convergence, may apply as well. The direct inversion of iterative subspace (DIIS) [17] method can be applied to speed up the convergence. This is done by using a weighted change of amplitude for the next iteration and details can be found in the appendix.

In excitonic CC method the $\Omega$ function and the procedures still hold but they work in the basis of fragment fluctuations. The $\varepsilon_\mu$ is the energy difference between different fragment eigenstates for excitation $\mu$. 
Chapter 2: Derivation of the Excitonic Coupled-Cluster Method

General Notation

In this section, we provide the essential equations for an abstracted variant of CC theory that operates in a basis of fragment fluctuations. This establishes a notation that will be a common language that focuses specifically on the exact transformation of the Hamiltonian for electronic systems into the necessary form in Chapter 4. In order to succinctly demonstrate that the proposed paradigm is promising, comparative timings are given in Chapter 3 for systems composed of “molecules” that are internally constructed of coupled harmonic oscillators.

To start with, sets will be abbreviated as \( \{a_i\} \), for example, to represent all \( a_i \) corresponding to those values of \( i \) that are defined for a mapping \( a \). Lower-case latin letters are used for integer indices, and upper-case latin letters are used for indices that are ordered tuples of integers, for example, \( I = (i_1, \ldots, i_{\ell_I}) \), where \( \ell_I \) is the length of tuple \( I \), and \( i_1 < i_2 < \cdots < i_{\ell_I} \). Any summation implicitly runs over all values of an index that are allowed by the mapping to which the index is attached in the summand.

In order to simplify the means by which states on different enumerated fragments are referenced, we will collect basis states of all fragments into a single set \( \{ |\psi_i\rangle \} \), with a running integer index that first enumerates the states of fragment 1, and then fragment 2, and so forth. If we wish to indicate that a state belongs to a specific fragment \( m \), then the notation \( |\psi_{i_m}\rangle \) is interpreted as placing a restriction on the value of \( i \). When used as a free index, such as in denoting a set \( \{ |\psi_{i_m}\rangle \} \), or in the summation \( \sum_{i_m} \), it runs over all values consistent with this restriction.
\{\psi_{im}\}\) is therefore the set of all basis states of fragment \(m\), which is a strict subset of \{\psi_i\}.

**Fluctuation Operators**

Consider a generic super-system composed of \(N\) fragments. Given a complete basis \{\psi_{im}\} for the many-body state space of each fragment \(m\), we start with the assumption that the super-system state space is completely spanned by a set of states \{\Psi_I\}, each having the form

\[|\Psi_I\rangle = |\psi_1 \cdots \psi_N\rangle \tag{2.1}\]

with \(I = (i_1, \cdots i_N)\). This notation is intended to imply, foremostly, that \(|\Psi_I\rangle\) is tensor-product-like in structure, such that it is meaningful to say that fragment \(m\) is in state \(\psi_{im}\). By collecting the fragment state labels into a single ket, it is implied that this is a valid state for the overall system, having proper inter-particle exchange symmetry among the primitive coordinates (e.g., electrons). The fact that the indexing of the sub-indices \(i_m\) of tuple \(I\) coincides with our notation for restricting an index to a specific fragment is a convenient coincidence that does not lead to ambiguity, since the first member of \(I\) does, in fact, refer to fragment 1, etc.

We next assert the existence of a set of fluctuation operators \(\{\hat{\tau}^m_{ij}\}\), where the lower and upper indices of each operator identify two states \(\psi_i\) and \(\psi_j\) (possibly the same), which must belong to the same fragment. The action of \(\hat{\tau}^m_{im}\) onto basis state \(|\Psi_K\rangle\) is defined as changing the state of the \(m\)-th fragment according to

\[\hat{\tau}^m_{im} |\psi_{k_1} \cdots \psi_{k_m} \cdots \psi_{k_N}\rangle = \delta_{jm} \delta_{km} |\psi_{k_1} \cdots \psi_{im} \cdots \psi_{k_N}\rangle \tag{2.2}\]

This action is constructed to be reminiscent of a number-conserving pair of field operators onto a single-determinant electronic state, such that the null state results if the upper ("destruction") index corresponds to an "empty" fragment state. These operators have the following commutation relation by construction

\[\left[\hat{\tau}^l_i, \hat{\tau}^l_k\right]_+ = \delta_{jk} \hat{\tau}^l_i - \delta_{il} \hat{\tau}^l_k \tag{2.3}\]
This is shown from the definition in eq. (2.2) by noting, first, that operators on different fragments commute, and, second, that a string of two operators on the same fragment gives null if the upper index of the left operator does not match the lower index of the right operator. (The preference for positioning indices is clarified in Chapter 4, where biorthogonal machinery becomes a necessity.)

The assumptions that a set of tensor products of such states builds a complete basis for the super-system space, and that the asserted fluctuation operators are well-defined in that space, are points that need to be proven for different classes of systems. The effort to show this and arrive at exact working expressions (also amenable to approximations) for fragment-decomposed electronic systems is found in the electronic chapter. For the model fragments explored in chapter 3, which are closed systems of distinguishable degrees of freedom, these assumptions are trivially valid.

**Coupled-Cluster Ansatz**

According to the forgoing definition of the fluctuation operators, any basis state $|\Psi_I\rangle$ may be reached from any other basis state $|\Psi_J\rangle$ via a string of $N$ (or fewer) fluctuation operators. Combined with the assumption of completeness of this basis, it is then straightforward to show that an arbitrary super-system state has a unique resolution in terms of the full $N$-th order CC Ansatz (Full CC, FCC) applied to a reference state $|\Psi_O\rangle$ conforming to $\langle\Psi_{\text{FCC}}|\Psi_O\rangle \neq 0$, as

$$|\Psi_{\text{FCC}}\rangle = e^{\hat{T}}|\Psi_O\rangle$$

(2.4)

We have hereby identified the tuple $O = (o_1, \cdots o_N)$ as special, in that $|\psi_{o_m}\rangle$ is taken to be the reference state of fragment $m$. The operator $\hat{T}$ consists only of fluctuations away from the reference, denoted $\hat{r}_{im}^{om}$ with $i_m \neq o_m$, referred to specifically as *excitations*

$$\hat{T} = \sum_m \sum_{i_m \neq o_m} t_{im}^{om} \hat{r}_{im}^{om} + \sum_{m_1 < m_2} \sum_{i_{m_1} \neq o_{m_1}} \sum_{i_{m_2} \neq o_{m_2}} t_{im_1}^{im_2} \hat{r}_{im_1}^{o_{m_1}} \hat{r}_{im_2}^{o_{m_2}} + \cdots$$

(2.5)
The notation \( m_1 < m_2 \) indicates that the summation runs over all unique pairs \( etc. \) of fragments. As with conventional CC theory, excitation operators all clearly commute with one another.

As an approximation, the expansion of \( \hat{T} \) will generally be truncated at finite fragment order, with the terms written explicitly in eq. (2.5) being those retained in the generalized excitonic CCSD variant. In this case, single substitutions are associated with monomers, and doubles are associated with dimers, \( etc. \) An interesting analogue to Hartree–Fock theory that would capture long-range induction, using polarizabilities from correlated levels of theory, would be the use of only single excitations in \( \hat{T} \) (CCS). Models beyond CCS introduce entangled fluctuations among internally correlated fragment states, accounting for dispersion forces, \( etc. \), in a manner that is both self-consistent and size-consistent. In all cases, since the states \( \{\psi_i\} \) are internally correlated, the most powerful local correlations are solved for with the introduction of the super-system basis, and the reference already includes a large fraction of correlation. The state spaces of the fragments are intended to be truncated according to schemes that consider the balance of cost against accuracy of the desired property, which is perhaps local to a region of interest.

**Hamiltonian**

With a general wavefunction Ansatz available, the central task is to iteratively determine the amplitudes \( t_{im}, t_{in_1 in_2}, etc. \), that approximate an eigenstate of Hamiltonian \( \hat{H} \). More precisely, the residual of the eigenstate condition must lie outside the space of variations. This involves the familiar step of evaluating the action of the similarity-transformed Hamiltonian \( e^{-\hat{T} \hat{H} e^{\hat{T}}} \) onto the reference \( |\Psi_0\rangle \) in the context of a well-chosen non-linear optimization algorithm. The amplitude update is related to the projection of the result of this action into the space spanned by excitations up to the specified Ansatz order. Technically, convergence is met
when this projection is suitably small, though this is usually signaled by the energy becoming approximately stationary between iterations.

This brings us now to the subject of the Hamiltonian itself. In order to avoid expensive recourse to the primitive degrees of freedom during the amplitude iterations, $\hat{H}$ must also be written as an expansion in terms of strings of the fluctuation operators

$$\hat{H} = \sum_m \sum_{i_m,j_m} H_{jm}^{im} \hat{\tau}_i^{jm} + \sum_{m_1 < m_2} \sum_{i_{m_1},j_{m_1}} \sum_{i_{m_2},j_{m_2}} H_{j_{m_1} j_{m_2}}^{i_{m_1} i_{m_2}} \hat{\tau}_{i_{m_1}}^{j_{m_1}} \hat{\tau}_{i_{m_2}}^{j_{m_2}} + \cdots \quad (2.6)$$

The elements $H_{jm}^{im}$ build the Hamiltonian matrix for fragment $m$, and the higher-order terms are responsible for couplings between fragments (up to $N$-th order, in principle, depending on the kind of system). If the Hamiltonian is written as such, then the generalized normal ordering of the nested commutators of the Baker-Campbell-Hausdorff (BCH) expansion of $e^{-\hat{T}} \hat{H} e^{\hat{T}}$ can be neatly divided into the usual four parts: (1) terms that result in the null state when acted onto $|\Psi_O\rangle$, (2) a constant, (3) terms representing excitations within the specified Ansatz, and (4) terms representing excitations outside (e.g., higher) than the Ansatz. Part (2) is the pseudo-energy at any iteration, part (3) determines the update to the amplitudes, and parts (1) and (4) need not be computed. As with conventional CC methods, deriving optimal formulas and algorithms for these evaluations demands substantial work, which will be undertaken in the future.

We have now added a third assertion, that the system Hamiltonian can be written in terms of fragment fluctuations, to our original two, which concerned basis completeness and existence of the fluctuation operators. It is likely possible to prove that such a form of $\hat{H}$ always exists for broad classes of systems, relying on only benevolent assumptions; however, while interesting, it would be useless without an explicit form and a computational recipe for the scalar coefficients therein. Therefore, we leave this as an assertion to be shown on a case-by-case for each class
of systems (e.g., all fragment-decomposed electronic systems). For the model systems in chapter 3, this will be trivial, but, for electronic systems that may overlap and transfer charge, such as in chapter 5, possibly also having linear dependencies in the one-electron basis, the exercise is more intense and is undertaken in chapter 4.

Finally, it is important that the BCH expansion naturally truncates at some low order, so that evaluation of the amplitude updates has manageable computational cost. As in conventional CC theory, if the Hamiltonian itself contains only few-body terms, the requirement for this self-truncation will be fulfilled if the commutator of any two fluctuations is an operator that has both the same excitation rank as the sum of the original two and a fragment rank of less than two.\[17\] That the fragment rank is reduced by commutation is already manifest in eq. (2.3). Excitation rank of a string of fluctuations can be defined by subtracting the number of lower indices that refer to a fragment reference state from the number of upper such indices (roughly, number of excitations minus number of de-excitations). In those commutators that are not already zero, these numbers are either individually preserved or each is decremented by one, thus preserving excitation rank. Therefore, the BCH expansion must naturally truncate for this generalized CC model.
Chapter 3: Excitonic Coupled-Cluster Theory for Quantum Harmonic Oscillator

This chapter starts by introducing the quantum harmonic oscillator (HO) model, which is extensively investigated as our proof-of-principle model. The existence of analytical solutions of linearly-coupled harmonic oscillator systems enables us to build our test models with the precise error estimates. We will be constructing oscillator fragments and the inter-fragment couplings all the way to global wavefunction step by step. The analytical solution is derived with the help of several layers of transformations of coordinates. The Configuration Interaction (CI) and the Coupled-Cluster (CC) methods are applied to these fragments to recover system-wide energy. The Full CI (FCI) method is used in this work to serve as our proof-of-principle method for its simplicity and completeness.

The One-Dimensional Quantum Harmonic Oscillator

The quantum harmonic oscillator is a quantized version of the Newtonian harmonic oscillator models, such as springs and pendulums. The harmonic oscillator Hamiltonian consists of a kinetic energy term in the form of the second derivative of coordinate $x$, plus a parabolic potential energy term. The Hamiltonian is written as,

$$\hat{H} = -\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + \frac{1}{2} k x^2$$  \hspace{1cm} (3.1)$$

where the $\hbar = h/2\pi$ ($h$ is the reduced Planck constant). $m$ is the mass of the oscillator, and $k$ is the force constant. The $x$ here is a position operator that can also be used to represent the dipole operator, assuming that the oscillating particle is charged and that there is a particle of opposite charge at the origin.
Figure 2: Wavefunctions of a quantum harmonic oscillator. The black parabolic curve depicts the potential energy surface given by force constant $k$. The other colored curves are the first five wavefunctions (shifted by energy) of one quantum harmonic oscillator.

This Hamiltonian is very simple and its Schrödinger equation can be solved analytically. The eigenvalues of quantum harmonic oscillator are given by,

$$E_n = (n + \frac{1}{2})\hbar \omega$$

(3.2)

$\omega$ is the angular frequency which is equal to $\sqrt{k/m}$. The $n$ is the quantum number that takes non-negative integer values: $n = 0, 1, 2, \cdots$

The first few wavefunctions are, (as plotted in Fig. 2)

$$\psi_0(x) = (\frac{m\omega}{\pi\hbar})^\frac{1}{2} e^{-m\omega x^2/2\hbar}$$

(3.3a)

$$\psi_1(x) = (\frac{m\omega}{\pi\hbar})^\frac{1}{2} \sqrt{\frac{2m\omega}{\hbar}} x e^{-m\omega x^2/2\hbar}$$

(3.3b)

$$\psi_2(x) = (\frac{m\omega}{\pi\hbar})^\frac{1}{2} \sqrt{\frac{1}{2}} \left[ \frac{2m\omega}{\hbar} - 1 \right] e^{-m\omega x^2/2\hbar}$$

(3.3c)

$\cdots \cdots$
General Linearly Coupled Harmonic Oscillators and Solutions

The linearly coupled oscillator problems can be solved analytically. Therefore, our test models will be built with linearly coupled quantum harmonic oscillators.

**Harmonic-oscillator couplings.** We will start building a two-oscillator problem then generalize it to a many-oscillator system. The two-oscillator Hamiltonian for oscillator \( i \) coupled to oscillator \( j \) is,

\[
\hat{H}_{ij} = \hat{H}_i + \hat{H}_j + \hat{V}_{ij}
\]

(3.4)

\( \hat{H}_i \) is the Hamiltonian of oscillator \( i \) of the form in eq. (3.4). \( \hat{H}_j \) is the Hamiltonian of oscillator \( j \). \( \hat{V}_{ij} \) represents the linear coupling operator between the oscillator \( i \) and \( j \). \( \hat{V}_{ij} \) is written as,

\[
\hat{V}_{ij} = k_{ij} x_i x_j
\]

(3.5)

where \( x \) is the position operator. Therefore we have this Hamiltonian in explicit form,

\[
\hat{H}_{ij} = -\frac{\hbar^2}{2m_i} \frac{\partial^2}{\partial x_i^2} + \frac{1}{2} k_i x_i^2 - \frac{\hbar^2}{2m_j} \frac{\partial^2}{\partial x_j^2} + \frac{1}{2} k_j x_j^2 + k_{ij} x_i x_j
\]

(3.6)

We may extend this Hamiltonian to a system which consists of \( n \) oscillators. We have this system Hamiltonian written as,

\[
\hat{H} = \sum_{i=1}^{n} \hat{H}_i + \sum_{j>i}^{n,n} \hat{V}_{ij}
\]

(3.7a)

\[
= \sum_{i=1}^{n} \left[ -\frac{\hbar^2}{2m_i} \frac{\partial^2}{\partial x_i^2} + \frac{1}{2} k_{ii} x_i^2 \right] + \sum_{j>i}^{n,n} k_{ij} x_i x_j
\]

(3.7b)

\[
= \sum_{i=1}^{n} \left[ -\frac{\hbar^2}{2m_i} \frac{\partial^2}{\partial x_i^2} + \frac{1}{2} k_{ii} x_i^2 \right] + \frac{1}{2} \sum_{i\neq j}^{n,n} k_{ij} x_i x_j
\]

(3.7c)

\[
= \sum_{i=1}^{n} -\frac{\hbar^2}{2m_i} \frac{\partial^2}{\partial x_i^2} + \frac{1}{2} \sum_{i,j}^{n,n} k_{ij} x_i x_j
\]

(3.7d)

Here \( i \) and \( j \) are summation indices for all the oscillators. By restricting \( j > i \) in eq. (3.7a), it avoids the double-counting of the number of oscillator interaction
pairs. Eq. (3.7d) combines the force constants and coupling constants into one sum-
mation over all oscillator pairs.

**Transformation to unitless coordinates.** The introduction of the atomic unit system largely simplifies the quantum chemistry calculations due to the fact that certain fundamental constants take the value of one, e.g., $\hbar$ for the unit angular momentum, Bohr for the unit length and Hartree as the unit energy. Any calculation starting with all physical quantities in atomic units would lead to results in atomic units as well, e.g., energy result in Hartree. If we isolate an energy prefactor from the Hamiltonian, this would leave the rest of equation as unitless pure scalar math. The point of doing this transformation is to be able to rotate coordinates and preserve LaPlacian. Different harmonic oscillators can take different masses and force constants and this would cause the energy prefactor to be different among oscillators. Thus a unitless transformation is necessary and will wrap all the differences into the newly transformed force constants or coupling constants, which will be discussed in detail.

We will start with a system of $n$ coupled harmonic oscillators,

$$
\hat{H} = \sum_{i=1}^{n} -\frac{\hbar^2}{2m_i} \frac{\partial^2}{\partial x_i^2} + \frac{1}{2} \sum_{i,j}^{n,n} k_{ij} x_i x_j
$$

(3.8)

Now let,

$$
x_i = \sqrt{\frac{m_0}{m_i}} L_0 \tilde{x}_i
$$

(3.9)

Here the $m_0$ is the unit mass. $L_0$ is the unit length. $x_i$ is the x coordinate for oscillator $i$, and the $\tilde{x}_i$ denotes the unitless coordinate $x$ for oscillator $i$. We have,

$$
\hat{H} = E_0 \left\{ \sum_{i=1}^{n} -\frac{1}{2} \frac{\partial^2}{\partial \tilde{x}_i^2} + \sum_{i,j}^{n,n} k_{ij} \tilde{x}_i \tilde{x}_j \right\}
$$

(3.10)

where $E_0$ is the unit energy,

$$
E_0 = \frac{\hbar^2}{m_0 L_0^2}
$$

(3.11)
and
\[
\begin{cases}
\tilde{k}_{ii} = \left(\frac{m_i L_0^2}{\hbar^2 m_i}\right)^2 k_{ii}, & \text{if } i = j \\
\tilde{k}_{ij} = \left(\frac{m_i L_0^2}{\hbar^2 \sqrt{m_i m_j}}\right)^2 k_{ij}, & \text{if } i \neq j
\end{cases}
\] (3.12)

\(\tilde{k}_{ii}\) and \(\tilde{k}_{ij}\) are the new force constants and the new coupling constants, respectively.

**Diagonalization of \(k\) matrix and analytical solution.** The Schrödinger equation of a linearly-coupled harmonic oscillator system can be solved exactly mathematically with a rotation of coordinates. We start solving the Schrödinger equation for eq. (3.10) by forming the coupling matrix \(\tilde{k}\)
\[
\tilde{k} = \begin{pmatrix}
\tilde{k}_{11} & \cdots & \tilde{k}_{1n} \\
\vdots & \ddots & \vdots \\
\tilde{k}_{n1} & \cdots & \tilde{k}_{nn}
\end{pmatrix}
\] (3.13)

If we could find a matrix \(U\) that satisfies,
\[
\begin{pmatrix}
U_{11} & \cdots & U_{1n} \\
\vdots & \ddots & \vdots \\
U_{n1} & \cdots & U_{nn}
\end{pmatrix}
\begin{pmatrix}
\tilde{k}_{11} & \cdots & \tilde{k}_{1n} \\
\vdots & \ddots & \vdots \\
\tilde{k}_{n1} & \cdots & \tilde{k}_{nn}
\end{pmatrix}
\begin{pmatrix}
U_{11} & \cdots & U_{1n} \\
\vdots & \ddots & \vdots \\
U_{n1} & \cdots & U_{nn}
\end{pmatrix} =
\begin{pmatrix}
\tilde{k}_{11} & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & \tilde{k}_{nn}
\end{pmatrix}
\] (3.14)

that is,
\[
U^+ \tilde{k} U = \tilde{k}
\] (3.15)

where \(\tilde{k}\) is a diagonal matrix with all off-diagonal elements being zero. This transformation is known as the diagonalization of a matrix. Matrix \(U^+\) is the Hermitian conjugate of matrix \(U\). This turns the potential energy part into new coordinates of \(\tilde{x}\):
\[
\hat{H} = E_0 \left\{ \sum_{i=1}^{n} -\frac{1}{2} \frac{\partial^2}{\partial x_i^2} + \sum_{i=1}^{n} \frac{1}{2} \tilde{k}_{ii} \tilde{x}_i^2 \right\}
\] (3.16)
According to the invariance of the LaPlacian, any unitary rotation of coordinates 
\( \tilde{x} \rightarrow \tilde{x} \), won’t change the form of the equation:
\[
\sum_i \frac{\partial^2}{\partial \tilde{x}_i^2} = \sum_i \frac{\partial^2}{\partial \tilde{x}_i^2}
\]  
(3.17)

Therefore we have,
\[
\hat{H} = E_0 \sum_{i=1}^{n} \left( -\frac{1}{2} \frac{\partial^2}{\partial \tilde{x}_i^2} + \frac{1}{2} \tilde{k}_{ii} \tilde{x}_i^2 \right)
\]  
(3.18)

This equation is familiar. It is simply a sum of \( n \) uncoupled effective harmonic oscillators. This rotation of coordinates couples all primitive oscillators (as in \( \tilde{k}_{ij} \)) but wrap all the information into the new effective force constants \( \tilde{k}_{ii} \) for collective modes. Thus the analytical energy,
\[
E_v = E_0 \sum_{i}^{n} (v_i + \frac{1}{2}) \sqrt{\tilde{k}_{ii}}
\]  
(3.19)

where \( v = (v_1 \cdots v_n) \) collects all the quantum numbers. The \( \tilde{k}_{ii} \) are the diagonal elements of
\[
\tilde{k} = U^+ k^2 U
\]  
(3.20)

So far, we have derived the analytical solution to any linearly coupled harmonic oscillator systems.

**Fragments and Super-Systems**

Here we construct the linearly coupled oscillator Fragment \( M \) which consists of an arbitrary number, \( n_M \), of same or different oscillators. Its Hamiltonian is,
\[
\hat{H}_M = \sum_{i}^{n_M} \hat{H}_i + \frac{1}{2} \sum_{i,j}^{n_M,n_M} \hat{V}_{ij} = \sum_{i=1}^{n_M} -\frac{\hbar^2}{2m_i} \frac{\partial^2}{\partial \tilde{x}_i^2} + \frac{1}{2} \sum_{i,j}^{n_M,n_M} k_{MM}^{ij} \tilde{x}_i \tilde{x}_j
\]  
(3.21)

Here \( k_{MM}^{ij} \) denotes this coupling constant within the Fragment \( M \). The total Hamiltonian for an N-fragment harmonic oscillator system can be written as,
\[
\hat{H} = \sum_{M=1}^{N} \hat{H}_M + \sum_{M>M'}^{N,N} \hat{V}_{MM'}
\]  
(3.22)
Here $\hat{V}_{M,M'}$ denotes the linear coupling operator between Fragment $M$ and Fragment $M'$ and it is written as,

$$\hat{V}_{M,M'} = k^{MM'} x_M x_{M'}$$  (3.23)

If $x_M$ is proportional to fragment dipole and may be expanded as,

$$x_M = \sum_{i=1}^{n_M} x_{i(M)}$$  (3.24)

then,

$$\sum_{M' > M}^{N,N} \hat{V}_{M,M'} = \frac{1}{2} \sum_{M \neq M'}^{N,N} \sum_{i=1}^{n_M} \sum_{j=1}^{n_{M'}} k^{MM'} x_{i(M)} x_{j(M')} = \frac{1}{2} \sum_{M \neq M'}^{N,N} \sum_{i=1}^{n_M} \sum_{j=1}^{n_{M'}} k^{MM'}_{ij} x_{i(M)} x_{j(M')}$$  (3.25)

We make the trivial substitution $k^{MM'}_{ij} = k^{MM'}$ in order to combine them with the intra-fragment coupling constants $k^{MM'}_{ij}$. Therefore the super-system Hamiltonian can be written as,

$$\hat{H} = \sum_{M=1}^{N} \left[ \sum_{i=1}^{n_M} \frac{\hbar^2}{2m_{i(M)}} \frac{\partial^2}{\partial x_i^2(M)} + \frac{1}{2} \sum_{i,j}^{n_M,n_M} k^{MM'}_{ij} x_{i(M)} x_{j(M')} \right]$$

$$+ \frac{1}{2} \sum_{M \neq M'}^{N,N} \sum_{i=1}^{n_M} \sum_{j=1}^{n_{M'}} k^{MM'}_{ij} x_{i(M)} x_{j(M')}$$  (3.26a)

$$= \sum_{M=1}^{N} \sum_{i=1}^{n_M} -\frac{\hbar^2}{2m_{i(M)}} \frac{\partial^2}{\partial x_i^2(M)} + \frac{1}{2} \sum_{M \neq M'}^{N,N} \sum_{i=1}^{n_M} \sum_{j=1}^{n_{M'}} k^{MM'}_{ij} x_{i(M)} x_{j(M')}$$  (3.26b)

From eq. (3.26b) we know that any linearly-coupled multiple-fragment oscillator systems is the same as the individual oscillators in a globally interactive problem. Therefore the analytical solution is also available for the oscillator super-system as well as the individual fragments.

**Numerical Solutions to the Recoupled Problems**

Equipped with the analytical solutions for linearly-coupled harmonic oscillator systems, we will be able to obtain method errors against the exact values. Error comparisons can be made between our recoupled scheme and the primitive way by solving identical oscillator problems. The Configuration Interaction (CI) and
the Coupled-Cluster (CC) methods are chosen in these comparisons. The values of Hamiltonian matrix elements are essential in both CI and CC calculations, therefore we will spend the early sections on deriving Hamiltonian matrix elements.

**Hamiltonian matrix elements in the primitive oscillator state basis.**

A $n_M$-oscillator-each-fragment of $N$-fragment problem is indeed a $\sum_{M=1}^{N} n_M$ oscillators in a globally interactive problem. Thus we have our super-system wavefunction in the $Q$-th state as,

$$|\Phi_Q\rangle = |\chi_{q_11}^{11} \chi_{q_{12}}^{12} \cdots \chi_{q_{21}}^{21} \cdots \chi_{q_{N,n}}^{N,nN}\rangle \quad (3.27)$$

where $\chi$ is a primitive oscillator state and the first super-script of $\chi$ defines which fragment this oscillator state is from. The second super-script of $\chi$ denotes which oscillator it belongs to in that fragment. The sub-script $q$ represents the quantum number of such oscillator state.

Therefore, the Hamiltonian matrix element between the $P$-th and $Q$-th super-system state can be obtained from applying the eq. (3.26a) onto them,

\[ H_{PQ} = \langle \Phi_P | \sum_{M=1}^{N} \left[ \sum_{i=1}^{n_M} - \frac{\hbar^2}{2m_i(M)} \frac{\partial^2}{\partial x_i^2(M)} + \frac{1}{2} \sum_{i,j}^{n_M,n_M} k_{ij}^{MM} x_i(M)x_j(M) \right] |\Phi_Q\rangle \quad (3.28a) \]

\[ + \langle \Phi_P | \frac{1}{2} \sum_{M \neq M'}^{N,N} \sum_{i=1}^{n_M} \sum_{j=1}^{n_{M'}} k_{ij}^{MM'} x_i(M)x_j(M') |\Phi_Q\rangle \quad (3.28b) \]

\[ = \langle \chi_{p11}^{11} \cdots \chi_{pN,n}^{N,nN} | \sum_{M=1}^{N} \left[ \sum_{i=1}^{n_M} - \frac{\hbar^2}{2m_i(M)} \frac{\partial^2}{\partial x_i^2(M)} + \frac{1}{2} \sum_{i,j}^{n_M,n_M} k_{ij}^{MM} x_i(M)x_j(M) \right] |\chi_{q11}^{11} \cdots \chi_{qN,n}^{N,nN}\rangle \quad (3.28c) \]

\[ + \langle \chi_{p11}^{11} \cdots \chi_{pN,n}^{N,nN} | \frac{1}{2} \sum_{M \neq M'}^{N,N} \sum_{i=1}^{n_M} \sum_{j=1}^{n_{M'}} k_{ij}^{MM'} x_i(M)x_j(M') |\chi_{q11}^{11} \cdots \chi_{qN,n}^{N,nN}\rangle \quad (3.28d) \]

The first part of eq. (3.28c),

\[ \langle \chi_{p11}^{11} \cdots \chi_{pN,n}^{N,nN} | \sum_{M=1}^{N} \sum_{i=1}^{n_M} - \frac{\hbar^2}{2m_i(M)} \frac{\partial^2}{\partial x_i^2(M)} |\chi_{q11}^{11} \cdots \chi_{qN,n}^{N,nN}\rangle \]
is equal to,
\[ \sum_{M=1}^{N} \sum_{i=1}^{n_M} \langle \chi_{PM,i}^{M,i} | i_{M,i}^{M,i} | \chi_{QM,i}^{M,i} \rangle \delta p_{11}, q_{11} \cdots \delta p_{M,i-1+Q_{M,i-1}} \cdots \delta p_{N,n_{N}} q_{N,n_{N}} \]
and the second part,
\[ \langle \chi_{P11}^{11} \cdots \chi_{P_N,n_N}^{N,n_N} \rangle \sum_{M=1}^{N} \frac{1}{2} \sum_{i,j}^{n_M,n_M} k_{ij}^{MM} \langle x_{i(M)} | x_{i(M)} \rangle \delta p_{11}, q_{11} \cdots \delta p_{M,i-1+Q_{M,i-1}} \cdots \delta p_{N,n_{N}} q_{N,n_{N}} \]
is equal to,
\[ \frac{1}{2} \sum_{M=1}^{N} \sum_{i,j}^{n_M,n_M} k_{ij}^{MM} \langle \chi_{PM,i}^{M,i} | x_{i(M)} \rangle \delta p_{11}, q_{11} \cdots \delta p_{M,i-1+Q_{M,i-1}} \cdots \delta p_{N,n_{N}} q_{N,n_{N}} \]
Eq. (3.28d) is equal to,
\[ \frac{1}{2} \sum_{M \neq M'}^{N,N} \sum_{i=1}^{n_M} \sum_{j=1}^{n_{M'}} k_{ij}^{MM'} \langle \chi_{PM,i}^{M,i} | x_{i(M)} \rangle \delta p_{11}, q_{11} \cdots \delta p_{M,i-1+Q_{M,i-1}} \cdots \delta p_{N,n_{N}} q_{N,n_{N}} \]
\[ \times \delta p_{M',j-1+Q_{M',j-1}} \delta p_{M',j+1+Q_{M',j+1}} \cdots \delta p_{N,n_{N}} q_{N,n_{N}} \]
\[ H_{PQ} \] is the Hamiltonian matrix value for the interaction between \( P \)-th and \( Q \)-th super-system state in terms of primitive oscillator basis.

**Hamiltonian matrix elements in the fragment state basis.** This is the same oscillator problem solved by, however, a two-step method. The first step is to compute the fragment eigenstates using Hamiltonian matrix element from eq. (3.28c), which is essentially the same procedure used in the primitive methods. But these values can be replaced by the analytical solutions given in eq. (3.19) for a better accuracy. This mathematical trick is only applicable to the linearly-coupled harmonic oscillators but not to any electronic problems. This replacement is achieved through a diagonalization of the fragment coupling matrix as shown in eq. (3.20). This diagonalization rotates the oscillator fragment into a set of effective oscillators with implicit couplings. These original couplings are wrapped inside the effective force constants and the fragment state energy is a summation of individual effective oscillator energies, which is just eq. (3.19).
If we define $\Phi_M$ as the $M$-th super-system wavefunction which is consist of $N$ fragment eigenstates denoted by $|\psi_{n_i}\rangle$. The sub-script of $\psi$ represents the $i$-th fragment in this $n_i$-th rotated fragment eigenstate. The total wavefunction is a tensor product of all fragments,

$$|\Psi_M\rangle = |\psi_{n_1}\rangle \otimes |\psi_{n_2}\rangle \otimes |\psi_{n_3}\rangle \otimes \cdots \otimes |\psi_{n_N}\rangle = |\psi_{n_1} \psi_{n_2} \psi_{n_3} \cdots \psi_{n_N}\rangle$$

(3.29)

Applying the super-system Hamiltonian onto this wavefunction, we obtain the super-system Hamiltonian matrix element for the interaction between super-system $M$-th state and $M$-th state,

$$H_{MM'} = \langle \Psi_M | \hat{H} | \Psi_M \rangle$$

(3.30a)

$$= \sum_{i} \langle \psi_{n'_i} | \hat{H}_i | \psi_{n_i} \rangle \langle \psi_{n'_i} | \psi_{n_i} \rangle \cdots \langle \psi_{n'_N} | \psi_{n_N} \rangle$$

(3.30b)

$$= \sum_{i} \langle \psi_{n'_i} | \hat{H}_i | \psi_{n_i} \rangle \delta_{n'_1,n_1} \delta_{n'_2,n_2} \cdots \delta_{n'_{i-1},n_{i-1}} \delta_{n'_{i+1},n_{i+1}} \cdots \delta_{n'_N,n_N}$$

(3.30c)

$$+ \sum_{i<j} \langle \psi_{n'_i} | \hat{V}_{i,j} | \psi_{n_i} \rangle \langle \psi_{n'_i} | \psi_{n_i} \rangle \cdots \langle \psi_{n'_N} | \psi_{n_N} \rangle$$

(3.30d)

$$= \sum_{i} \langle \psi_{n'_i} | \hat{H}_i | \psi_{n_i} \rangle \delta_{n'_1,n_1} \delta_{n'_2,n_2} \cdots \delta_{n'_{i-1},n_{i-1}} \delta_{n'_{i+1},n_{i+1}} \cdots \delta_{n'_N,n_N}$$

(3.30e)

$$+ \sum_{i<j} \langle \psi_{n'_i} | \hat{V}_{i,j} | \psi_{n_i} \rangle \delta_{n'_1,n_1} \delta_{n'_2,n_2} \cdots \delta_{n'_{i-1},n_{i-1}} \delta_{n'_{i+1},n_{i+1}} \cdots \delta_{n'_N,n_N}$$

(3.30f)

$$= \sum_{i} E_{n_i} \delta_{n'_1,n_1} \delta_{n'_2,n_2} \cdots \delta_{n'_i,n_i} \cdots \delta_{n'_N,n_N}$$

(3.30g)

$$+ \sum_{i<j} \langle \psi_{n'_i} | \hat{V}_{i,j} | \psi_{n_i} \rangle \delta_{n'_1,n_1} \delta_{n'_2,n_2} \cdots \delta_{n'_{i-1},n_{i-1}} \delta_{n'_{i+1},n_{i+1}} \cdots \delta_{n'_N,n_N}$$

(3.30h)
Here the $E_{ni}$ from eq. (3.30g) can be obtained from eq. (3.19) for fragment $i$. Two-body coupling term in eq. (3.30h) can be expanded as,

$$
\langle \psi_{n'i'}| \langle \psi_{n'i'} | \hat{V}_{ij} | \psi_{ni} \rangle \psi_{nj} \rangle = k^{ij} \langle \psi_{n'i'} | x_i | \psi_{ni} \rangle \langle \psi_{n'i'} | x_j | \psi_{nj} \rangle
$$

where,

$$
\langle \psi_{n'i'} | x_i | \psi_{ni} \rangle = \sum_{p=1}^{n_i} \langle \psi_{n'i'} | \tilde{x}_{np}(i) | \psi_{ni} \rangle
$$

$$
= \sum_{p=1}^{n_i} \langle \tilde{x}_{n_i}^{i} \tilde{x}_{n_2}^{i} \cdots \tilde{x}_{n_{n_i}}^{i} | \tilde{x}_{np}(i) | \tilde{x}_{n_1}^{i} \tilde{x}_{n_2}^{i} \cdots \tilde{x}_{n_{n_i}}^{i} \rangle
$$

$$
= \sum_{p=1}^{n_i} \langle \tilde{x}_{n_p}^{i} | \tilde{x}_{np}(i) | \tilde{x}_{n_p}^{i} \rangle \delta_{n'_1,m_1} \cdots \delta_{n'_{p-1},m_{p-1}} \cdots \delta_{n'_{p+1},m_{p+1}} \cdots \delta_{n'_i,n_i}
$$

Here the

$$
\langle \tilde{x}_{n_p}^{i} | \tilde{x}_{np}(i) | \tilde{x}_{n_p}^{i} \rangle = \sqrt{\frac{1}{2}} \left( \sqrt{n_p} \delta_{n'_p,n_{p-1}} + \sqrt{n_p+1} \delta_{n'_p,n_{p+1}} \right)
$$

A non-zero value from this computation tells us this transition would generate a non-zero dipole moment which can be coupled with another harmonic oscillator’s transition dipoles. So far we have derived the formulas for the super-system Hamiltonian elements.

Recoupled-system configuration-interaction results from model molecules. The Configuration Interaction is conceptually a very simple method. The basic approach is to build a CI Hamiltonian and acquire the lowest eigenvalue from the Hamiltonian matrix diagonalization as the ground-state energy.

$$
U^+ H U = E
$$

The lowest value of this diagonal matrix $E$ is the super-system ground state energy. The remaining eigenvalues are the super-system excited state energies. If we include all possible states of each fragment in the building of the CI Hamiltonian matrix, this procedure is called the Full Configuration Interaction (FCI) method. The lowest eigenvalue is the FCI ground state energy. The FCI method is the exact solution within the given basis, whose applications are severely limited to very
small systems due to its exponentially scaling in computer resources and computation time.
Figure 3: Excitonic FCI method accuracy. Smaller errors are obtained at lower computational cost in the excitonic basis. For a dimer of “molecules” consisting of two oscillators each, calculations of equal diagonalization cost (and having the same color) are compared. The notation for the primitive calculations gives the number of basis functions afforded to each of the four individual oscillators, whereas the notation for the excitonic calculation indicates the number of states used for each “molecule”. The error of the excitonic calculation is generally much lower than primitive calculations of the same cost, and it exhibits better convergence as the coupling strength becomes weaker. Curves become ill-behaved near the bottoms of the plots, which is near the 64-bit floating-point precision limit of the computer.

Here we present two comparisons in terms of accuracy relative to exact solution and computation time, between the recoupled-system FCI approach and the primitive FCI method. The results are shown in Fig. 3.

**Recoupled-system coupled-cluster results from model molecules.** The computational efficiency of the excitonic CC scheme will be demonstrated by comparing it to a CC calculation that operates on fluctuations of the primitive oscillators that build the fragments. This “control” part of the experiment is intended to mimic the conventional practice of working with creation and annihilation
Figure 4: Excitonic FCI method onto different oscillator molecules. The excitonic model handles degeneracy (multi-reference) character more robustly. For a dimer of “molecules” consisting of two oscillators each, but with different internal structures, the accuracy of an FCI calculation using 10 basis functions per primitive oscillator is compared to that of a cheaper calculation that uses 25 eigenstates per molecule. As the mixings of the internal coordinates become more substantial due to a degeneracy, the accuracy of the excitonic calculation is seemingly unaffected, in contrast to using primitive coordinates. Curves become ill-behaved near the bottoms of the plots, which is near the 64-bit floating-point precision limit of the computer.
operators for individual electrons. In these calculations, the reference state consists of each primitive oscillator in its isolated ground state. For the “test” part of the experiment, we must choose bases for the many-body state spaces of the fragments. The internally correlated basis \( \{ |\psi_{i_m}\rangle \} \) for fragment \( m \) is taken here to be the set of exact energy eigenstates of the isolated fragment. These states and the exact values of the matrix elements of \( \hat{H}^{(m)} \) and \( \hat{\mu}^{(m)} \) in this basis are available via the aforementioned diagonalization of the internal coupling matrices for the individual fragments.

For these model systems, both the primitive and excitonic variants have abstractly the same form (each primitive oscillator can be mapped to a fragment), such that exactly the same computer program is used for the control and test, only with input tensors of dramatically different dimensions. A completely in-house implementation of the CC algorithm is used; evaluation of the BCH expansion is coded in C and called as a shared library from a generic Python-based quasi-Newton/DIIS driver.\[17\] For the BCH evaluation, expressions for the necessary matrix elements of nested commutators with respect to the reference and excitation manifold were arrived at by straightforward algebraic application of eq. (2.3), the derivation and implementation of which were both done by hand. It is worth noting that the fragment-local nature of the fluctuations provides what would map onto index restrictions in conventional diagrams, massively simplifying this brute-force algebraic approach. Such restrictions also result in an algorithm that has nominal third-order scaling with the system size, for a fixed number of states per fragment. It is worthwhile to note that prototype calculations were done by using automatic code generation, detailed in Appendix B, before this simplification, described in Appendix A, was discovered. The automatic code was dropped due to the concerns of errors in dealing with high method complexity. Though a generalized diagrammatic approach would certainly be more elegant and expose optimal factorizations,
our purpose here is only to provide relative timings. Optimization was indeed sac-
crificed at some points to produce code that is most easily read and least prone to
error, within the restriction of not introducing loops of spuriously high scaling. The
timings were run on a single 2.20 GHz Intel Xeon (E5) core, and all quantities were
kept in core memory.

Our test systems consist of linear chains of 2 to 30 oscillator-model molecules,
equally spaced by either 5 or 10 a₀. Each such fragment consists of 8 internal oscil-
lators with force constants spread evenly over the range of 1 to 2 Eₗ/a₀² (increment
of 1/7 Eₗ/a₀²). Each harmonic potential contains a particle with the same mass as
an electron, and its coordinate is interpreted as the displacement of a charge of −e
relative one of +e. The coupling constant between each pair of internal oscillators
has a positive (repulsive) value whose magnitude is 1/3 of the difference between
their force constants, such that oscillators with quite different force constants are
also substantially mixed in the fragment reference states.

For any given system, the total energy and the energies of both possible ref-
erence states (primitive or excitonic) are well defined, aside from any approxima-
tions. The energy drop to the exact solution, relative to the corresponding refer-
ence, is much larger for the primitive case, where it is the entire correlation energy,
as compared to the excitonic case, where only the inter-fragment interaction en-
ergy remains to be computed. For the systems described, the asymptotic (with re-
spect to N) energy correction for the excitonic case is −5.5 × 10⁻⁴ Eₗ/fragment
or −8.5 × 10⁻⁶ Eₗ/fragment, for 5 a₀ or 10 a₀ separation, respectively, whereas
the respective corrections for the primitive case are −5.0 × 10⁻²Eₗ/fragment or
−4.9 × 10⁻²Eₗ/fragment. Naturally, the excitonic CC correction depends much
more strongly on the interaction strength and asymptotes to zero with separation.

In both cases, we allow only single and double substitutions. Owing to the lim-
ited excitation level in the Ansatz, both the primitive and excitonic variants
Figure 5: Excitonic CCSD method tested on groups of harmonic oscillators. For two different spacings (interaction strengths) between linearly arranged “molecules”, the computational time for the CC algorithm operating on the primitive subsystem coordinates is compared to that for excitonic CC, which operates in terms of fluctuations between correlated subsystem eigenstates. Not only is the error associated with the excitonic model much smaller, but the accuracy is much better. The connecting curves are best-fit monomials with non-integer exponents, and all are very well described as cubic functions.
approach a finite inherent model error, even as the size of the basis set is increased to completeness (either number of states per oscillator or number of states per fragment, respectively). All reported timings are for the smallest basis (fastest timing) that is effectively reaches this basis set limit. This can be evaluated by comparison to the exact solution. In principle, we assert convergence with respect to basis for a given system when the order of magnitude of this error is constant and the mantissa no longer changes in the second significant digit. For convenience, in practice, a threshold is established that is used for all system sizes for a given primitive or excitonic variant that overestimates the asymptotic (with respect to \( N \)) value of this error by a few percent, since the method error as a fraction of the CCSD energy correction is fairly independent of system size for larger \( N \). For the primitive variant, this threshold error fraction is roughly \( 1.7 \times 10^{-2} \) (both spacings), and it is \( 2.6 \times 10^{-3} \) (5 \( a_0 \)) or \( 4.2 \times 10^{-5} \) (10 \( a_0 \)) for the excitonic variant. Irrespective of spacing or number of fragments, application of this criterion uniformly demands 9 states per fragment for the excitonic calculations and 4 states per oscillator for the primitive calculations (therefore, 32 states per fragment). Any individual calculation is considered converged when the energy correction changes by less than 1 part in \( 10^8 \) between BCH updates of \( \hat{T} \). Note that the demanding convergence criteria used further increases absolute timings.

The computation times for each variant and inter-fragment spacing are plotted against the number of fragments in Fig. 5. The excitonic CCSD variant is seen to be a factor of 20 to 30 times faster than for the primitive variant for the larger numbers shown. The best fit monomials to the timings for the largest 15 systems are \( 0.0025 \, s \times N^{2.94} \) vs. \( 0.051 \, s \times N^{2.97} \) for 5 \( a_0 \) spacing and \( 0.0021 \, s \times N^{2.93} \) vs. \( 0.052 \, s \times N^{2.96} \) for 10 \( a_0 \) spacing, which are also consistent with our expectation of 3rd-order scaling. This reduced computational time for the excitonic CC calculations is accompanied by results that also 3 to 6 orders of magnitude more
accurate on an absolute scale, depending on the spacing. Since the tighter basis-set convergence criterion for the excitonic results were relative to an already smaller energy correction, the accuracy of the excitonic CC results is much better than that which could be demanded from the primitive CC results. The absolute error for the basis-converged excitonic calculation of the 30-fragment system was $1.4 \times 10^{-6} E_h/\text{fragment (5 } a_0\text{)}$ or $3.2 \times 10^{-10} E_h/\text{fragment (10 } a_0\text{)}$, as opposed to $8.3 \times 10^{-4} E_h/\text{fragment (5 } a_0\text{)}$ or $8.2 \times 10^{-4} E_h/\text{fragment (10 } a_0\text{)}$ for the primitive calculation.

Clearly ignored in this discussion is the price of computing the fragment-based Hamiltonian. While this could be a formidable task for a real system, it is important to recognize that it will nevertheless present a lower scaling than the global calculation. This topic is discussed in Chapter 4.

**Conclusions**

In this section a numerical demonstration of the promise of this approach has been provided. The methods proposed here are almost assured to more efficiently provide an accuracy that is comparable to or better than conventional CC calculations of the same substitution level. There are also many features of this work which hold promise as a framework to build for finely tunable approximation schemes, in order to capture precise properties of systems interacting ever more realistically with a large number of other systems.
Chapter 4: Excitonic Coupled-Cluster Theory in Electronic Systems

Background

The general derivations from Chapter 2 presented the generic equations for an excitonic variant of coupled-cluster theory (excitonic CC). This paradigm operates directly on fluctuations between the states of entire fragments, as opposed to the states of individual electrons. The potential power of this method is rooted in restricting each fragment to the space of its lowest-energy internally correlated states, thus compressing the description of the most relevant part of the Hilbert space.

In chapter 3, a numerical demonstration was performed to illustrate the promise of the method, but this was done for model systems that consisted entirely of distinguishable degrees of freedom, forgoing the difficulties of inter-fragment electron exchange and charge transfer, the details of which are handled in this section.

The generic excitonic CC formalism asserts three conditions that must be fulfilled by the target super-systems. These are:

1. It must be possible to span the complete super-system many-electron space with a set of basis states that each have tensor-product-like structure in terms of internally correlated states of the fragments.

2. It must be possible to define fluctuation operators whose action on such a super-system basis state is to produce another basis state, wherein the state of only one fragment is changed between two of its internally correlated fragment basis states.

3. It must be possible to exactly resolve the super-system Hamiltonian in terms of these fluctuation operators. The algorithmic complexity of performing this
resolution with acceptable accuracy must be simple enough that the method is practically viable.

In this section, we take on these assertions specifically for fragment-decomposed electronic systems, including a formally exact treatment of electron-exchange antisymmetry. Beyond purely proving truth (which is trivial for point 1 and easily anticipated for point 2), we perform constructive demonstrations that lead to practicable recipes for building fragment-decomposed Hamiltonians (point 3). Regarding practicality, we ultimately show that the exact Hamiltonian can be constructed with quartically scaling resources, and this can be reduced to asymptotically quadratic for an arbitrary finite error tolerance. Many systematically improvable approximation schemes are available, starting with the user choice of wavefunction-based method for each fragment and the number of states admitted for that fragment.

Though framed in the context of excitonic CC theory, for which it was conceived, the derivations here have broader applicability to the inter-fragment electron-exchange problem. We now outline the loosely deductive process that guides us to the resolution of the exchange issue applied here.

At the root of the exchange problem is that the one-electron spaces describing each fragment cannot be orthogonalized against each other without damaging the descriptions of the correlations internal to the fragments (and in ill-defined ways that would depend on the orthogonalization scheme). We take as a premise then that the one-electron spaces available to each of the fragments are not to be altered. This means that the available space for electrons on one fragment will change whenever a neighbor changes state, due to Pauli exclusion. The available space on one fragment will then be entangled with the fluctuating, internally correlated electronic structures of its neighbors. The fact that a fluctuation then effectively changes the states of all neighbor fragments initially casts doubt on whether
the Hamiltonian can be expressed in terms of a low-order expansion of suitably defined single-fragment fluctuation operators.

From a more formal perspective, although it is trivial to assert a basis of antisymmetrized tensor products of normalized many-electron fragment states, such a super-system basis is not generally orthonormal, a reflection of the overlap-driven exchange interaction. The lack of an orthonormal basis is naturally handled by the introduction of biorthogonal complements.[77] These biorthogonal complements are, in principle, described by the inverse of the overlap matrix for the original basis states. This is purely formal, however, since direct numerical inversion of the exponentially large matrix is intractable, especially since it will not be sparse nor exactly factorizable.

Practical aspects aside, let us first consider more fundamental questions of structure, under the presumption that a biorthogonal basis is somehow available. The unique association of each complement with the fragment-state index string of precisely one tensor-product state can be leveraged to construct single-fragment fluctuation operators. Though in one-to-one correspondence with tensor-product states, the complements do not themselves have tensor-product structure in terms of the original fragment states. Many states of different fragments are required for the description of each. Furthermore, for a given truncation of the bases describing the fragments (critical for efficiency), a different effective Hamiltonian may be obtained, depending on how the complements are defined. This ambiguity in the definition of the complements results because the projected inverse of a matrix is not the same as the inverse of a projected matrix. Insisting that the complements live in the post-truncation model space would be different than choosing them from the set of complements to the untruncated tensor-product set, or any larger set. This unsettling ambiguity would likely be tolerable as an approximation for sufficiently
large fragment spaces, but the choice of complement set presumably also effects the fragment order of the resulting Hamiltonian expansion.

The choice of super-system space for the definition of the biorthogonal complements is indeed crucial to arriving at a low-fragment-order expansion for the Hamiltonian. There does exist a particular choice of complements that have simple structure, amenable to algebraic manipulation. These are the full-space complements of the complete tensor-product set of fragment bases, which themselves span the complete Fock spaces of their respective fragments (for given one-electron bases). Even though a practical model Hamiltonian must be restricted to a small subspace of the complete super-system Fock space, the complements to the basis states that define the model space cannot formally be restricted to this model space without destroying their algebraic structure; they are required to span a different space of the same dimensionality.

The nature of the relative simplicity of the aforementioned complements is that each, in fact, has tensor-product-like structure. To avoid contradiction, this is only possible if these are tensor products of secondary sets of states associated with each fragment. These secondary states are defined by applying the biorthogonal complements of the wavefunction expansion coefficients used to define the original correlated fragment states to orbital configurations that are themselves built from the biorthogonal complements of the orbitals on that fragment (as defined with respect to the full one-electron basis of the super-system). In this way, each member of the secondary many-electron basis associated with a given fragment has unit overlap with exactly one of the original states of that fragment, and it is strictly orthogonal to all other fragment basis states, both on that same fragment or on any other fragment.

In addition to spanning a different space than the original fragment basis, these complement states implicitly contain charge-transfer components, due to the
biorthogonalization of the one-electron basis. This charge transfer is the origin of needing to work formally with the fragment Fock spaces to construct the super-system biorthogonal complements. This is somewhat intuitive though, even if the model space is chosen to nominally conserve fragment charges. The fundamental ambiguity in the fragment locations of electrons in overlapping orbitals is intimately related to the concept of charge transfer. In spite of requirement to work with fragment Fock spaces, neither the original nor complement bases for the super-system space mix particle number, and only overall number-conserving strings will appear in the Hamiltonian.

Having recounted the rough arguments that expose the centrality of sets of biorthogonal orbitals and configurations built from them, this section starts by introducing notations for these constructs and then proceeds to the higher-level tasks. Once the determinants are defined (which are trivially decomposed by fragment), we then apply fragment-wise transformation matrices to define tensor products of correlated states, carrying along the inverses of those transformations to define the set of biorthogonal complements to this target basis. Expressions for the fluctuation operators are then explicitly constructed for both the super-system determinant basis and for the target internally correlated basis. This then gives an avenue for concomitant transformation of the fluctuation operators and matrix elements to the target basis from the determinant representation. These transformations importantly conserve fragment rank. When written in the biorthogonal orbital representation, the \textit{ab initio} Hamiltonian is shown to have relatively simple action on the determinant states, and therefore a simple low-fragment-rank resolution in terms of the determinant-basis fluctuation operators. This structure does not change upon transformation to the target basis. Therefore, though more involved, we can show that the matrix elements for couplings between groups of fragments in the target basis may be done in complete ignorance of the other fragments (formally, up to
tetramers are needed). A final section is dedicated to discussing the implications of some proposed approximation schemes and handling the case of linear dependency arising in the union of one-electron bases for the fragments.

Conventions and Notations

The scope of this work is limited to the case of the non-relativistic electronic Hamiltonian for fixed nuclear positions. The system is divided into \( N \) fragments, which are comprised of disjoint groups of atoms, where atoms in a fragment generally have some contiguous proximity to each other, and they may constitute what is considered to be a molecule within the system. For conceptual ease, we take it that the atoms of each fragment occupy the spatial positions that they have in the bulk system, but the many-electron state space associated with each is defined as though it were in isolation at that location.

Following the notation established in the general derivation Chapter 2, \( a_i \) denotes a member of a set denoted \( \{a_i\} \), for all \( i \) allowed by the mapping \( a \). Summations over an index implicitly run over all values allowed by the mapping to which that index is attached. A subscript on an index of a many-electron state (but not of an orbital), such as \( i_m \) or \( P_m \), serves to restrict that index to those values associated with a given fragment \( m \). On matrix-valued quantities, subscripts and superscripts will be used to index covariant and contravariant dimensions, respectively. If a matrix has both a covariant and a contravariant index, then, for purposes of matrix multiplication, the rows are taken to be enumerated by the superscripted contravariant (typically bra) index.

Lower-case latin letters will be used for integer indices, and upper-case latin letters will be used for indices that are ascending-ordered tuples of integers, for example, \( I = (i_1, \cdots i_{\ell_I}) \), where \( \ell_I \) is the length of tuple \( I \). In the case of tuples that collect many-electron-state indices, the convention of using a subscript to label the
member indices and to restrict the so-labeled indices to specific fragments will always conveniently coincide.

The greek letters $\Psi$ and $\psi$ will be used to refer to general states of the super-system and fragments, respectively, and $\Phi$ and $\phi$ will refer to the respective single-determinant states from which these are built. We use $\chi$ for the one-electron orbitals.

**The One-Electron Basis**

We presume a set of linearly independent orbitals $\{|\chi_p\rangle\}$, where each orbital may be associated with a specific fragment, e.g., atomic orbitals of constituent atoms, fragment molecular orbitals, etc. The index that enumerates this set is taken to be “blocked” by fragment, such that the first block of consecutive values enumerates the orbitals on fragment 1, and so forth. However, to avoid ambiguity later with the indexing of members of tuples, we specifically disallow the convention that a subscript on an orbital index indicates the fragment to which it belongs.

A central tool in the discussion to come will be the construction of a biorthogonal basis $\{|\chi^p\rangle\}$ for the one-electron space, such that both of the following are true

$$\text{span}(\{|\chi^p\rangle\}) = \text{span}(\{|\chi_p\rangle\})$$

$$\langle \chi^p | \chi_q \rangle = \delta_{pq} \quad (4.1)$$

The existence of such a set of biorthogonal complements is guaranteed if the original set is linearly independent, and it is impossible otherwise. Should a linearly dependent set of functions be proposed to span the one-electron state space, then some (circumventable) difficulty arises. A full discussion of this takes place later. Importantly, the resolution of this issue maps the linearly dependent case onto a problem of the same structure; therefore the remainder of the discussion of the linearly independent case is general.

Once the one-electron basis is established, we take the space spanned by it as the definition of the complete one-electron space, with the consequence that this
defines what is meant by completeness in the many-electron space, for a given total number of electrons \( n \). The complete Fock space of the super-system is then the direct sum of such complete Hilbert spaces for all \( n \), from zero up to the cardinality of \( \{|\chi_p\} \). Although concrete algorithms are presently envisioned to work in number-conserving spaces, the ability of the formalism to handle open systems could be advantageous in the future.

Many-Electron Bases

**Antisymmetry.** Those states which are antisymmetric with respect to all pairwise electron permutations build a proper subspace of the space spanned by the set of all raw (asymmetric) orbital-tensor-product states for all numbers of electrons. Therefore, there exists a linear operator \( \hat{P}_A \), that acts as an orthogonal projector onto this space of antisymmetric many-electron states, this being the complete Fock space for the super-system. We can use this construction to factor out the dependency of antisymmetrization on electron number; both the linearity and the idempotency of the projector will later be convenient. We then define what is traditionally known as a determinant state (and is identical to it) as follows

\[
|\Phi_P\rangle = |\chi_{p_1} \cdots \chi_{p_{n_P}}\rangle = \sqrt{n_P!} \hat{P}_A [ |\chi_{p_1}\rangle \otimes \cdots \otimes |\chi_{p_{n_P}}\rangle ]
\]

where \( P = (p_1, \cdots p_{n_P}) \) is an ordered tuple, and where the first state in the raw tensor product is the state of electron 1, and so forth, up to \( n_P \) electrons. It is straightforward to see that this definition is coincident with the usual definition of a determinant (including normalization) if the set \( \{|\chi_p\}\} \) were to be orthonormal. Using the biorthogonal basis, it can be shown that this is also a valid definition for determinants of non-orthonormal orbitals. Such determinants are generally not normalized nor mutually orthogonal.

**Super-system determinant basis.** The set \( \{|\Phi_P\rangle\} \) spans the complete Fock space of the super-system. The index \( P \) runs over all possible tuples of all allowed
lengths (from zero up to the cardinality of the one-electron basis). If the component indices of a tuple $P$ are chosen such that they all identify orbitals on a single fragment, then the use of the lower case $|\phi_P\rangle$ will emphasize this. The set $\{|\phi_P\rangle\}$, a strict subset of $\{|\Phi_P\rangle\}$, is then the set of all possible single-fragment determinant states on all possible fragments. The set $\{|\phi_{P_m}\rangle\}$ is a complete basis for the Fock space of fragment $m$.

Let us now make explicit the relationship between the set of all super-system determinants $\{|\Phi_P\rangle\}$ and tensor products of fragment determinants $|\phi_{P_m}\rangle$. Any tuple $P$ may be subdivided into the (potentially empty) tuples $P_1$ through $P_N$, each containing only the component indices of $P$ that refer to orbitals on the fragment indicated by its subscript. (Again, it is simply convenient that the indexing of the sub-tuples of $P$ is coincident with the convention used to indicate fragment-based restrictions of tuples.) Since all tuples in this work are taken to be ordered, and since the indexing of the orbitals is blocked by fragment, $P$ is reconstructed by simple concatenation of these sub-tuples. We may then write

\[
\begin{align*}
|\Phi_P\rangle &= \sqrt{n_P!} \hat{P}_\Lambda[|\chi_{p_1}\rangle \otimes \cdots |\chi_{p_{n_P}}\rangle] \\
&= \sqrt{\frac{n_P!}{n_{P_1}! \cdots n_{P_N}!}} \hat{P}_\Lambda[|\phi_{P_1}\rangle \otimes \cdots |\phi_{P_N}\rangle] \\
&= |\phi_{P_1} \cdots \phi_{P_N}\rangle \\
\end{align*}
\]

The first two lines of eq. (4.3) are equal according to the definition given in eq. (4.2), though care is needed to handle the nested antisymmetrizations in the second line.

The third line establishes a condensed notation, with the definition of antisymmetrization of such states (i.e., normalization) being given by the second line. We hereby say that fragment $m$ is in state $|\phi_{P_m}\rangle$. It is now clear that an alternate description of $\{|\Phi_P\rangle\}$ is as the set of all antisymmetrized tensor products of determinants of the constituent fragments. It is intuitive that the tensor product of the Fock-space bases for the fragments builds a basis for the super-system Fock space.
Biorthogonal complements of the determinant basis. Since the orbitals that construct the members of \(\{|\Phi_P\}\) are not orthonormal, this set is also not orthonormal, but it is linearly independent as a consequence of each orbital contributing a unique component to the one-electron space. We can construct the set of biorthogonal complements \(\{|\Phi^P\}\) explicitly as
\[
|\Phi^P\rangle = |\chi^{p_1} \cdots \chi^{p_n}\rangle = |\phi^{P_1} \cdots \phi^{P_N}\rangle
\]  
(4.4)
where the set of all allowed indices for \(\{|\Phi_P^P\}\) is the same as for \(\{|\Phi_P\}\), but these are mapped to a different set of states in the same Fock space. Let us verify that the set \(\{|\Phi^P\}\) is indeed the desired biorthogonal basis. Assuming that \(P\) and \(Q\) are tuples of equal length \(n = n_P = n_Q\), then
\[
\langle \Phi^P | \Phi_Q \rangle = \langle \chi^{p_1} \cdots \chi^{p_n} | \chi_{q_1} \cdots \chi_{q_n} \rangle = \sqrt{n!} \left[ \langle \chi^{p_1} | \otimes \cdots \otimes \langle \chi^{p_n} | \right] | \chi_{q_1} \cdots \chi_{q_n} \rangle
\]
\[
= \left| \begin{array}{ccc}
\langle \chi^{p_1} | \chi_{q_1} & \cdots & \langle \chi^{p_1} | \chi_{q_n} \\
\vdots & \ddots & \vdots \\
\langle \chi^{p_n} | \chi_{q_1} & \cdots & \langle \chi^{p_n} | \chi_{q_n} \\
\end{array} \right|
\]
\[
= \delta_{PQ}
\]  
(4.5)
In doing so, we have made use of the idempotency of \(\hat{\mathcal{P}}_\Lambda\) in order to project the ket determinant onto a raw tensor product state. The biorthogonal relationship of the bra and ket bases for the one-electron space ensures that all of the matrix elements inside the determinant evaluate to either zero or one. If \(P = Q\), then the determinant of the identity matrix will result. Since we have insisted that the tuples are ordered, if \(P \neq Q\), then they are different in composition, and at least one row and column of the matrix will be zero. Finally, if \(P\) and \(Q\) do not have the same length, then the states are orthogonal on account of having different particle number. We have therefore completed the proof that \(\{|\Phi^P\}\), as defined by eq. (4.4), is the set of biorthogonal complements to \(\{|\Phi_P\}\).
**General tensor-product basis and complements.** We may now proceed to define a more general basis for the super-system Fock space. This begins by defining a more general set of fragment states \( \{ |\psi_i \rangle \} \), which collects all such basis states for all fragments. These are defined by introduction of and invertible matrix \( z \) with elements \( z_i^P \).

\[
|\psi_i \rangle = \sum_P z_i^P |\phi_P \rangle
\]  

(4.6)

Since a given \( |\psi_i \rangle \) can be general mixture of determinants for a given fragment, it is indexed by a simple integer, resulting in a square matrix with rows indexed by a tuple and columns indexed by an integer. The matrix \( z \) is presumed to be block diagonal, not mixing determinants on separate fragments, nor determinants of different particle number on the same fragment. One consequence of this is that each value of the index \( i \) may be associated with a specific fragment, and, as before, we may use the notation \( i_m \) to refer to one of the states on fragment \( m \). This allows us to also write

\[
|\psi_{i_m} \rangle = \sum_{P_m} z_{i_m}^{P_m} |\phi_{P_m} \rangle
\]  

(4.7)

As needed, we let the diagonal blocks of \( z \) that transform those states on fragment \( m \) be denoted as \( z^{(m)} \), composed of elements \( z_{i_m}^{P_m} \).

In the most straightforward conceptualization, the orbitals on each fragment could be taken to be orthonormal among each other, giving an orthonormal set of determinants on each fragment, and \( z \) could be chosen to be a unitary matrix, leading to general states on a given fragment that are orthonormal among each other. In particular, \( |\psi_{i_m} \rangle \) might be an energy eigenstate of fragment \( m \), or a close relative thereof. However, overlaps between orbitals on different fragments require us to invoke the biorthogonal machinery. Below, we will also suggest an approach in which \( z \) is not unitary. In passing, we note that the condition of particle-number definiteness of the states \( |\psi_i \rangle \) could be relaxed with minor modifications below, but
it is not immediately clear what the utility of this might be, so we opt to assume it for simplicity.

Let us now construct the set of biorthogonal complements \( \{ | \psi^i \rangle \} \) to the members of \( \{ | \psi_i \rangle \} \). For ease of notation let us introduce \( \tilde{z} = z^{-1} \), with elements \( \tilde{z}_P^i \).

Using the biorthogonality of the fragment-determinant bases, it is straightforward to show that \( \langle \psi^i | \psi_j \rangle = \delta_{ij} \) for the set \( \{ | \psi^i \rangle \} \) satisfying

\[
\langle \psi^i | = \sum_P \tilde{z}_P^i \langle \phi^P | 
\]  
(4.8)

Since \( z \) is block-diagonal by fragment, \( \tilde{z} \) is also necessarily so blocked, and the notation \( | \psi^i_m \rangle \) to refer to a state associated with fragment \( m \) is therefore sensical, regardless of ambiguous physical interpretation as such. It also follows from the assumptions above that the biorthogonal complements have definite particle number.

This basis of general fragment states may be used to now further construct another basis for the super-system Fock space, \( \{ | \Psi_I \rangle \} \), where \( I = (i_1, \cdots i_N) \) gives the indices of the states of each of the sub-systems

\[
| \Psi_I \rangle = | \psi_{i_1} \cdots \psi_{i_N} \rangle
\]

\[
= \sqrt{n_I! \prod_{i_{1}} n_{i_{N}} !} \hat{P}_A \prod_{i_{1}} n_{i_{1}} ! \cdots \prod_{i_{N}} n_{i_{N}} ! | \psi_{i_1} \rangle \otimes \cdots \otimes | \psi_{i_N} \rangle
\]

\[
= \sum_{P_1} \cdots \sum_{P_N} \prod_{i_{1}} z_{i_{1}}^{P_1} \cdots \prod_{i_{N}} z_{i_{N}}^{P_N} | \phi_{P_1} \cdots \phi_{P_N} \rangle
\]

\[
= \sum_{P} Z_I^P | \Phi_P \rangle
\]
(4.9)

where the definition of antisymmetrization applied relies on the states \( | \psi_i \rangle \) having definite particle number. Again, \( P = (P_1, \cdots P_N) \), and we have used the elements a matrix \( Z \), defined as

\[
Z_I^P = \prod_m z_{i_m}^{P_m}
\]
(4.10)

with \( m \) running over all fragments. Since \( Z \) is invertible, the set \( \{ | \Psi_I \rangle \} \) is complete. For convenience, we let the inverse of \( Z \) be denoted \( \bar{Z} \) with elements \( \Bar{Z}_P^i = \prod_m z_{i_m}^{P_m} \). We use this to construct the set of biorthogonal complements \( \{ | \Psi^I \rangle \} \)
according to

$$\langle \Psi^I | = \langle \psi^{i_1} \cdots \psi^{i_N} | = \sum_P Z_P^I \langle \Phi^P |$$  \hspace{1cm} (4.11)

such that $\langle \Psi^I | \Psi_J \rangle = \delta_{IJ}$.

**Single-Fragment Fluctuation Operators**

In the chapter two that outlined the generic excitonic CC formalism, the existence of a set of sub-system fluctuation operators $\{ \hat{\tau}_j^I \}$ was asserted, having the following action on a super-system basis state

$$\hat{\tau}_j^m | \psi_1 \cdots \psi_m \cdots \psi_N \rangle = \delta_{jm} | \psi_1 \cdots \psi_{im} \cdots \psi_N \rangle$$  \hspace{1cm} (4.12)

where, as shown, the upper and lower indices must refer to two (potentially identical) states of the same fragment. This abstraction is sufficient to allow for the construction of a CC algorithm, as was done for model systems of distinguishable coordinates in that work. Here, we construct explicit expressions for operators that effect transitions of single fragments from one electronically correlated state to another, regardless of the states of the other sub-systems. We furthermore show that these well-defined operators may be used to construct a complete basis for the space of all operators acting in the super-system Fock space.

**Definitions.** For fragment-decomposed electronic systems, the following may be regarded as a definition of a sub-system fluctuation operator on fragment $m$

$$\hat{\tau}_m^{jm} = \sum_{k_1} \cdots \sum_{k_{m-1}} \sum_{k_{m+1}} \cdots \sum_{k_N} | \psi_1 \cdots \psi_{k_{m-1}} \psi_{im} \psi_{k_{m+1}} \cdots \psi_N \rangle$$

$$\times \langle \psi_1 \cdots \psi_{k_{m-1}} | \psi_{im} \psi_{k_{m+1}} \cdots \psi_N \rangle$$  \hspace{1cm} (4.13)

On account of the biorthogonality of the bases $\{|\Psi_I\rangle\}$ and $\{|\Psi^I\rangle\}$, a basis state acted upon by this operator will have non-zero projection onto, at most, one bra in the summation, and that will only happen if fragment $m$ is in state $|\psi_{jm}\rangle$, which gives unit coefficient to the super-system basis state that simply has $|\psi_{jm}\rangle$ replaced by $|\psi_{im}\rangle$. 
Clearly, the choice of basis states for the single-fragment Fock spaces does not change the discussion. We may therefore introduce an analogous set of operators defined with respect to the determinant bases, denoted for convenience as the set \{\hat{\sigma}_Q^P\} in the following equation,

\[
\hat{\sigma}_Q^{P_m} = \sum_{R_1} \cdots \sum_{R_{m-1}} \sum_{R_{m+1}} \cdots \sum_{R_N} |\phi_{R_1} \cdots \phi_{R_{m-1}} \phi_{R_m} \phi_{R_{m+1}} \cdots \phi_{R_N}\rangle \\
\times \langle \phi^{R_1} \cdots \phi^{R_{m-1}} \phi^{Q_m} \phi^{R_{m+1}} \cdots \phi^{R_N}|
\]

(4.14)

**Completeness of fluctuation operators.** Let us now demonstrate that the action of an arbitrary operator \(\hat{O}\) in the super-system Fock space may be fully represented in terms of the set of all fluctuation operators \{\hat{\tau}_i^j\}. The action of any operator in a space is fully determined by the collection of its matrix elements in a linearly independent set that spans that space, or, equivalently, taken with respect to biorthogonal bases for that space. Therefore, if we can obtain an expression for a general operator in terms of the \{\hat{\tau}_i^j\}, wherein each such matrix element is an independent degree of freedom, we will have the desired proof. This is the same as asserting that we can use this set to construct an operator that has only a single, arbitrarily chosen non-zero matrix element, such that a weighted linear combination of these is sufficient to construct an arbitrary \(\hat{O}\). Choosing the \(IJ\)-th element to be non-zero, we define the operator \(\hat{\sigma}_I^{J_L}\) such that

\[
\langle \Psi^K | \hat{\sigma}_I^{J_L} | \Psi_L \rangle = \delta_{IK} \delta_{JL} = \prod_m \delta_{i_m,k_m} \delta_{j_m,l_m}
\]

(4.15)

This is easily accomplished by setting

\[
\hat{\sigma}_I^{J_L} = |\Psi_I\rangle \langle \Psi^J| = \prod_m \hat{\tau}_{i_m}^{j_m}
\]

(4.16)

which is readily verified by showing that it matrix elements are as demanded eq. (4.15). Again, since the particular choice of fragment basis plays no role in this discussion, this conclusion applies equally well to the set \{\hat{\sigma}_Q^P\} using the determinant bases and slightly adjusted notation (tuple indices, instead of integers).
Since both $\{\hat{\tau}^j_i\}$ and $\{\hat{\sigma}^Q_P\}$ can be used to build complete bases for the space of super-system Fock-space operators, then, at an abstract level, we are assured that the Hamiltonian is expressible in terms of linear combinations of product strings of either set. Furthermore, any member of one set must be resolvable in terms of the other set. Concretely, the transformation is seen to be rather simple. Insertion of the resolution the members of $\{|\psi_i\rangle\}$ and $\{|\psi^i\rangle\}$ in terms of the members of $\{|\phi_P\rangle\}$ and $\{|\phi^P\rangle\}$, or vice versa, into the definition of either $\hat{\tau}^j_{im}$ or $\hat{\sigma}^Q_{Pm}$ in eqs. (4.13) and (4.14) results in $N - 1$ contractions of the diagonal blocks of $z$ with $\bar{z}$. Resolving the consequent $N - 1$ Kronecker deltas results the appearance of the definitions of the operators from the other set to obtain

$$\hat{\tau}^j_{im} = \sum_{Pm} \sum_{Qm} z^P_m \bar{z}^Q_m \hat{\sigma}^Q_{Pm}$$
$$\hat{\sigma}^Q_{Pm} = \sum_{i_m} \sum_{j_m} \hat{\tau}^i_{jm} \bar{z}^j_m \hat{\tau}^j_{im}$$

(4.17)

Inserting one of these identities into the other results in a truism.

We now have a rather simple transformation between fluctuation operators in the determinant basis and in the basis of correlated fragment states. Therefore any operator which is conveniently written in one basis may be straightforwardly transformed to the other. Furthermore, the transformation preserves fragment rank, i.e., a single-fragment fluctuation in one basis is represented as a superposition of single-fragment fluctuations in any basis.

Eq. (4.16) essentially provides a recipe for constructing an arbitrary operator, one matrix element at a time. Although any given operator may be represented as such, it does not have a unique resolution as a linear combination of all products of fluctuation operators. The set of all products of $N$ fluctuations (one for each fragment) is a complete and linearly independent basis for the space of all super-system Fock-space operators. Adding to this set all possible fluctuation products of lengths less than $N$ then builds a linearly dependent set of operators. A product of less
than $N$ fluctuations has multiple non-zero matrix elements, and this is an important point as pertains to choosing compact representations of operators in terms of fluctuations. Consider that we intuitively expect the Hamiltonian to couple only small numbers of fragments simultaneously, but to also generate many non-zero matrix elements, analogous to its form as a linear combination of relatively short strings of field operators in the \textit{ab initio} representation.

The Hamiltonian in terms of Fragment Fluctuation Operators

Having now shown that we have a complete tensor-product-like basis and associated fragment fluctuation operators that conform to the assertions of the second chapter, the final task in order to be able to build an excitonic CC theory for fragment-decomposed electronic systems is to provide an exact resolution of the Hamiltonian $\hat{H}$ in terms of these fluctuations. We have already established a connection between the complete super-system Fock-space basis $\{|\Psi_i\rangle\}$ and a basis of fragment-partitioned determinants $\{|\Phi_P\rangle\}$. Therefore, if we can resolve $\hat{H}$ as an expansion in terms of the fluctuation operators $\{\hat{Q}_P\}$ for the determinant basis (step 1), then it can be transformed to be in terms of the fluctuation operators $\{\hat{\tau}_i\}$ for the target basis (step 2). Though the scalar coefficients of this expansion will be expressed here in the language of full-configuration-interaction matrix elements, they are amenable to established approximations in electronic structure theory.

Biorthogonal representation of the \textit{ab initio} Hamiltonian. To perform step 1, we need to be able to compute the matrix elements of $\hat{H}$ in the determinant basis. In order to evaluate $\langle \Phi_P | \hat{H} | \Phi_Q \rangle$, we will write the \textit{ab initio} expression for the Hamiltonian in a basis of field operators that reference the biorthogonal one-electron bases.

Let the members of the sets $\{\hat{c}_p\}$ and $\{\hat{c}^p\}$ be one-electron field operators with the following actions on some determinant $|\cdots\rangle$ built from arbitrary orbitals

$$\hat{c}_p |\cdots\rangle = |\chi_p \cdot \cdots\rangle \quad \hat{c}^p |\cdots\rangle = |\chi^p \cdot \cdots\rangle$$  \hspace{1cm} (4.18)
where the result of this action may produce a state of lesser (or zero) norm, due to antisymmetrization. Let \( \{ \hat{c}_p \} \) and \( \{ \hat{d}^p \} \) be the sets of Hermitian conjugates of \( \{ \hat{c}^p \} \) and \( \{ \hat{c}^p \} \), respectively, which act as creation operators from the right onto bra states. It is straightforward to show that the biorthogonal field operators satisfy modified anticommutation relationships, by using the fact that their resolutions into any hypothetical orthonormal basis are generated by mutually inverse transformations

\[
\left[ \hat{c}_p, \hat{d}^q \right]_+ = \delta_{pq} \quad \left[ \hat{c}^p, \hat{d}_q \right]_+ = \delta_{pq}
\]

(4.19)

This is useful in showing that \( \hat{d}^q \) removes orbital \( |\chi_q\rangle \) from a determinant built from the set \( \{ |\chi_p\rangle \} \)

\[
d^q|\Phi_P\rangle = \hat{d}^q |\chi_{p_1} \cdots \chi_{p_n}\rangle
\]

\[
= \hat{d}^q \hat{c}_{p_1} \cdots \hat{c}_{p_{n-1}} |\chi_{p_1} \cdots \chi_{p_{n-1}}\rangle
\]

\[
= \sum_i \delta_{q,p_i} (-1)^{i-1} \hat{c}_{p_1} \cdots \hat{d}^{p_i} \hat{c}_{p_i} \cdots \hat{c}_{p_{n-1}} |\chi_{p_1} \cdots \chi_{p_{n-1}}\rangle
\]

\[
= \sum_i \delta_{q,p_i} (-1)^{i-1} |\chi_{p_1} \cdots \chi_{p_{i-1}} \chi_{p_{i+1}} \cdots \chi_{p_{n-1}}\rangle (4.20)
\]

where \( |\rangle \) is the vacuum state.

By elimination of transformations that again reference some hypothetical orthonormal basis, it can be shown that the following expression of the \textit{ab initio} many-electron Hamiltonian is identical to its usual expansion in an orthonormal basis

\[
\hat{H} = \sum_{p,q} h_{pq} \hat{c}_p \hat{c}_q + \sum_{p,q,r,s} v_{rs}^{pq} \hat{c}_p \hat{c}_q \hat{d}_r \hat{d}_s
\]

\[
h_{pq} = \langle \chi_p | \hat{h}_1 | \chi_q \rangle \quad v_{rs}^{pq} = \frac{1}{4} \langle \chi_p \chi_q | \hat{v}_{12} | \chi_r \chi_s \rangle (4.21)
\]

where \( \hat{h}_1 \) is the combined kinetic energy and nuclear attraction operator, and \( \hat{v}_{12} \) represents the electron–electron repulsion potential. The advantage of writing it in this manner is that, when any Hamiltonian term is acted on a determinant built from the nonorthogonal set \( \{ |\chi_p\rangle \} \), it simply induces index substitutions of the
orbitals in that determinant (or gives zero), in analogy to what is familiar from working in an orthonormal basis.

The *ab initio* expansion of $\hat{H}$ may be decomposed into terms that act on different numbers of fragments as

$$\hat{H} = \hat{H}_1 + \hat{H}_2 + \hat{H}_3 + \hat{H}_4$$

(4.22)

$\hat{H}_1$ collects together all terms from both the one-electron and two-electron parts of $\hat{H}$ that have all indices referring to orbitals of *any* single fragment, and $\hat{H}_2$ similarly collects terms for all pairs of fragments (dimers). This decomposition truncates after $\hat{H}_4$, since there are a maximum of four orbital indices. All terms in $\hat{H}_3$ and $\hat{H}_4$ must induce an inter-fragment charge transfer somewhere in the system. In taking matrix elements it will be useful to remember that each of the four $\hat{H}_M$ may be further decomposed by collecting together all the terms that act on a specific group of fragments, for example, for $\hat{H}_4$,

$$\hat{H}_4 = \sum_{m_1 < m_2 < m_3 < m_4} \hat{H}^{(m_1, m_2, m_3, m_4)}_4$$

(4.23)

where the notation $m_1 < m_2 < m_3 < m_4$ means to sum over all unique tetramers.

**Fragment-fluctuation expansion of the Hamiltonian.** We may now proceed with step 1, the building of $\hat{H}$ in terms of the set $\{\hat{Q}_P\}$, by handling each of the $\hat{H}_M$ separately. In parallel to standard practice for one- and two-electron operators, for a matrix element $\langle \Phi^P | \hat{H}_M | \Phi_Q \rangle$, it will be convenient to frame the discussion in terms of the number of fragments that have changed state in the bra, relative to the ket (henceforth, the number of “substitutions”). To express this, the number of primes on a tuple index will be used to denote the number of substitutions relative to the unprimed index, and an overbar will denote a changed value of a sub-index. For example, for two substitutions, and identifying the fragments undergoing the substitution as $m'$ and $m''$, $P'' = (P_1, \ldots P_{m'} \cdots P_{m'''} \cdots P_N)$, where $P = (P_1, \ldots P_{m'} \cdots P_{m''} \cdots P_N)$. We will always assume $m' < m'' < \cdots$. 
We know a priori, that a matrix element of a given $\hat{H}_M$ will be zero if the number of substitutions is greater than $M$. Also, the fact that any term of any $\hat{H}_M$ operates on a maximum of two electrons places further restrictions on non-zero elements. For example, $\hat{H}_4$ can have no non-zero matrix element between states that differ by less than four substitutions; in order for a two-electron operator to have indices on four separate fragments, it must induce two disjoint charge-transfers and must therefore change the states of no fewer than four fragments. Similarly, $\hat{H}_3$ only has non-zero matrix elements between states that differ by only two or three substitutions (doubly substituted matrix elements represent the effect of the average field of one fragment on a disjoint charge transfer).

As an illustration of the logic by which matrix elements are derived, let us consider the simplest case of a diagonal matrix element of $\hat{H}_1$

$$\langle \Phi_P | \hat{H}_1 | \Phi_P \rangle = \sum_m \langle \phi^{P_1} \cdots \phi^{P_m} \cdots \phi^{P_N} | \hat{H}_1^{(m)} | \phi^{P_1} \cdots \phi^{P_m} \cdots \phi^{P_N} \rangle$$

$$= \sum_m \langle \phi^{P_m} | \hat{H}_1 | \phi^{P_m} \rangle$$

(4.24)

Since the states of fragments other than fragment $m$ contribute only factors of their biorthogonal overlaps to matrix elements of $\hat{H}_1^{(m)}$, the logic for obtaining this result mirrors closely the rules for determinant matrix elements of one- and two-electron operators. In order to declutter the expression, the second line is written in terms of $\hat{H}_1$, instead of $\hat{H}_1^{(m)}$, which is valid since the matrix element will be zero for any Hamiltonian term that does not act exclusively on the fragments whose states appear in the bra and ket. Using similar logic, we obtain the following expressions for the complete collection of non-zero matrix elements between states that have the same total number of electrons (zero if the number of electrons differs).
\[
\langle \Phi^P | \hat{H}_1 | \Phi_P \rangle = \sum_m \langle \phi^P_m | \hat{H}_1 | \phi_P \rangle \\
\langle \Phi^{P'} | \hat{H}_1 | \Phi_P \rangle = \langle \phi^{P'}_m | \hat{H}_1 | \phi_{P_{m'}} \rangle \\
\langle \Phi^P | \hat{H}_2 | \Phi_P \rangle = \sum_{m_1 < m_2} \langle \phi^P_{m_1} \phi^P_{m_2} | \hat{H}_2 | \phi_{P_{m_1}} \phi_{P_{m_2}} \rangle \\
\langle \Phi^{P'} | \hat{H}_2 | \Phi_P \rangle = \sum_m \langle \phi^{P'}_m \phi^P_m | \hat{H}_2 | \phi_{P_{m'}} \phi_{P_m} \rangle \\
\langle \Phi^{P''} | \hat{H}_2 | \Phi_P \rangle = \langle \phi^{P''}_{m'} \phi^{P''}_{m''} \phi^P_m | \hat{H}_2 | \phi_{P_{m'}} \phi_{P_{m''}} \phi_{P_m} \rangle \\
\langle \Phi^{P'''} | \hat{H}_3 | \Phi_P \rangle = \sum_m (-1)^{f_{m''''}} \langle \phi^{P_{m'}} \phi^{P_{m''}} \phi^{P_{m'''}} \phi^P_m | \hat{H}_3 | \phi_{P_{m'}} \phi_{P_{m''}} \phi_{P_{m'''}} \phi_{P_m} \rangle \\
\langle \Phi^{P''''} | \hat{H}_3 | \Phi_P \rangle = \langle \phi^{P_{m'}} \phi^{P_{m''}} \phi^{P_{m'''}} \phi^{P_{m'''}} \phi^P_m | \hat{H}_3 | \phi_{P_{m'}} \phi_{P_{m''}} \phi_{P_{m'''}} \phi_{P_{m'''}} \phi_{P_m} \rangle \\
\langle \Phi^{P'''''} | \hat{H}_4 | \Phi_P \rangle = \langle \phi^{P_{m'}} \phi^{P_{m''}} \phi^{P_{m'''}} \phi^{P_{m''''}} \phi^{P_{m'''''}} \phi^P_m | \hat{H}_4 | \phi_{P_{m'}} \phi_{P_{m''}} \phi_{P_{m'''}} \phi_{P_{m''''}} \phi_{P_{m'''''}} \phi_{P_m} \rangle \\
\tag{4.25}
\]

For the sake of compact expressions, the summations sometimes admit two copies of the same fragment state into a determinant; clearly, this evaluates to zero due to antisymmetry and is inconsequential. Similarly, the summations also allow the states of the fragments to appear out of order. This does not contradict our established notation, which only insists that \textit{tuple} components are ordered. Only in one case does the reordering lead to a sign change, and that is for a matrix element of \( \hat{H}_3 \) in which a charge has been transferred between fragments \( m' \) and \( m'' \), and when the summation index \( m \) is between \( m' \) and \( m'' \); the exponent \( f_{m''''} \) is one in this case and zero otherwise.

It can now be verified that the following expressions for the \( \hat{H}_M \) have the same matrix elements as above and are therefore identically the same operators
\[ \hat{H}_1 = \sum_m \sum_{P_m, Q_m} \langle \phi P_m | \hat{H}_1^{(m)} | \phi Q_m \rangle \sigma^{Q_m}_{P_m} \]
\[ \hat{H}_2 = \sum_{m_1 < m_2} \sum_{P_{m_1}, P_{m_2}, Q_{m_1}, Q_{m_2}} \langle \phi P_{m_1} | \phi P_{m_2} | \hat{H}_2 | \phi Q_{m_1} \phi Q_{m_2} \rangle \sigma^{Q_{m_1}}_{P_{m_1}} \sigma^{Q_{m_2}}_{P_{m_2}} \]
\[ \hat{H}_3 = \sum_{m_1 < m_2 < m_3} \sum_{P=(P_{m_1}, P_{m_2}, P_{m_3}), Q=(Q_{m_1}, Q_{m_2}, Q_{m_3})} \langle \Phi | \hat{H}_3 | \Phi_Q \rangle \sigma^{Q_{m_1}}_{P_{m_1}} \sigma^{Q_{m_2}}_{P_{m_2}} \sigma^{Q_{m_3}}_{P_{m_3}} \]
\[ \hat{H}_4 = \sum_{m_1 < m_2 < m_3 < m_4} \sum_{P=(P_{m_1}, P_{m_2}, P_{m_3}, P_{m_4}), Q=(Q_{m_1}, Q_{m_2}, Q_{m_3}, Q_{m_4})} \langle \Phi | \hat{H}_4 | \Phi_Q \rangle \sigma^{Q_{m_1}}_{P_{m_1}} \sigma^{Q_{m_2}}_{P_{m_2}} \sigma^{Q_{m_3}}_{P_{m_3}} \sigma^{Q_{m_4}}_{P_{m_4}} \] (4.26)

The slow shift in notation over the first three lines is only for the sake of clarity, serving both to shorten the latter expressions and to ensure they are unambiguous by expanding the former. This completes step 1. As discussed, since the basis of all possible products is overcomplete, the expression for the Hamiltonian is not unique, but we conjecture that this the most compact expansion. In practice, the numerical evaluation of the matrix elements contained in these expressions proceeds via insertion of the \textit{ab initio} expansion of the Hamiltonian, but, conveniently, for small numbers of fragments. One advantage of breaking \( \hat{H} \) into the \( \hat{H}_M \) is simplification of the matrix element expressions; as written, the two-fragment component does not require us to subtract off any double-counted single-fragment energies.

Finally, inserting the transformations of the operators \( \{ \sigma^Q_P \} \) in terms of the set \( \{ \hat{\tau}_i^j \} \), and recognizing the definitions of the \( \{ | \Psi_i \rangle \} \) and \( \{ | \Psi^I \rangle \} \) in terms of the \( \{ | \Phi_P \rangle \} \) and \( \{ | \Phi^P \rangle \} \) we complete step 2 to obtain (in the most condensed notation)
\[
\mathcal{H} = \sum_m \sum_{I=\{i_m\}} \langle \Psi^I | \hat{H}_1 | \Psi_J \rangle \hat{\tau}^{j_m}_{i_m} + \sum_{m_1 < m_2} \sum_{I=\{i_{m_1},i_{m_2}\}} \langle \Psi^I | \hat{H}_2 | \Psi_J \rangle \hat{\tau}^{j_{m_1}}_{i_{m_1}} \hat{\tau}^{j_{m_2}}_{i_{m_2}} + \sum_{m_1 < m_2 < m_3} \sum_{I=\{i_{m_1},i_{m_2},i_{m_3}\}} \langle \Psi^I | \hat{H}_3 | \Psi_J \rangle \hat{\tau}^{j_{m_1}}_{i_{m_1}} \hat{\tau}^{j_{m_2}}_{i_{m_2}} \hat{\tau}^{j_{m_3}}_{i_{m_3}} + \sum_{m_1 < m_2 < m_3 < m_4} \sum_{I=\{i_{m_1},i_{m_2},i_{m_3},i_{m_4}\}} \langle \Psi^I | \hat{H}_4 | \Psi_J \rangle \hat{\tau}^{j_{m_1}}_{i_{m_1}} \hat{\tau}^{j_{m_2}}_{i_{m_2}} \hat{\tau}^{j_{m_3}}_{i_{m_3}} \hat{\tau}^{j_{m_4}}_{i_{m_4}}
\]

(4.27)

Practical Considerations

Calculation and approximation of matrix elements. Once the Hamiltonian in terms of fragment fluctuations is obtained, the global excitonic CC calculation proceeds as outlined in the second chapter (and as demonstrated there for the oscillator-model molecules). The non-trivial aspect of calculations for real systems is the computation of the matrix elements that resolve the Hamiltonian. We will now show rigorously that these may be obtained from completely independent calculations on small groups of fragments, in spite of the insistence on a globally biorthogonalized one-electron basis. Secondarily, the general features of approximation schemes will briefly be addressed.

The matrix elements we need are of the general form

\[
\langle \psi^{i_{m_1}} \cdots \psi^{i_{m_M}} | \hat{H}_M^{(m_1,\cdots,m_M)} | \psi^{j_{m_1}} \cdots \psi^{j_{m_M}} \rangle
\]

for \( M \) up to 4. We recall that \( \hat{H}_M^{(m_1,\cdots,m_M)} \) collects together all terms in the \textit{ab initio} expression for \( \hat{\mathcal{H}} \) where at least one field operator references an orbital on each of the fragments \( m_1, \cdots m_M \), and no others. It can therefore, be thought of as a subset of the terms that build the Hamiltonian for that \( M \)-fragment system, except that, technically, in order to build any one of the destruction operators \( \hat{d}^\dagger \), we need information about all orbitals in the system, since the orbital \( | \chi^a \rangle \) that \( \hat{d}^\dagger \) references has
been constructed to be orthogonal all except \(|\chi_q\rangle\). The biorthogonal complements that one would construct by considering only the orbitals on fragments \(m_1, \cdots m_M\) are then different from the complements to these orbitals considering the full system basis. However, the difference is entirely accounted for by addition of components that are strictly orthogonal to the original basis attached to these fragments. This means that these components are of no consequence in both the action of the destruction operators onto the ket and when they are projected onto in the bra. Therefore computations for matrix elements on the group of fragments \(m_1, \cdots m_M\) may be done in complete ignorance of any information about other fragments.

The use of a biorthogonal basis within a group for which a matrix element is being computed, however, is crucial to obtaining the correct value for a globally consistent set of monomer fluctuations. This impacts computational schemes for the matrix elements. To make this concrete in the context of possible families of realistic approximations, let us consider sets of operators \(\{\hat{z}_i\}\) and \(\{\hat{\bar{z}}^i\}\)

\[
\hat{z}_i = \sum_P z_i^P \hat{c}_p \cdots \hat{c}_{p_n}, \quad (\hat{z}_i)\dagger = \sum_P \bar{z}_i^P \hat{d}_{p_n} \cdots \hat{d}_p
\]

where, as before, \(P = (p_1, \cdots p_n)\), such that the correlated basis states and complements are given by

\[
|\psi_i\rangle = \hat{z}_i |\rangle \\
|\psi^i\rangle = \hat{\bar{z}}^i |\rangle
\]

The relevant matrix elements may now be written as vacuum expectation values

\[
\langle \psi^i_{m_1} \cdots \psi^i_{m_M} | \hat{H}^{(m_1, \cdots m_M)}_{M} | \psi_{j_{m_1}} \cdots \psi_{j_{m_M}} \rangle \\
= \langle [\hat{z}_{i_{m_M}}]\dagger \cdots [\hat{z}_{i_{m_1}}]\dagger | \hat{H}^{(m_1, \cdots m_M)}_{M} | [\hat{z}_{j_{m_1}}] \cdots [\hat{z}_{j_{m_M}}] \rangle
\]

It is important to realize that, once \(\hat{H}^{(m_1, \cdots m_M)}_{M}\) is known in the biorthogonal one-electron basis, the rules for evaluating matrix elements of this form using field-
operator algebra are exactly the same as in an orthonormal basis, since the strings that compose the $(\hat{z}^i_m)^\dagger$ and $\hat{z}_j^n$ contain only operators from the sets $\{\hat{d}^p\}$ and $\{\hat{c}_p\}$, respectively, which we have noted earlier adhere to fermionic anticommutation relationships. This moves all the complexity of working in a non-orthogonal basis into the one-electron basis transformation of the *ab initio* Hamiltonian representation. If the states of a single fragment are computed using a traditional CI approach in a one-electron basis that is orthonormal on that fragment, then the diagonal blocks $z^{(m)}$ are just unitary matrices, and the $\bar{z}^{(m)}$ are their transpose conjugates (though only a few columns/rows of each may be used). More sophisticated parameterization of the matrices $z^{(m)}$ and their inverses may also be used, which need not be unitary, as would be the case if equation-of-motion CC theory[74, 75] were used to obtain the fragment excited states.

We have finally arrived at a point where it is clear that the computational inputs into the global excitonic CC calculation are completely independent calculations on small of groups of fragments, using an *ab initio* Hamiltonian with matrix elements modified to account for nonorthogonalities of the fragment orbital spaces. It is important to realize that the formally quartic scaling of the Hamiltonian, entering here through a requirement to perform calculations on all possible tetramers, is simply a reflection of the quartic scaling of the two-electron integrals tensor. The expansion for $\hat{H}$ would truncate after dimer interactions if inter-fragment charge-transfer were to be suppressed, which may be reasonable in many cases where the fragments are closed-shell molecules. Furthermore, some amount of inter-fragment charge resonance (likely the most important physics) may be represented using only dimer terms. The transformation to a non-orthogonal representation also does not change basic scaling relationships. The charge-transfer interaction decays exponentially with distance, and two fluctuations that occur outside the overlap radius of
each other interact purely electrostatically, decaying asymptotically with the inverse third power.

As is familiar from the properties of the two-electron integrals tensor in a local basis, since there are only a linear-scaling number of non-negligible independent fluctuations (local to a fragment or short-range charge transfers), there are at most a quadratically scaling number of non-negligible terms in the excitonic Hamiltonian at the mesoscopic scale (linear in the bulk limit). In turn, all but a linear-scaling number of these pairwise couplings are in the electrostatic coupling regime, outside of exchange range. Those belonging to this majority fraction may further be efficiently decomposed into single-fragment quantities that can be used to compute all long-range interactions. Namely, we require the one-electron transition density operators[78] (and perhaps only multipoles thereof) between each pair of states on each fragment, which importantly do not contain detailed information about internal correlations.

Handling linear dependencies. Since the framework proposed in this chapter depends so critically on both the ability to assign one-electron functions to specific fragments and on the linear independence of that basis, it behooves us to address the topic of what to do when a local function set \( \{ |\chi_p\rangle \} \) is linearly dependent, causing the sets \( \{ |\Phi_p\rangle \} \) and \( \{ |\Psi_I\rangle \} \) to also be linearly dependent. The linear dependent case is important, as it is certain to arise with large single-electron basis sets using diffuse functions. The general problem is that the ambiguity in fragment location is complete for an electron in an orbital belonging to one fragment, if that orbital can be constructed as a linear combination of orbitals solely on other fragments. We must somehow “remove” this linear dependency while while simultaneously preserving the local structure of the framework. The direct removal of null vectors (or “nearly” null vectors, in numerical algorithms) at the one-electron level produces a basis whose members are not generally conducive to being assigned to
specific atoms or fragments. Yet, removal of individual fragment-local or atom-local
functions to resolve a linear dependency can create artificial asymmetries.

We will now present an alternate framing of the above derivation at a level of
abstraction that allows us to handle both the linearly independent and linearly de-
pendent cases together. This allows us to show the path necessary to obtain work-
ing expressions for the linearly dependent case without going into details.

For any set of many-electron states \( \{ |\Psi_I\rangle \} \), linear dependence notwithstanding,
we may write a resolution of the identity in the following form

\[
1 = \sum_I |\Psi_I\rangle \langle \Psi_I|
\]

for some (potentially non-unique) choice of the set \( \{ |\Psi_I\rangle \} \), which is a point we will
discuss carefully momentarily. By inserting this resolution of the identity into the
Schrödinger equation solved by \( |\Psi_{GS}\rangle \) for the ground state, we arrive at the ma-
trix eigenvalue equation

\[
H |\Psi_{GS}\rangle = E_{GS} |\Psi_{GS}\rangle
\]

where \( |\Psi_{GS}\rangle \) is a column matrix with elements \( \Psi_{GS}^I = \langle \Psi_I | \Psi_{GS} \rangle \), and the matrix \( H \)
has elements \( H^I_J = \langle \Psi_I | \hat{H} | \Psi_J \rangle \). In the linearly dependent case, there are clearly
some redundant degrees of freedom, and multiple choices of \( |\Psi_{GS}\rangle \) can be used to
represent the physical state \( |\Psi_{GS}\rangle \); however, for a given choice of \( \{ |\Psi_I\rangle \} \), the ma-
trix \( H \) has fixed definition. Assuming non-zero \( E_{GS} \), the eigenvector of \( H \) with
eigenvalue of \( E_{GS} \) is unique. The physically redundant degrees of freedom are de-
termined by the necessity that the projection of this \( |\Psi_{GS}\rangle \) into the null space of
\( H \) is zero. (This can be generalized in the case of a degenerate ground state.) \( H \)
also has spurious eigenvectors with eigenvalue zero, corresponding to its null space,
and, if the physical state \( |\Psi_{GS}\rangle \) happened to have eigenvalue of zero, these spuri-
ous eigenvectors (with no physical norm) could mix with it. This could be easily
remedied by a scalar Hamiltonian shift, but this is also rare accident in quantum chemistry.

In the case where \( \{ |\Psi_I \rangle \} \) is linearly independent, it is straightforward to show that the members of the set \( \{ |\Psi^I \rangle \} \) are the biorthogonal components uniquely determined by

\[
|\Psi^I \rangle = \sum_J \bar{S}^{IJ} |\Psi_J \rangle
\]  

(4.33)

where the elements \( \bar{S}^{IJ} \) belong to the matrix \( \bar{S} = S^{-1} \), where \( S \) is the matrix of overlaps \( S_{IJ} = \langle \Psi_I | \Psi_J \rangle \). Recalling that both \( S \) and \( \bar{S} \) are self-adjoint matrices, we then straightforwardly arrive at the expected result that \( H = \bar{S} \tilde{H} \), where \( \tilde{H} \) has matrix elements \( \tilde{H}_{IJ} = \langle \Psi_I | \tilde{H} |\Psi_J \rangle \). The remainder of the work (already complete) can then be viewed as an exercise in constructing explicit forms of the members of \( \{ |\Psi^I \rangle \} \) in terms of the fragment basis and underlying orbitals, such that we may write \( \tilde{H} \) in terms of fragment fluctuations with explicit formulas for the necessary scalar coefficients (matrix elements) of strings of those fluctuations, opening the door to the more compact non-linear CC parameterization of \( |\Psi_{GS} \rangle \).

In the linearly dependent case, we will find it convenient to define fluctuation operators directly in the auxiliary space of coefficients. Let the set of column matrices \( \{ |\Psi_I \rangle \} \) represent the orthonormal basis vectors of this coefficient space, such that

\[
|\Psi_{GS} \rangle = \sum_I |\Psi_I \rangle |\Psi^I \rangle
\]

\[
= \sum_{i_1} \cdots \sum_{i_N} \Psi_{(i_1 \cdots i_N)}^{(i_1 \cdots i_N)} |\Psi_{GS} \rangle
\]  

(4.34)

where, in the second line, we remind ourselves of the structure of the index \( I \) as a tuple that is decomposable in terms of fragment-state labels. Let us now define a set of fluctuation matrices \( \{ \tau_i^j \} \) with the following action

\[
\tau_{i m}^j \Psi_{(k_1 \cdots k_m \cdots k_N)} = \delta_{jm} \Psi_{(k_1 \cdots i_m \cdots k_N)}
\]  

(4.35)
where, as with the physical states, these fluctuation matrices may be defined as superpositions of appropriate dyads. Similarly, the proof that any matrix in the coefficient space may be written as a sum of products of fluctuation matrices proceeds along the same lines, assuring us that $H$ may be built from such fluctuations. The precise fragment-wise structure of this expansion of $H$ (and therefore $\Psi_{GS}$) depends on determining the elements $\langle \Psi^I | \hat{H} | \Psi_J \rangle$. The primary difference, relative to the linearly independent case, is that the choice of $\{ |\Psi^I \rangle \}$ is not unique, and the null space of $H$ depends on this choice.

According to our definition of antisymmetrization as application of a projection operator, and in spite of any linear dependencies, eq. (4.9) still provides the definition of the antisymmetrized tensor-products of correlated fragment states in terms of orbital configurations. Therefore, in the linearly dependent case, eq. (4.11) is still a suitable choice of the members of $\{ |\Psi^I \rangle \}$ that satisfy eq. (4.31), in terms of a suitable choice of $\{ |\Phi^P \rangle \}$ that satisfy

$$1 = \sum_P |\Phi_P \rangle \langle \Phi^P|$$

(4.36)

which is quickly verified by inserting these resolutions of $|\Psi_J \rangle$ and $|\Psi^I \rangle$ into eq. (4.31), and realizing that the invertibility of $Z$ is independent of linear dependencies in the set of physical states. The important consequence of this is that, once formulas for the elements of a matrix $H'$ (with elements $H'_{PQ} = \langle \Phi^P | \hat{H} | \Phi_Q \rangle$) are known for a given choice of $\{ |\Phi^P \rangle \}$, and their structure in terms of numbers of fragments coupled is analyzed, then the path to constructing $H$ in terms of fluctuations $\{ \tau_i \}$ via application of fragment-local transformations is the same as for the linearly independent case.

As with the linearly independent case, the construction of explicit forms for the members of $\{ |\Phi^P \rangle \}$ can be straightforward, so long as the matrix $s$ built from the one-electron overlaps $s_{pq} = \langle \chi_p | \chi_q \rangle$ can be partially inverted. Let the matrix $\bar{s}$
satisfy

\[ p_1 = p_1 \bar{s} s = s \bar{s} p_1 \]

\[ 1 = p_0 + p_1 \]  \hspace{1cm} (4.37)

where \( p_0 \) is the orthogonal projector onto the null space of \( s \), and \( p_1 \) is the orthogonal projector onto the range (complement of the null space) of \( s \). Notably, only the projection of \( \bar{s} \) onto the range of \( s \) is uniquely defined. Because of this, \( \bar{s} \) is not necessarily self adjoint for any choice, but we do have \( p_1 \bar{s} p_1 = (p_1 \bar{s} p_1)^\dagger \), which is equally valuable. For a given choice of \( \bar{s} \), a choice of the members of \( \{|\Phi^F\rangle\} \) may be specified again by eq. (4.4), but using

\[ \langle \chi_p | = \sum_q \bar{s}^{pq} \langle \chi_q | \]  \hspace{1cm} (4.38)

where \( \bar{s}^{pq} \) is an element of \( \bar{s} \).

We may, as before, define \( \hat{c}_p \) and \( \hat{\bar{c}}^p \) as the creation operators associated with \( |\chi_p\rangle \) and \( |\chi^p\rangle \), respectively, and define \( \hat{d}_p \) and \( \hat{\bar{d}}^p \) as their respective Hermitian conjugates. The Hamiltonian again takes exactly the same form as in eq. (4.21) in terms of these operators, shown in the same way as before. The difference is that there is now redundancy in the operator set, and the anticommutation rules do not follow. A given string in the field-operator resolution of the Hamiltonian will act not only on the fragments to which the indices in that string belong, but also on any fragments whose orbitals can be linearly combined to build the orbital in question. If a given orbital is linearly independent of the rest of the basis (likely the majority), then only expected fragment is involved. Regardless, this does not increase the fragment order of the Hamiltonian, but it increases the number of fragments that could be associated with a given orbital index, which is an intuitive consequence of having a linear dependency arise due to overlaps of diffuse orbitals on different fragments.
A final comment is worthwhile in terms of connecting back to the abstract framework of resolutions of the identity. Similar to what we have done in the one-electron space, the matrices $\bar{S}$ and $\bar{S}'$ may be defined as satisfying

\[ P_1 \bar{S} S = S \bar{S} P_1 = P_1 = 1 - P_0 \]
\[ P_1' \bar{S}' S' = S' \bar{S}' P_1' = P_1' = 1 - P_0' \]  \hspace{1cm} (4.39)

where $P_0$ and $P_0'$ are the orthogonal projectors into the null spaces of $S$ and $S'$, respectively, where $S'$ has elements $S'_{PQ} = \langle \Phi_P | \Phi_Q \rangle$. The members of the sets $\{ |\Psi^I\rangle \}$ and $\{ |\Phi^P\rangle \}$ may be expressed for a given choice of $\bar{S}$ and $\bar{S}'$ as

\[ \langle \Psi^I | = \sum_j \bar{S}^{IJ} \langle \Psi_J | \]
\[ \langle \Phi^P | = \sum_Q \bar{S}'_{PQ} \langle \Phi_Q | \]  \hspace{1cm} (4.40)

which coincides with the unique choice of biorthogonal complements in the linearly independent case, when $\bar{S} = S^{-1}$ and $\bar{S}' = S'^{-1}$. In the linearly dependent case, the use of these definitions in resolutions of the identity [eqs. (4.31) and (4.36)] correspond to additions of and projections onto null vectors. If we make the choice that the arbitrary part of $\bar{s}$ is zero, or equivalently that the arbitrary part of $\bar{S}'$ is zero, which is likely the most convenient and practical choice, and we furthermore insist on eq. (4.11) for our choice of the set $\{ |\Psi^I\rangle \}$, then this is equivalent to choosing

\[ \bar{S} = \bar{Z} \; P_1' \bar{S}' P_1' \bar{Z}^\dagger \]  \hspace{1cm} (4.41)

This is clearly a valid pseudo-inverse of $S = Z^\dagger S' Z$, but it has non-zero components in the arbitrary part, due the fact that some eigenvectors of $S'$ with non-zero eigenvalue transform to vectors that lie partly in the null space of $S$, and vice versa. The result of this is that, if the full Hamiltonian matrix were to be built, we would have

\[ H = \bar{Z} H' Z \]  \hspace{1cm} (4.42)
which is a similarity transformation of the Hamiltonian in the configuration basis, as expected, but where

\[ P_0 H \neq 0 \] (4.43)

meaning that this matrix does not have the same null space as \( S \). \( H \) still has a null space of the same dimension, but it corresponds to a specific non-intuitive (but implicit and innocuous) choice of representation of the vector in the basis of tensor products of correlated states.

**Conclusions**

In this chapter we have shown that it is theoretically grounded to construct fragment-based fluctuation operators with the necessary properties to perform excitonic coupled-cluster calculations in the context of an exact method for electronic systems (for the non-relativistic, Born-Oppenheimer Hamiltonian, projected into a basis). Explicit and tractable recipes have been given for constructing the necessary Hamiltonian in the fragment-fluctuation representation, accounting exactly for inter-fragment electron exchange, charge transfer, and even the possibility of a linearly dependent set of underlying one-electron orbitals.

There are many features of this work that hold promise for using this as a framework to build for finely tunable and systematically improvable methods to capture precise properties of systems interacting with a large number of other systems. To begin with, each fragment may be handled at varying level of quantum-mechanical detail, eventually giving way to force-fields far from the region of interest. Importantly, the Fock-space formulation of this work allows for inter-fragment charge resonance, potentially already represented well by the dimer terms of the Hamiltonian. This could possibly even create covalent linkages between fragments.

The fragment-fluctuation resolution of the Hamiltonian may be constructed with quadratically scaling resources to within an arbitrary finite error tolerance, and all but a linear-scaling number terms may be efficiently decomposed in terms
of single-fragment quantities, these being one-electron transition density matrices. Pressing further into unknown territory, it may be possible to parameterize or interpolate these matrices as functions of the internal nuclear coordinates of the fragments. Additionally, the full range of local approximations that are applied to conventional CC methods have analogues that can be applied to excitonic CC methods.

In essence, the construction of the exact Hamiltonian of a super-system in the representation of fragment fluctuations draws on the common chemical intuition that atoms, molecules, and functional groups do not completely forgo their individual properties upon interaction with others, but that these may be heavily perturbed. Force-fields are the simplest conceptual model of this intuition, and fragment-based methods constitute formalizations, amenable to approximation, of the underlying quantum mechanics. Given a proper, systematically improvable, global ground-state wavefunction for a super-system and a straightforward extension to the excited-state regime, then the range of phenomena that can be explored is effectively unlimited, from bond breaking in complex environments to time-dependent explorations of charge transfer. As respects the latter, this section of derivations could also be viewed as a framework by which to make tractable systematic improvements to recent work on \textit{ab initio} exciton theory.\cite{56, 57} Indeed, the compelling results of such work thus far are encouraging for the present proposition.
Chapter 5: Excitonic Coupled-Cluster Theory Initial Tests: Beryllium Atoms

In this chapter, the excitonic coupled-cluster theory is put into a series of tests to demonstrate the theory’s capability and efficiency in handling large molecular systems. The resolution of the electronic Hamiltonian, eq. (4.27), requires several detailed procedures to be developed. Due to the fact that up to two-fragment Hamiltonian terms have been implemented currently, the goal of this chapter is mainly to compute monomer Hamiltonian terms $\langle \Psi^I | \hat{H}_1 | \Psi_J \rangle$ for all fragments and dimer Hamiltonian terms $\langle \Psi^I | \hat{H}_2 | \Psi_J \rangle$ for all pairs of fragments. The values of Hamiltonian terms rely on the state $\Psi$ being used. It becomes important to know what type of and how these monomer states are chosen.

This chapter will start out with introducing the test systems, beryllium atoms in a chain. One-fragment and two-fragment ground state calculations are carried out for doing a decomposition of dimer states into tensor products of monomer states. This is done by a transformation of basis. Then Hamiltonian terms of $\hat{H}_1$ and $\hat{H}_2$ can be resolved in those states. A technique involves with reduced density matrix is applied to obtain better fragment states for more efficient representation of fragment fluctuations. In the end, FCI method and electronic CC methods are used in error and timing comparisons.

Beryllium Bonding Problem

The test system is chosen to be a chain of beryllium atoms. The beryllium atom is a small system, for which FCI energy results can be obtained to serve as exact values in error comparisons. Even though beryllium atom may seem quite simple, the near degenerate orbitals make the molecular modeling for Be-Be
bonding interactions surprisingly difficult. Inside each beryllium atom there are two valence electrons and four spatial orbitals, namely the 2s and the 2p orbitals. Despite the fact that beryllium atom has all valence electrons paired up, there still exists a weak interaction between Be atoms as shown in Fig. 6. The curve on the bottom is a high quality fit to the experimental data by Meshkov, *et al.* [79] Other curves are calculated theoretically using FCI method with various basis sets in the range of 3 Å to 10 Å. Small basis sets, like cc-pVDZ set, encountered failures in describing the dissociation potential. Consequently the test cases in this chapter focus on using beryllium atom systems as a model problem to demonstrate the robustness of excitonic coupled-cluster theory, rather than giving a comprehensive chemically relevant solution to the beryllium bonding interactions.
The model systems consist of a linear chain of \( n \) beryllium atoms, \( \text{Be}_n \). For each calculation beryllium atoms are evenly spaced at a chosen distances. The \( \text{Be}_n \) systems are modeled by both electronic CC calculations and X-CCSD calculations. The simplest test case is to reconstruct the Be dimer dissociation potential at a range of 3 Å to 10 Å. Trimer and higher order systems are also investigated for their dissociation potentials. The largest system calculated in this work is a \( \text{Be}_{100} \) system with a Be-Be bond distance of 4.5 Å which is the \( \text{Be}_2 \) equilibrium bond distance obtained at FCI/6-31G level.

**Fragmentation and Fragment Eigenstates**

A proper fragmentation scheme is very crucial for obtaining accurate energetic modeling results. Poorly chosen fragments can give rise to troublesome cases, e.g., breaking of multiple bonds, which increase computational complexity of the problems. In the \( \text{Be}_n \) test cases each Be atom is naturally chosen as a fragment.

At the beginning of an excitonic CCSD calculation, fragment eigenstates are resolved at the absence of other fragments. In this work, these eigenstates are obtained by FCI/6-31G calculations. The 6-31G basis set is chosen due to the low computational cost and the FCI calculations would give the exact results for each fragment. Therefore, the errors generated in the super-system calculations should originate solely from X-CCSD approximations, which comes very handy for evaluating the capabilities of X-CC scheme.

The FCI calculation is carried out by first generating all possible electronic configurations. The program then iterates over all pairs of electronic configurations and apply electronic Hamiltonian, eq. (1.25), onto each pair to compute corresponding matrix element, eq. (1.11). The FCI Hamiltonian matrix can be built from these elements. A full diagonalization of the Hamiltonian matrix, built in 6-31G basis, eq. (1.12), can be easily accomplished on any modern desktop computer. The lowest eigenvalue from the diagonalization is the FCI ground state energy for one
beryllium atom, one fragment as well in this case. The eigenvector that corresponds to the lowest eigenvalue holds all the coefficients (in the basis of all FCI configurations) for constructing the FCI ground state wavefunction. Other eigenvalues and eigenvectors describe excited state energies and excited state wavefunctions, respectively. These obtained wavefunctions are used as the fragment fluctuations in super-system calculations.

The charge transfer states are also obtained in this work. It is not common to have two positive or two negative charges on the same atom. Therefore, only Be$^+$ and Be$^-$ configurations are needed. These will be used in constructing the dimer interaction terms for Be$^+$-Be$^-$, Be$^+$-Be, Be$^-$-Be, Be$^+$-Be$^+$ and Be$^-$-Be$^-$ states. The eigenvalues and eigenstates of Be$^+$ and Be$^-$ are solved using FCI/6-31G method.

**Decomposition of Dimer States into Monomer States**

The Hamiltonian requires dimer interaction terms to be expressed in terms of monomer states. This fragment locality is the key to reduce computational cost for fragment-based methods. In excitonic coupled-cluster calculations, locality is enforced by only allowing each fragment to fluctuate within its own orbital space. These restrictions are implemented by a transformation of one-fragment basis into many-fragment basis. The procedure equivalently decomposes many-fragment (currently two-fragment) states into tensor products of monomer states, in which fragment state locality is explicitly enforced.

In the interest of working expeditiously, we chose to compute all dimer Hamiltonian elements using a third-party programming platform (PyQuante quantum chemistry suite [80]) in terms of the Be$_2$ molecular orbital basis. The required transformations for this are now described.

**Be atom eigenstates in the dimer basis.** The first step in computing dimer interactions is to resolve the states of monomers in the dimer basis. In order to accomplish such transformations of basis, one needs to have an overlap matrix built
between one-fragment Hartree-Fock orbitals and two-fragment Hartree-Fock orbitals. This starts with the beryllium atom and the beryllium dimer being modeled by HF/6-31G method, respectively, for their Hartree-Fock eigenvectors (orbitals). The unitary matrix holding all the eigenvectors for Be atom is denoted as $C_{\text{Be}}$. The unitary matrix holding all the eigenvectors for Be$_2$ is named $C_{\text{Be}_2}$. These unitary matrices are square matrices that contain all the coefficients for constructing any HF orbitals in terms of a linear combination of basis functions (6-31G basis functions in this case).

The $C_{\text{Be}}$ matrix in this case has the $\alpha$ and $\beta$ spins separated by blocks as shown in Fig. 7. The horizontal axis corresponds to eigenstates (converged HF orbitals) and the vertical axis represents the primitive 6-31G basis functions (atomic basis). The $\alpha\beta$ and $\beta\alpha$ blocks are zero since $\alpha$ and $\beta$ spin wavefunctions are orthogonal. The $\alpha\alpha$ and $\beta\beta$ blocks will have non-zero values. But what is needed in this transformation is a double-height matrix, denoted as $C\{\text{Be in Be}_2\}$, which can be built simply by stacking two $C_{\text{Be}}$ matrices vertically and it is shown in Fig. 8.

The Be$_2$ Hartree-Fock eigenvectors are held in $C_{\text{Be}_2}$ as shown in Fig. 9. The upper half of the $C_{\text{Be}_2}$ correspond to the 6-31G basis functions of Be atom $A$ and the lower half for Be atom $B$. Both atoms have their spin orbitals separated into $\alpha$ and $\beta$ blocks vertically. The horizontal axis represents the Be$_2$ Hartree-Fock molecular orbitals.

The third matrix needed here is the overlap matrix for 6-31G basis functions in the Be$_2$ calculation. It is named $S_{\text{Be}_2}$. This overlap matrix is computed in the preparation of SCF procedure for Be$_2$ calculation. The matrix holds the overlap integrals between contracted 6-31G basis functions.

The first transformation is obtained by multiplying three matrices together as shown in eq. (5.1)

$$
\tilde{S}_{\text{Be,Be}_2} = C^T\{\text{Be in Be}_2\} S_{\text{Be}_2} C_{\text{Be}_2}^T
$$

(5.1)
Figure 7: Unitary matrix of Be atom HF orbitals. Vertical axis: atomic orbitals; Horizontal axis: Hartree-Fock orbitals

Figure 8: $C_{\{\text{Be in Be}_2\}}$ matrix. Two same unitary matrices stacked together. The unitary matrix is for Be atom Hartree-Fock orbitals. Vertical axis: atomic orbitals for Be atom A and Be atom B; Horizontal axis: Hartree-Fock orbitals
Figure 9: Unitary matrix of $\text{Be}_2$ HF orbitals. Vertical axis: atomic orbitals for atom A and atom B; Horizontal axis: Be dimer Hartree-Fock orbitals

Figure 10: Molecular orbital overlap matrix for beryllium atom and beryllium dimer.
The matrix $\tilde{S}_{\text{Be,Be}_2}$ has a rectangular (not square) shape. The vertical axis corresponds to the one-fragment HF orbital basis while the horizontal axis represents the two-fragment HF orbital basis. The $\tilde{S}_{\text{Be,Be}_2}$ is shown in Fig. 10.

With this overlap matrix one can transform all the monomer FCI eigenstates from one-fragment HF orbital basis into two-fragment HF orbital basis by using the following equations.

\[ |\Phi_{j'}^{\text{Be}}\rangle = \sum_j \det U_{j'j} |\Phi_j^{\text{Be}_2}\rangle \]  

(5.2)

Here $j$ and $j'$ are collective indices representing one-fragment HF orbital basis and two-fragment HF orbital basis, respectively. They can be defined as the following,

\[ |\Phi_{j'}^{\text{Be}}\rangle = |p'q' \cdots r' \cdots s'\rangle \]  

(5.3)

\[ |\Phi_j^{\text{Be}_2}\rangle = |pq \cdots r \cdots s\rangle \]  

(5.4)

where the $|p, q, \cdots, r, \cdots, s\rangle$ is a determinant wavefunction in two-fragment HF orbital basis. and the $|p', q', \cdots, r', \cdots, s'\rangle$ denotes a determinant wavefunction in one-fragment HF orbital basis. The determinant $U_{j'j}$ is given as,

\[ \det U_{j'j} = \begin{vmatrix} \langle p|p' \rangle & \cdots & \langle p|s' \rangle \\ \vdots & \ddots & \vdots \\ \langle s|p' \rangle & \cdots & \langle s|s' \rangle \end{vmatrix} \]  

(5.5)

The values of any determinant element, e.g., the $\langle p|p' \rangle$, are taken from the overlap matrix $\tilde{S}_{\text{Be,Be}_2}$. Then any Be atom FCI state $i$ can be expanded in the two-fragment HF orbital basis as,

\[ |\Psi_i^{\text{Be FCI}}\rangle = \sum_{j'} C_{j'i}^{\text{FCI}} |\Phi_{j'}^{\text{Be}}\rangle = \sum_{j'} C_{j'i}^{\text{FCI}} \sum_j \det U_{j'j} |\Phi_j^{\text{Be}_2}\rangle \]  

(5.6)

The $C_{j'i}$ represents the Be atom FCI eigenvector elements for $i$-th eigenstate. Now the transformation for non charge transfer wavefunctions is complete. The FCI eigenstates of Be$^+$ and Be$^-$ are also transformed in a similar fashion before building charge transfer wavefunctions.
**Construction of dimer wavefunctions.** Monomer tensor products now can be resolved in dimer molecular orbital basis. In total there are six different two-fragment super-system wavefunctions to be constructed. The very first one is the non charge transfer super-system wavefunction. It is just a simple tensor product of FCI eigenstates of two Be atoms, which is shown in eq. (5.7) through eq. (5.9).

\[
|\psi_{ij}^{\text{Be}_2}\rangle = \hat{A}[|\psi_i^{\text{Be}_A}\rangle \otimes |\psi_j^{\text{Be}_B}\rangle] = \hat{A} \left[ \sum_{i=(p_1p_2p_3p_4)} C_p^i |p_1 p_2 p_3 p_4\rangle \right] \left[ \sum_{j=(q_1q_2q_3q_4)} C_Q^j |q_1 q_2 q_3 q_4\rangle \right] (5.7)
\]

\[
= \sum_{i=(p_1p_2p_3p_4)} \sum_{j=(q_1q_2q_3q_4)} C_p^i C_Q^j (-1)^k \text{sorted}(p_1 p_2 p_3 p_4 q_1 q_2 q_3 q_4) (5.8)
\]

where the \(p\) and \(q\) denote different orbitals from two-fragment HF basis. \(\hat{A}\) is the antisymmetrizer and \(k\) is the number of swaps needed for sorting the \(p\) and \(q\) orbitals to achieve an ascending order. The \(C\) represents the Be atom FCI eigenvector elements.

The second type of super-system wavefunctions simulates single charge transfer between two fragments. It is built from the tensor product of FCI eigenstates of Be\(^+\) and Be\(^-\), which is shown in eq. (5.10) through eq. (5.12). The \(C\) here represents the Be\(^+\) or Be\(^-\) FCI eigenvector elements. The reversed order tensor product, Be\(^-\) and Be\(^+\), are needed as well. They can be built similarly by constructing

\[
|\psi_{ij}^{\text{Be}^-\text{Be}^+}\rangle = \hat{A}[|\psi_i^{\text{Be}^-}\rangle \otimes |\psi_j^{\text{Be}^+}\rangle].
\]

\[
|\psi_{ij}^{\text{Be}^+\text{Be}^-}\rangle = \hat{A}[|\psi_i^{\text{Be}^+}\rangle \otimes |\psi_j^{\text{Be}^-}\rangle] = \hat{A} \left[ \sum_{i=(p_1p_2p_3)} C_p^i |p_1 p_2 p_3\rangle \right] \left[ \sum_{j=(q_1q_2q_3q_4)} C_Q^j |q_1 q_2 q_3 q_4\rangle \right] (5.10)
\]

\[
= \sum_{i=(p_1p_2p_3)} \sum_{j=(q_1q_2q_3q_4q_5)} C_p^i C_Q^j (-1)^k \text{sorted}(p_1 p_2 p_3 q_1 q_2 q_3 q_4 q_5) (5.11)
\]

Other four cases of charge transfer super-system wavefunctions are listed in
eq. (5.13) through eq. (5.16), whose reversed order tensor products are also needed if two fragment wavefunctions are different.

\begin{align*}
|\Psi_{ij}^{\text{Be}^+\text{Be}^+}\rangle &= \hat{A}[(|\psi_i^{\text{Be}^+}\rangle \otimes |\psi_j^{\text{Be}^+}\rangle)] \\
|\Psi_{ij}^{\text{Be}^+\text{Be}^-}\rangle &= \hat{A}[(|\psi_i^{\text{Be}^+}\rangle \otimes |\psi_j^{\text{Be}^-}\rangle)] \\
|\Psi_{ij}^{\text{Be}^-\text{Be}^-}\rangle &= \hat{A}[(|\psi_i^{\text{Be}^-}\rangle \otimes |\psi_j^{\text{Be}^-}\rangle)] \\
|\Psi_{ij}^{\text{Be}^-\text{Be}^+}\rangle &= \hat{A}[(|\psi_i^{\text{Be}^-}\rangle \otimes |\psi_j^{\text{Be}^+}\rangle)]
\end{align*}

\textbf{Dimer Interaction Terms}

Now having described the construction of dimer states as tensor products of monomer states, we may proceed with the construction of matrix elements, which requires the actions of the Hamiltonian, resolved in the dimer orbital basis.

Typically, a Hamiltonian can be represented in different ways. One way that has already been shown is the matrix representation in the CI section. However, the matrix representation is in practice very resource hungry if the Hamiltonian is built explicitly in computer memory. And explicit FCI Hamiltonian matrices cannot be built for large chemical systems. Therefore, the function-like Hamiltonian representation must be used. This representation is similar to the effect of a Hamiltonian operator acting onto a wavefunction. The function-like form would trade code execution speed for consuming less computer memory which cannot be expanded infinitely. But with modern processors running at several GHz, this is not a big concern as before.

The X-CCSD(2) Hamiltonian explored in this work has only up to two-fragment interaction terms and it is denoted by "(2)" here. It is due to the ease of implementation to have two-fragment Hamiltonian terms at the moment. Besides, dipole–dipole coupling happens mostly in the cases that two fragments fluctuate and interact, which is addressed with the two-fragment Hamiltonian terms. Three-fragment
and four-fragment Hamiltonian terms will be added in the future to achieve better computational accuracy.

In X-CCSD(2) Hamiltonian, due to the presence of the large number of FCI eigenstates, the Hamiltonian action algorithms must be optimized to speed up executions. The most conceptually simple Hamiltonian action can be realized by looping over all possible FCI states and apply the electronic Hamiltonian onto each of them to obtain a new FCI vector element. However this is a numerical disaster. The Hamiltonian matrix, if ever built, is very sparse (most of the values are zero). All we need here is to loop over the set of significant matrix values, hopefully the set is small, and let them act on limited numbers of non-vanishing cases to generate possible significant updates for vector elements rather than blindly computing an enormous amount of zero terms. This algorithm of Hamiltonian action can be implemented by first finding the significant values with a pre-defined threshold ($V_{pqrs} > 10^{-6}$) in electronic repulsion matrix $V$ and having them saved into a list-like data structure. The corresponding indices are also saved for later reference. Then the code iterates over all the significant $V$ and loops over all potentially non-vanishing FCI states by loading up saved indices. The code uses another threshold checking in which the absolute value of FCI eigenvector $C$ from eq. (5.6) must be greater than the preset threshold as well. Next, the code proceeds to compute the two electron repulsion value for current orbitals and the phase for the current term. The index of sorted wavefunction is computed to have the calculation results accumulated into the correct locations in the vector. These procedures can be performed on the core Hamiltonian $\hat{h}$ as well. By using this algorithm, the Hamiltonian action code now runs much more efficient than doing primitive loops. It is true that there still exists a lot of sparse elements but they will be identified and skipped in later implementations.
Optimal Monomer States

In practice, it has been found that the number of monomer FCI states necessary to achieve good results is impractical. This was somewhat anticipated as the best states to describe an interaction are not necessarily the best description of isolated excitations.

With the dimer Hamiltonian in hand, already resolved in a monomer tensor product basis, it is at liberty to solve for the dimer ground state at the nearest-neighbor distance. The monomer excitations most highly involved in this converged interaction then form an optimal many-electron basis for the monomers, in which to describe the dimer interaction, hypothetically at any distance.

In order to isolate the most important monomer states, which are linear combinations of the FCI eigenstates, the diagonalization is carried out for the averaged reduced density matrix built from two beryllium atoms as shown in eq. (5.17) and applying a transformation of basis onto primitive fragment FCI states, eq. (5.18).

\[
U^\rho + \left[ \frac{\rho_L^{(1)} + \rho_R^{(1)}}{2} \right] U^\rho = E \tag{5.17}
\]

\[
|\Psi_i^{\text{Be FCI}}\rangle = \sum_k U_{ki}^\rho |\Psi_k^{\text{Be FCI}}\rangle \tag{5.18}
\]

The \( \rho_L^{(1)} \) is the reduced density matrix for the fragment on the left side and \( \rho_R^{(1)} \) for the fragment on the right side. The \( U^\rho \) is the unitary matrix from this diagonalization. The matrix \( E \) holds all the eigenvalues from this diagonalization. These eigenvalues are the importance (or probabilities) of each reconditioned fragment state contributing to the fragment fluctuations. For the Be\(_n\) in-a-chain problem these states are unbiased through the averaging of the reduced density matrices with the left and the right neighbors. With this transformation, the number of states was cut down to 30 for every beryllium fragment.
Results and Discussion

The comparison curves of traditional electronic correlation methods are obtained by the Psi4 quantum chemistry package. [81] The Psi4 package is an open-source highly optimized production code that specializes in coupled-cluster theory and symmetry-adapted perturbation theory. The performance of the loosely optimized X-CCSD(2) pilot code is compared against the highly optimized Psi4 code in both energy modeling accuracy and computational cost.

The first test case is the reconstruction of the Be$_2$ dissociation potential using X-CCSD(2) method. In Fig. 11, the horizontal axis indicates the distance between two Be atoms, and the vertical axis represents the Be$_2$ system energy. The dissociation potential is scanned from 3 Å to 10 Å. The 6-31G basis set is used for all the calculations in this figure. The CCSD, CCSD(T) and FCI dissociation curves are also plotted. The X-CCSD(2) method happens to be the exact solution to the Be$_2$ problem under 6-31G basis. This is due to the two-fragment Hamiltonian terms implemented in X-CCSD(2) method and also the exact FCI eigenstates used for each fragment. Therefore the X-CCSD(2) curve matches the FCI curve exactly. However if the charge transfer states are turned off, the resulting X-CCSD(2) potential is the curve on the top in Fig. 12, which turns out to be a repulsive potential. This explains the previous statement that the charge transfer effect makes major contribution to the Be$_2$ bonding.

The Be$_3$ results resolved by the X-CCSD(2) method still exhibit satisfying accuracy as shown in Fig. 13. The horizontal axis represents the distance between adjacent Be atoms in Ångströms. The vertical axis shows the calculated super-system energies. The trend of these curves look reasonable. Pulling away the two Beryllium atoms from both sides results in a rise in total energy. In this test, the excitonic full coupled-cluster theory [X-FCC(2)] with two-fragment Hamiltonian terms, curve is also computed for elucidating the decomposition of the remaining error,
Figure 11: The Be dimer dissociation potential modeled by X-CCSD(2), FCI, CCSD and CCSD(T) methods.

Figure 12: The Be dimer dissociation potential modeled by X-CCSD(2) method without charge transfer. The black curve is the Be dimer dissociation potential modeled by FCI method. The dotted curve is by the X-CCSD(2) method with charge transfer turned off. The big discrepancy indicates that charge transfer effect contributes to the majority of the bonding interaction.
since in this calculation we have the exact wavefunction for an approximate Hamiltonian. The X-FCC(4) method with up to four-body Hamiltonian terms would be the exact solution in this case since trimer terms are included. The X-CCSD(2) method demonstrates an accuracy of roughly CCSD(T) level at all distances in the plot. Remarkably, beyond 10 fragments, the X-CCSD(2) method still exhibits a matching energy accuracy with electronic CCSD method. For systems below 10-fragment level, the X-CCSD(2) demonstrates a better accuracy than electronic CCSD method.

The largest test case in this work is a Be$_{100}$ system with the 100 fragments even spaced at a fixed distance of 4.5 Å. The 4.5 Å is the equilibrium bond distance calculated at FCI/6-31G level. This is a 100-fragment X-CCSD(2) calculation that runs on one CPU core and takes only 7 hours while maintaining a traditional electronic CCSD level of accuracy. This result is truly phenomenal. The timing comparisons are plotted in Fig. 14 with number of fragments up to thirty.
Figure 14: The scaling comparisons between X-CCSD(2) method and traditional CCSD and CCSD(T) methods.

The reason for not showing any larger system is that most of the quantum chemistry packages used for Be\textsubscript{n} calculations did not succeed beyond 30-fragment level. However, X-CCSD(2) can handle 100 fragments with ease. Once again, the capability of X-CC theory for handling large molecular systems is demonstrated. The X-CCSD(2) code runs slower than production electronic CCSD code in Fig. 14. This is due to the fact that the X-CCSD(2) has not yet been optimized. Even with this pilot X-CCSD(2) code, the X-CCSD(2) computational cost would become cheaper than extrapolated cost for production electronic CCSD code beyond 40-fragment level. Indeed, the most important factor in a scaling problem is the power for system size \( N \). The power is inherently fixed due to method complexity and cannot be easily reduced. With a smaller pre-factor, the X-CCSD(2) would surely be more cost-effective than both CCSD(T) and CCSD production code already at small system sizes.
In future work, more efforts would be put into optimizing current X-CCSD(2) code which has a great potential to execute much faster than the current timings. The scaling of X-CCSD(2) can be further improved to a probable second order by applying mathematical techniques.
Chapter 6: Outlook

The X-CC method displayed great capability in handling large molecular systems with limited computer resources and low excitation levels. Since X-CC theory operates on the collective coordinates of fragment fluctuations, the algorithm that works on beryllium problem can be extended to work with general inhomogeneous super-system problems. However, the equation-of-motion coupled-cluster (EOM-CC) theory [75] may have to be used to obtain fragment excited states. This is due to the fact that the FCI method has an extremely high computational cost.

With a loosely optimized code, the X-CCSD(2) scheme is capable of simulating the 100-fragment system with ease. It is worth noting that, the modeling of beryllium atoms is a very difficult case due to the presence of many near degenerate orbitals. These degeneracies give the beryllium problem a multi-reference nature. If the X-CCSD(2) method were applied to molecules with big HOMO/LOMO (highest unoccupied molecular orbital/lowest occupied molecular orbital) energy gaps, for example, a water molecule, one would expect that a smaller number of fragment fluctuations is required for each fragment or effectively the X-CC scheme may be able to handle more water molecules inside the same calculation. The X-CC scheme also deals with charge transfer and bond breaking processes. Therefore, the possible applications for X-CC theory are many. To name a few of them, the ligand-protein binding modeling, the enzymatic catalysis and the electronically excited solvents studies. They all require an accurate modeling of small energy differences inside large chemical systems and a quantum mechanical treatment to fundamentally quantum mechanical electrons, which forms the main goal of the X-CC theory.
Interestingly, it is possible to apply X-CC method within the QM region in a QM/MM calculation. The QM region would receive an size increase and/or a better accuracy if similar computational cost is paid for as traditional electronic methods. The QM/MM scheme would further reduce the computational cost for molecular simulations of large systems.
REFERENCES


DOI:10.1021/acs.jctc.7b00174.
APPENDIX A: X-CCSD(2) AMPLITUDE EQUATIONS

We begin solving excitonic coupled-cluster amplitude equations by defining operators that have properties closely related to normal ordering of field operators (under the mapping $\hat{\tau}_i^j \rightarrow \hat{a}_i^\dagger \hat{a}_j$, with restrictions on index pairs $i,j$).

\begin{align*}
\hat{c}_{um}^{0m} &= \hat{\tau}_{um}^{0m} \\
\hat{f}_{om}^{0m} &= 1 - \hat{\tau}_{om}^{0m} \\
\hat{f}_{vm}^{vm} &= \hat{\tau}_{vm}^{vm} \\
\hat{d}_{om}^{um} &= -\hat{\tau}_{om}^{um}
\end{align*}

(6.1)

The indices $u_m, v_m, w_m, x_m$ refer to any state of fragment $m$ except the reference state, $o_m$. These letters have been chosen to be reminiscent of the “occupied” and “unoccupied/virtual” nomenclature familiar from conventional CC theory, and yet be distinct from it in notation. Similarly, the letters used for the four different types of operators stand for “excitation,” “flat,” and “de-excitation.” Of primary importance is that all operators of the flat and de-excitation type produce the null state when acting on the many-fragment reference. The sign convention shown has been chosen for convenience (fewest sign changes in derivations).

Any Hamiltonian $\hat{H}$ originally expressed in terms of the set of fluctuation operators $\{\hat{\tau}_i^j\}$ may be expressed in terms of these new operators by simple substitution. In general, it will be valuable to consider a generalized definition of normal ordering for strings of such operators, in which any excitation operators appear to the left of all other operators. For the Hamiltonian this is trivial; no string in the Hamiltonian contains more than one operator that acts on any given fragment, such
that all operators in any Hamiltonian string commute. The substitution of \( \hat{\tau}_{o_m} \) by \( 1 - \hat{f}_{o_m} \) does introduce a constant into the normal-ordered Hamiltonian expansion, however, which is equal to \( \langle \Psi^O|\hat{\mathcal{H}}|\Psi_O \rangle \), in analogy to the effect of normal ordering of conventional field operators.

The cluster operator \( \hat{T} \), which must be repeatedly commuted with \( \hat{\mathcal{H}} \), is obviously composed only of operators of the excitation class. The following commutators (special cases of eqnref?) will therefore be useful

\[
\begin{align*}
[\hat{e}_{um}, \hat{e}_{wm}] &= 0 \\
[\hat{f}_{o_m}, \hat{e}_{wm}] &= \hat{e}_{wm} \\
[\hat{f}_{v_m}, \hat{e}_{wm}] &= \hat{e}_{wm} \delta_{vm,wm} \\
[\hat{d}_{o_m}, \hat{e}_{wm}] &= \hat{f}_{wm} + \hat{f}_{o_m} \delta_{um,wm} - \delta_{um,wm}
\end{align*}
\]

(6.2)

where we recall that commutators between operators belonging to different fragments are always zero. This now makes explicit earlier arguments concerning truncation of the BCH expansion, in that repeated commutations of any operator with excitations always ends with zero, and no operator survives more than two such nested commutations.

In order to be generic at some points, we will let \( \hat{g}_m \) represent an arbitrary single-fragment operator on fragment \( m \). This could be any one of \( \hat{e}_{o_m} \), \( \hat{f}_{o_m} \), \( \hat{f}_{v_m} \), or \( \hat{d}_{o_m} \), or linear combinations thereof, and we also permit linear combinations that include constants to be denoted as such. We then furthermore define the abbreviations

\[
\begin{align*}
\hat{g}_m^{[wm]} &= [\hat{g}_m, \hat{e}_{wm}] \\
\hat{g}_m^{[wm] [x_m]} &= [[\hat{g}_m, \hat{e}_{wm}], \hat{e}_{x_m}]
\end{align*}
\]

(6.3)

noting that these are themselves single-fragment operators (for purposes of recursion). Since an arbitrary \( \hat{g}_m \) may be decomposed as a linear combination of \( \hat{e}_{o_m} \), \( \hat{f}_{o_m} \), \( \hat{f}_{v_m} \), \( \hat{d}_{o_m} \), and a constant, by the forgoing arguments, all nested commutations
higher than third order are zero. Using this notation, combined with recursion of
the well-known formula for commutation with a simple operator product, we then
resolve commutation of a single-fragment operator with a string of $M$ excitations as

$$[\hat{g}_m, \hat{e}_{w_{m_1}}^{o_{m_1}} \cdots \hat{e}_{w_{m_M}}^{o_{m_M}}] = \sum_{i=1}^M \delta_{m,m_i} \hat{e}_{w_{m_1}}^{o_{m_1}} \cdots \hat{e}_{w_{m_{i-1}}}^{o_{m_{i-1}}} \hat{e}_{w_{m_{i+1}}}^{o_{m_{i+1}}} \cdots \hat{e}_{w_{m_M}}^{o_{m_M}} \hat{g}_m$$

(6.4)

under the condition that the indices $m_1, \cdots m_M$ identify distinct fragments, thus
allowing rearrangement of the operator strings. Also, in this case, maximally one
term of this expansion is nonzero.

We now write the cluster operator $\hat{T} = \sum_M \hat{T}_M$ as a summation over all exci-
tation orders $M \geq 1$ present in the Ansatz, where

$$\hat{T}_M = \sum_{m_1} \sum_{m_2 > m_1} \cdots \sum_{m_M > m_{M-1}} \left( \sum_{w_{m_1}} \cdots \sum_{w_{m_M}} t_{m_1}^{w_{m_1}} \cdots t_{M}^{w_{m_M}} \hat{e}_{w_{m_1}}^{o_{m_1}} \cdots \hat{e}_{w_{m_M}}^{o_{m_M}} \hat{g}_m \right)$$

$$= \frac{1}{M!} \sum_{m_1} \cdots \sum_{m_M} \left( \sum_{w_{m_1}} \cdots \sum_{w_{m_M}} t_{m_1}^{w_{m_1}} \cdots t_{M}^{w_{m_M}} \hat{e}_{w_{m_1}}^{o_{m_1}} \cdots \hat{e}_{w_{m_M}}^{o_{m_M}} \hat{g}_m \right)$$

(6.5)

In the first line, we sum over all unique $M$-tuples of fragments, and in the second
line we account for redundancy by dividing by $M!$ and insisting that the tensor of
amplitudes $t_M$ is invariant with respect to all index permutations. We may also
insist that an amplitude is zero if any two indices belong to the same fragment, ef-
fectively removing from further consideration those operator strings in which the
same fragment occurs twice (though this is not strictly necessary, since such strings
are themselves zero). We then arrive at

$$[\hat{g}_m, \hat{T}_M] = \frac{1}{M!} \sum_{m_1} \cdots \sum_{m_M} \left( \sum_{w_{m_1}} \cdots \sum_{w_{m_M}} t_{m_1}^{w_{m_1}} \cdots t_{M}^{w_{m_M}} \hat{g}_m \hat{e}_{w_{m_1}}^{o_{m_1}} \cdots \hat{e}_{w_{m_M}}^{o_{m_M}} \hat{g}_m \right)$$

$$= \frac{1}{(M-1)!} \sum_{m_1 \neq m} \cdots \sum_{m_{M-1} \neq m} \left( \sum_{w_{m_1}} \cdots \sum_{w_{m_{M-1}}} \sum_{w_{m}} t_{m_1}^{w_{m_1}} \cdots t_{M-1}^{w_{m_{M-1}} w_{m}} \hat{e}_{w_{m_1}}^{o_{m_1}} \cdots \hat{e}_{w_{m_{M-1}} w_{m}}^{o_{m_{M-1}}} \hat{g}_m \right)$$

$$= \sum_{w_{m}} \hat{T}_{M-1}^{w_{m}} \hat{g}_m$$
with

\[ T_{M}^{w_m} = \frac{1}{M!} \sum_{m_1 \neq m} \cdots \sum_{m_M \neq m} \left( \sum_{w_{m_1}} \cdots \sum_{w_{m_M}} t_{M+1}^{w_{m_1} \cdots w_{m_M} w_m} e_{w_{m_1}} \cdots e_{w_{m_M}} \right) \]

\[ = \frac{1}{M!} \sum_{m_1} \cdots \sum_{m_M} \left( \sum_{w_{m_1}} \cdots \sum_{w_{m_M}} t_{M+1}^{w_{m_1} \cdots w_{m_M} w_m} e_{w_{m_1}} \cdots e_{w_{m_M}} \right) \]  

(6.6)

and

\[ \hat{T}_0^{w_m} = t_1^{w_m} \]

(6.7)

The logical process by which this is deduced is to decompose each of the \( M \) summations over all fragments in the first line into one component for fragment \( m \) and a summation over all fragments other than \( m \). This gives a total of \( 2^M \) terms when the resulting \( M \)-fold binary product is expanded, one in which all indices equal \( m \), one in which none equal \( m \), and all other possible combinations. Only \( M \) of these terms survive, since the amplitudes are zero if any two indices both belong to fragment \( m \) and the commutator is zero if no fragments in the summations are equal to \( m \); therefore, only one fragment can be equal to \( m \) in any surviving term, of which there are \( M \) choices. Each surviving term contains \( M - 1 \) summations over the other fragments. Since all of the other fragments and indices are summed over, and since all of the operators in the strings commute and amplitudes are permutationally symmetric, these \( M \) terms are all identical, simply reducing the prefactor from \( 1/M! \) to \( 1/(M - 1)! \) for the single such term written explicitly in the second line.

In the last line, the restrictions on the summations is removed since the amplitudes of the superfluous terms introduced are zero. This has the advantage of giving \( \hat{T}_M^{w_m} \) an identical structure to \( \hat{T}_M \) (unless it is a constant), but with different amplitudes, permitting us to write

\[ [\hat{g}_m, \hat{T}_M^{w_{m'}}] = \sum_{w_m} \hat{T}_{M-1}^{w_m w_{m'}} \hat{g}_m^{w_m} \]
with
\[
\hat{a}_{m}' = \frac{1}{M!} \sum_{m_1} \cdots \sum_{m_M} \left( \sum_{w_{m_1}} \cdots \sum_{w_{m_M}} t_{M+2}^{w_{m_1} \cdots w_{m_M} w_{m}' \cdots w_{m_M}} e_{w_{m_1}} \cdots e_{w_{m_M}} \right) \quad (6.8)
\]
and
\[
\hat{a}_{0}' = t_{2}^{w_{m} w_{m}'} \quad (6.9)
\]
For consistency, both \(\hat{a}_{M}'\) and \(\hat{a}_{0}'\) are zero for \(M < 0\), but the former will never arise in the derivation. This recursion will be necessary later, in order bring the transformation of products of operators into generalized normal-ordered form.

It will now be expedient to define the following
\[
\hat{a}_{m}' = \sum_{M} \hat{a}_{M-1}^{w_{m}}
\]
\[
\hat{a}_{m} w_{m}' = \sum_{M} \hat{a}_{M-2}^{w_{m} w_{m}'}
\]
where the summation over \(M\) is over all orders originally in the user Ansatz. These allows us to write more compactly
\[
[g_{m}, \hat{a}] = \sum_{m} \hat{a}^{g_{m}}\]
\[
[g_{m}, \hat{a} w_{m}'] = \sum_{m} \hat{a}^{g_{m}}\]
(6.11)

By recursion, we then also arrive at
\[
[[g_{m}, \hat{a}], \hat{a}] = \sum_{m} \sum_{x_{m}} \hat{a}^{g_{m}}\hat{a}^{x_{m}} g_{m}^{[}\bar{a}_{m} |x_{m}]\]
(6.12)
where it is clear that any triply nested commutator vanishes. This allows us to then use the BCH expansion to finally write, for any single-fragment operator,
\[
e^{-\hat{a}} g_{m} e^{\hat{a}} = g_{m} + \sum_{m} \hat{a}^{g_{m}} + \frac{1}{2} \sum_{m} \sum_{x_{m}} \hat{a}^{g_{m}}\hat{a}^{x_{m}} g_{m}^{[}\bar{a}_{m} |x_{m}]\]
(6.13)

Importantly, as a consequence of the restrictions on the indices in \(\hat{a}\) referring to distinct fragments in each string, each term in this expansion is already in generalized normal-ordered form.

The forgoing suffices to perform the similarity transformation of the single-fragment part of the Hamiltonian and it can also be used to construct expressions
for the higher-fragment-order terms. First, we note that any operator \(\hat{o}\) may be decomposed as \((\hat{o})_x + (\hat{o})_o\), such that,

\[
(\hat{o})_x|\Psi_O\rangle = \hat{o}|\Psi_O\rangle \\
(\hat{o})_o|\Psi_O\rangle = 0
\]

Although such a decomposition is not unique as described, if only the action upon \(|\Psi_O\rangle\) is relevant for a specific purpose, then any convenient such partitioning will suffice. If \(\hat{o}\) is already written as a linear combination of normal-ordered strings, the straightforward choice of \((\hat{o})_o\) consists of summing all such terms whose string contains at least one operator that is not an excitation (necessarily to the right of the other operators). The corresponding choice of \((\hat{o})_x\) then consists of the remaining terms, \(i.e.,\) constants and strings of excitations only. For convenience, we also allow commutator brackets to be subscripted as \([,]_x\), indicating that only the constant and excitation part of the normal-ordered form of the result is retained. The central task in a given CC iteration may then be framed in terms of the operator \(\hat{\Omega} = (e^{-\hat{T}\hat{H}e^{\hat{T}}} )_x\). The constant part of \(\hat{\Omega}\) is the CC pseudo-energy, which can be written as \(\langle \Psi^O|\hat{\Omega}|\Psi_O\rangle\), and the projections \(\langle \Psi^I|\hat{\Omega}|\Psi_O\rangle\) are used to define the iterative update to \(\hat{T}\) for \(I\neq O\) in the user-specified Ansatz.

We will then proceed to decompose \(\hat{\Omega}\) for the individual terms in the Hamiltonian. Using the abbreviated notation \(\tilde{g}_m = e^{-\hat{T}\hat{g}_m e^{\hat{T}}}\), we then have for interaction
terms involving up to the maximum number (four) of single-fragment operators

\[
(e^{-\hat{T}} g_m e^{\hat{T}})_x = (\hat{g}_m)_x
\]

\[
(e^{-\hat{T}} g_{m_1} g_{m_2} e^{\hat{T}})_x = (\hat{g}_{m_1} \hat{g}_{m_2})_x
\]

\[
= (\langle (\hat{g}_{m_1})_x + (\hat{g}_{m_1}_o)(\hat{g}_{m_2})_x \rangle_x
\]

\[
= (\langle \hat{g}_{m_1})_x(\hat{g}_{m_2})_x + (\hat{g}_{m_1}_o)(\hat{g}_{m_2})_x \rangle_x
\]

\[
= (\hat{g}_{m_1})_x(\hat{g}_{m_2})_x + [(\hat{g}_{m_1}_o) (\hat{g}_{m_2})_x]_x
\]

\[
(e^{-\hat{T}} g_{m_1} g_{m_2} g_{m_3} e^{\hat{T}})_x = (\hat{g}_{m_1})_x(\hat{g}_{m_2})_x(\hat{g}_{m_3})_x
\]

\[
+ (\hat{g}_{m_1})_x[(\hat{g}_{m_2}_o) (\hat{g}_{m_3})_x]_x
\]

\[
+ [(\hat{g}_{m_1}_o) (\hat{g}_{m_2})_x(\hat{g}_{m_3})_x]_x
\]

\[
+ [(\hat{g}_{m_1}_o) [(\hat{g}_{m_2}_o) (\hat{g}_{m_3})_x]_x]_x
\]

\[
(e^{-\hat{T}} g_{m_1} g_{m_2} g_{m_3} g_{m_4} e^{\hat{T}})_x
\]

\[
= (\hat{g}_{m_1})_x(\hat{g}_{m_2})_x(\hat{g}_{m_3})_x(\hat{g}_{m_4})_x
\]

\[
+ (\hat{g}_{m_1})_x(\hat{g}_{m_2})_x[(\hat{g}_{m_3})_o (\hat{g}_{m_4})_x]_x
\]

\[
+ (\hat{g}_{m_1})_x[(\hat{g}_{m_2}_o)(\hat{g}_{m_3})_x(\hat{g}_{m_4})_x]_x
\]

\[
+ (\hat{g}_{m_1})_x[(\hat{g}_{m_2}_o)(\hat{g}_{m_3})_x(\hat{g}_{m_4})_x]_x
\]

\[
+ [(\hat{g}_{m_1}_o) (\hat{g}_{m_2}_o)[(\hat{g}_{m_3})_o (\hat{g}_{m_4})_x]_x]_x
\]

\[
+ [(\hat{g}_{m_1}_o) [(\hat{g}_{m_2}_o) (\hat{g}_{m_3})_x(\hat{g}_{m_4})_x]_x]_x
\]

\[
\hspace{1cm} (6.15)
\]

The results for trimers and tetramers are obtained by recurring the procedure shown for the dimer term. The logic in resolving the dimer term is as follows. After inserting \(1 = e^{\hat{T}} e^{-\hat{T}}\) between the two single-fragment operators, each of the resulting transformed operators is trivially divided into the parts that do and do not destroy the reference. Only the \(()_x\) part of the right-most operator needs to be retained, since inclusion of the \(()_o\) part simply results in additional normal-ordered terms that all destroy the reference, and therefore do not survive the outermost retention.
of only non-reference-destroying terms. Likewise, inclusion of the second term of
the commutator shown does not change anything, since it consists only of reference-
destroying terms that are not retained. However, expressing the result in terms of
this commutator will prove valuable; since the second argument to the commutator
consists only of constants and excitation strings, fragment rank is thereby reduced.

We will confine our attention here to the dimer Hamiltonian terms, since those
were implemented for this work, and since the procedure for higher-fragment-order
terms is a simple repetition of this. For dimer interactions we have 10 classes of
terms in the normal-ordered Hamiltonian, if we arbitrarily decide that normal or-
dering also involves having any de-excitation operators to the far right and having
any virtual-rearrangement flat operators to the right of any reference-hole-check flat
operators. We now require explicit forms of these transformed operators. Again us-
ing an overbar to denote the similarity transformed operator, we have

\[\bar{e}_{\nu m} = \hat{e}_{\nu m}\]
\[\bar{f}_{\omega m} = \hat{f}_{\omega m} + \sum_{w m} \hat{T}_{w m} \hat{e}_{w m}\]
\[\bar{v}_{\nu m} = \hat{v}_{\nu m} + \hat{T}_{v m} \hat{e}_{\nu m}\]
\[\bar{d}_{\omega m} = \hat{d}_{\omega m} + \hat{T}_{d m} \hat{f}_{\omega m} + \sum_{w m} \hat{T}_{w m} \hat{f}_{w m} + \hat{T}_{w m} \left( \left( \sum_{w m} \hat{T}_{w m} \hat{e}_{w m} \right) - 1 \right)\]

(6.16)

each of which is easily divided into an excitation part (the latter or sole term) and
another reference-destroying term (or terms). The commutators that are needed to
evaluate the dimer interaction terms are then

\[
\begin{align*}
[\{\epsilon_{um_1}\}_{o}, \{\epsilon_{um_2}\}_{x}]_x &= 0 \\
[\{\epsilon_{um_1}\}_{o}, \{\bar{\epsilon}_{um_2}\}_{x}]_x &= 0 \\
[\{\epsilon_{um_1}\}_{o}, \{\bar{\epsilon}_{um_2}\}_{x}]_x &= 0 \\
[\{\epsilon_{um_1}\}_{o}, \{d_{um_2}\}_{x}]_x &= 0 \\
[\{\bar{\epsilon}_{um_1}\}_{o}, \{\bar{\epsilon}_{um_2}\}_{x}]_x &= \sum_{\bar{um}_1, \bar{um}_2} \hat{T}^{\bar{um}_1, \bar{um}_2} \epsilon_{um_1} \epsilon_{um_2} \\
[\{\bar{\epsilon}_{um_1}\}_{o}, \{\bar{\epsilon}_{um_2}\}_{x}]_x &= -\sum_{\bar{um}_1} \hat{T}^{\bar{um}_1, \bar{um}_2} \epsilon_{um_1} \epsilon_{um_2} + \sum_{\bar{um}_2} \left( \hat{T}^{\bar{um}_1, \bar{um}_2} \hat{T}^{\bar{um}_1, \bar{um}_2} \right) \epsilon_{um_1} \epsilon_{um_2} \\
[\{f_{um_1}\}_{o}, \{f_{um_2}\}_{x}]_x &= \hat{T}^{\bar{um}_1, \bar{um}_2} \epsilon_{um_1} \epsilon_{um_2} \\
[\{f_{um_1}\}_{o}, \{\bar{f}_{um_2}\}_{x}]_x &= -\hat{T}^{\bar{um}_1, \bar{um}_2} \epsilon_{um_1} + \sum_{\bar{um}_2} \left( \hat{T}^{\bar{um}_1, \bar{um}_2} \hat{T}^{\bar{um}_1, \bar{um}_2} \right) \epsilon_{um_1} \epsilon_{um_2} \\
[\{d_{um_1}\}_{o}, \{d_{um_2}\}_{x}]_x &= -\sum_{\bar{um}_1} \left( \hat{T}^{\bar{um}_1} \hat{T}^{\bar{um}_1} \right) \epsilon_{um_1} \epsilon_{um_2} - \sum_{\bar{um}_2} \left( \hat{T}^{\bar{um}_1} \hat{T}^{\bar{um}_1} \right) \epsilon_{um_1} \epsilon_{um_2} + \sum_{\bar{um}_1, \bar{um}_2} \left( \hat{T}^{\bar{um}_1, \bar{um}_2} \hat{T}^{\bar{um}_1, \bar{um}_2} \right) \epsilon_{um_1} \epsilon_{um_2}
\end{align*}
\]
assuming that $m_1 \neq m_2$. We now further confine our attention to the X2-CCSD model, for which we have

$$
\hat{T}^{w_m} = t_1^{w_m} + \sum_{m'} \sum_{w_{m'}} t_2^{w_{m'} w_m} c_{w_{m'}}^{m'}
$$

$$
\hat{T}^{w_{m_1} w_{m_2}} = t_2^{w_{m_1} w_{m_2}}
$$

(6.17)
Writing explicitly only those terms of excitation order two or less, we then have

\[
\begin{align*}
(e^{-T} \hat{e}_{om} e^{\hat{T}})_x &= \hat{e}_{um} \\
(e^{-T} \hat{f}_{om} e^{\hat{T}})_x &= \sum_{w_m} t_{1}^{w_m} \hat{e}_{om} + \sum_{m', w_m} \sum_{w_m} t_{2}^{w_m, w_m} \hat{e}_{om} \\
(e^{-T} \hat{f}_{um} e^{\hat{T}})_x &= t_{1}^{v_m} \hat{e}_{um} + \sum_{m', w_m} t_{2}^{w_m, v_m} \hat{e}_{om} \\
(e^{-T} \hat{d}_{om} e^{\hat{T}})_x &= -t_{1}^{u_m} + \sum_{w_m} t_{1}^{u_m, w_m} \hat{e}_{om} - \sum_{m', w_m} t_{2}^{u_m, w_m} \hat{e}_{om} \\
&\quad + \sum_{w_m} m', w_{m'} \left( t_{1}^{u_m, t_{2}^{w_m, w_m}} + t_{1}^{u_m, t_{2}^{w_m, u_m}} \right) \hat{e}_{om} e_{om} + \cdots \\
(e^{-T} \hat{e}_{um1} \hat{e}_{om2} e^{\hat{T}})_x &= \hat{e}_{um1} \hat{e}_{om2} \\
(e^{-T} \hat{e}_{um1} \hat{f}_{om2} e^{\hat{T}})_x &= \sum_{w_m} t_{1}^{w_m} \hat{e}_{um1} \hat{e}_{om2} + \cdots \\
(e^{-T} \hat{e}_{um1} \hat{d}_{om2} e^{\hat{T}})_x &= -t_{1}^{u_m} \hat{e}_{um1} + \sum_{w_m} t_{1}^{u_m, w_m} \hat{e}_{om2} \\
&\quad - \sum_{m', w_{m'}} \left( t_{2}^{u_m, w_m} \hat{e}_{om1} \hat{e}_{om2} + \cdots \right) \\
(e^{-T} \hat{f}_{om1} \hat{e}_{om2} e^{\hat{T}})_x &= \sum_{w_{m1}, w_{m2}} \left( t_{1}^{w_{m1}, w_{m1}} + t_{2}^{w_{m1}, w_{m2}} \right) \hat{e}_{om1} \hat{e}_{om2} + \cdots \\
(e^{-T} \hat{f}_{om1} \hat{f}_{om2} e^{\hat{T}})_x &= \sum_{w_{m1}, w_{m2}} \left( t_{1}^{w_{m1}, w_{m1}} + t_{2}^{w_{m1}, w_{m2}} \right) \hat{e}_{om1} \hat{e}_{om2} + \cdots \\
(e^{-T} \hat{f}_{om1} \hat{d}_{om2} e^{\hat{T}})_x &= -\sum_{w_{m1}} \left( t_{1}^{w_{m1}, u_{m1}} + t_{2}^{w_{m1}, u_{m2}} \right) \hat{e}_{om1} \hat{e}_{om2} \\
&\quad + \sum_{w_{m1}, w_{m2}} \left( t_{1}^{w_{m1}, t_{2}^{u_{m1}, u_{m2}}} + t_{1}^{u_{m1}, t_{2}^{w_{m1}, w_{m2}}} \right) \hat{e}_{om2} + \cdots \\
&\quad + t_{2}^{u_{m1}, t_{2}^{w_{m1}, u_{m2}}} \hat{e}_{om1} \hat{e}_{om2} \\
&\quad - \sum_{m', w_{m'}} \sum_{w_{m1}, w_{m2}} \left( t_{2}^{w_{m1}, w_{m2}} + t_{1}^{u_{m1}, t_{2}^{w_{m1}, w_{m2}}} \right) \hat{e}_{om1} \hat{e}_{om2} + \cdots \\
\end{align*}
\]
\[
\left( e^{-\hat{T}} \hat{d}_{\omega m_1} \hat{d}_{\omega m_2} e^{\hat{T}} \right)_x = t_1^{\omega m_1} t_1^{\omega m_2} + t_2^{\omega m_1} t_2^{\omega m_2} \\
+ \sum_{m', \omega_{m'}} (t_2^{\omega_{m'} \omega m_1} t_1^{\omega m_2} + t_1^{\omega m_1} t_2^{\omega_{m'} \omega m_2}) \hat{e}_{\omega_{m'}}^{\omega m_1} \\
- \sum_{\omega m_1} (t_1^{\omega m_1} t_1^{\omega m_2} + t_1^{\omega m_1} t_2^{\omega m_1} + t_1^{\omega m_1} t_2^{\omega m_2}) \hat{e}_{\omega m_1}^{\omega m_1} \\
- \sum_{\omega m_2} (t_1^{\omega m_1} t_1^{\omega m_2} + t_1^{\omega m_2} t_2^{\omega m_1} + t_2^{\omega m_1} t_1^{\omega m_2}) \hat{e}_{\omega m_2}^{\omega m_2} \\
+ \sum_{m', \omega_{m'}} \sum_{m'', \omega_{m''}} t_2^{\omega_{m'} \omega m_1} t_2^{\omega_{m''} \omega m_2} \hat{e}_{\omega_{m'}}^{\omega m_1} \hat{e}_{\omega_{m''}}^{\omega m_2} \\
- \sum_{\omega m_1} \sum_{m', \omega_{m'}} (t_2^{\omega_{m'} \omega m_1} t_1^{\omega m_1} t_1^{\omega m_2} + t_2^{\omega_{m'} \omega m_1} t_1^{\omega m_2} t_1^{\omega m_1}) \\
+ t_1^{\omega m_1} t_2^{\omega_{m'} \omega m_1} t_1^{\omega m_2} \\
+ t_2^{\omega_{m'} \omega m_1} t_2^{\omega m_1} t_1^{\omega m_2} + t_2^{\omega_{m'} \omega m_1} t_2^{\omega m_1} t_2^{\omega m_2} \hat{e}_{\omega_{m'}}^{\omega m_1} \hat{e}_{\omega_{m''}}^{\omega m_2} \\
- \sum_{\omega m_2} \sum_{m', \omega_{m'}} (t_1^{\omega m_1} t_1^{\omega m_2} t_2^{\omega_{m'} \omega m_1} + t_1^{\omega m_1} t_1^{\omega m_2} t_2^{\omega_{m'} \omega m_2}) \\
+ t_1^{\omega m_1} t_2^{\omega_{m'} \omega m_1} t_1^{\omega m_2} \\
+ t_2^{\omega_{m'} \omega m_1} t_2^{\omega m_1} t_1^{\omega m_2} + t_2^{\omega_{m'} \omega m_1} t_2^{\omega m_1} t_2^{\omega m_2} \hat{e}_{\omega_{m'}}^{\omega m_2} \hat{e}_{\omega_{m''}}^{\omega m_2} \\
+ \sum_{\omega m_1} \sum_{\omega m_2} (t_1^{\omega m_1} t_1^{\omega m_2} t_1^{\omega m_2} t_1^{\omega m_2}) \\
+ t_1^{\omega m_1} t_2^{\omega m_1} t_1^{\omega m_2} t_1^{\omega m_2} \\
+ t_1^{\omega m_1} t_2^{\omega m_1} t_1^{\omega m_2} t_2^{\omega m_2} \\
+ t_1^{\omega m_1} t_2^{\omega m_1} t_1^{\omega m_2} t_1^{\omega m_2} + t_2^{\omega m_1} t_2^{\omega m_2} t_2^{\omega m_1} t_2^{\omega m_2} \hat{e}_{\omega m_1}^{\omega m_1} \hat{e}_{\omega m_2}^{\omega m_2} \\
+ \ldots \tag{6.18}
\]
APPENDIX B: GENERAL DISCUSSION OF AUTOMATIC CODE GENERATION

In order to solve the projected CCSD equations, from section 1.3 we understand that evaluating the BCH expansion, eq. (6.19), is the key step of optimizing coupled-cluster amplitudes. Here the same BCH expansion is given again,

\[
e^{-\hat{T}}\hat{H}e^\hat{T} = \hat{H} + [\hat{H}, \hat{T}] + \frac{1}{2}[[\hat{H}, \hat{T}], \hat{T}] + \frac{1}{6}[[[\hat{H}, \hat{T}], \hat{T}], \hat{T}] + \frac{1}{24}[[[[\hat{H}, \hat{T}], \hat{T}], \hat{T}], \hat{T}] \quad (6.19)
\]

The \( \hat{H} \) is previously defined as,

\[
\hat{H} = \sum_{pq} h_{pq} a_p^+ a_q + \frac{1}{4} \sum_{pqrs} V_{pqrs} a_p^+ a_q^+ a_r a_s \quad (6.20)
\]

The \( \hat{T} \) is truncated at double excitation level,

\[
\hat{T} = \sum_{ia} \hat{t}_i^a + \sum_{ijab} \hat{t}_{ij}^{ab} = \sum_{ia} t_i^a a_i^+ a_i + \sum_{ijab} t_{ij}^{ab} a_j^+ a_j a_i^+ a_i \quad (6.21)
\]

and we have additional creation and annihilation operators from singly and doubly excited states,

\[
|\Psi_i^{a'}\rangle = a_i^{a'} a_i^+ |\Psi_0\rangle \quad (6.22)
\]

\[
|\Psi_i^{a'b'}\rangle = a_j^{a'} a_j^+ a_i^{b'} a_i^+ |\Psi_0\rangle \quad (6.23)
\]

where the \( \Psi_0 \) is the reference state (in this case the Hartree-Fock ground state) and the \( \Psi_i^{a'} \) denotes a single excited state and the \( \Psi_i^{a'b'} \) represents a double excited state.

One can compute the projected CC equation, the eq. (1.42), for single and double excitations as the following,

\[
\langle \Psi_i^{a'} | e^{-\hat{T}} \hat{H} e^\hat{T} | \Psi_0 \rangle =
\]
\langle \Psi_0 | a_{i'}^+ a_{\alpha'} \left\{ \hat{H} + [\hat{H}, \hat{T}] + \frac{1}{2} [[\hat{H}, \hat{T}], \hat{T}] + \frac{1}{6} [[[\hat{H}, \hat{T}], \hat{T}], \hat{T}] + \frac{1}{24} [[[[[\hat{H}, \hat{T}], \hat{T}], \hat{T}], \hat{T}] \right\} | \Psi_0 \rangle \\
(6.24)

\langle \Psi_{i'j'}^{a'b'} | e^{-\hat{T} \hat{H} e^{\hat{T}}} | \Psi_0 \rangle = \\
\langle \Psi_0 | a_{i'}^+ a_{\alpha'} a_{j'}^+ a_{\beta'} \left\{ \hat{H} + [\hat{H}, \hat{T}] + \frac{1}{2} [[\hat{H}, \hat{T}], \hat{T}] + \frac{1}{6} [[[\hat{H}, \hat{T}], \hat{T}], \hat{T}] + \frac{1}{24} [[[[[\hat{H}, \hat{T}], \hat{T}], \hat{T}], \hat{T}] \right\} | \Psi_0 \rangle \\
(6.25)

and if we plug in the definition of \( \hat{H} \), eq. (6.20), and \( \hat{T} \), eq. (6.21), into eq. (6.24) and eq. (6.25), quickly we obtain a huge number of terms which is beyond human's ability to keep track of on paper. This gives the fundamental reason of implementing automatic code generation. Using the commutator relations defined in the CC overview section, certain computer algorithms can be developed to analyze each term and to simplify them by swapping creation and annihilation operators and looking for zero terms (that is, creation on occupied and annihilation on unoccupied orbitals). The zero terms will be deleted from the huge list of equations and the surviving equations will be translated from python object representations into actual C code automatically. Certain shell script is responsible for compiling those C files into shared libraries that are ready to be called from python ctypes.cdll.LoadLibrary class objects. A python wrapper is also generated on the fly for combining and coordinating all the python calls to these thousands of C functions residing in the shared libraries.

The most difficult part in automatic code generation involves the pattern recognitions of correct commutator relation to use and discovering zero terms. Substantial amount of work has been done to have these autocode equations derived and debugged. It all starts with the generation of all the terms in eq. (6.24) and eq. (6.25) since we are deriving the equations for excitonic CC theory up to double excitations. In the autocode implementation, there are three classes designed for the representation of commutator terms. They are the base class, the summation class,
and the multiplication class. The base class represent any single most fundamental term in an equation. The multiplication class is a representation of a string of terms, objects of the base class. The summations are also common when two or more equations have additive relation. A mixture of the above objects can represent all the equations encountered in CC theory. On top of the formalism, these classes also have pre-defined rules for pattern recognitions and the capability to rearrange operators to simplify themselves. They are implemented as member functions that are called every time a term is generated. Recursion is the ideal technique to use when multiple layers of objects are present. Therefore all the generated equations are recursively iterated over to allow simplifications to be thoroughly carried out.

The key idea for running such simplifications on equation terms is to move creation-on-occupied and annihilation-on-unoccupied operators to the right side of the equation, where the HF ground state locates. A creation of occupied orbitals or an annihilation on unoccupied orbitals will result in zero term while acting on the HF ground state. When the simplification member function is invoked for certain term, it will look through the whole sequence of operators and start moving these operators using the commutator relations. Swapping the same type of operators will result in a coefficient of $-1$. However, swapping creation and annihilation operators is more complicated as it will generate two terms in a subtraction form. The first term is typically a classical delta while the other one is the term of swapped operators in opposite phase. By doing this procedure over and over, one simple equation can populate into a long list of equations, which mostly are zeros and will be checked again for possibility of further simplifications. The typical timing on a Sandy bridge-EP Xeon E5 2.2 GHz CPU core is about 40 minutes for the longest derivation. The resulted list of equations in terms of those three classes are saved into binary data file using python pickle module. The advantage of using pickled
format files is that a direct copy of the objects in memory is saved to the disk. The loading time of this huge list is almost instant. There are more than two thousand individual equations that survived the above derivations and were saved to pickle files. It’s worth noting that, the \_\_str\_\_ method is overwritten by a function that generates Latex format outputs, which made these equations easily visualized and simpler to be debugged.

The other challenge we faced was how to convert the python object representation of equations into compilable C functions. This automation required several months of work. The reason that C programming language was chosen is that the python programming language is an interpreted language, which will, in most cases, execute substantially slower than most compiled languages, such as C/C++ and Java. Therefore we designed our automatic code generator to read pickled equation objects and produce C code for the computationally intense steps. This is implemented by detecting the top-level object type. If it is a summation object, a python list will be used to keep track of each summand in this case. If the top-level object is a multiplication object, then only one function will be generated in C language. The base class object will not appear at the top level. In the case of a single scalar left in the equation, a trivial multiplication would be used to wrap the single term for the sake of consistency.

When dealing with a single multiplication term, the automatic code generator will first read the summation loop information in terms of the indices and limits. This information is converted to for loops accordingly. In the case of conflicting summation limits, helper algorithms were programmed to resolve the disputes. As the code generator passes the looping information part of the term, it reaches the matrix element terms. Things start getting complicated here. Different equations require different fluctuation blocks’ information. This is also linked with the input arguments for C functions. Therefore, different blocks needed in the equations will
result in different input arguments for the C function definitions. All these must be handled properly at the python caller level as we are passing down python data objects into C function using python ctypes module.

When all the matrix elements were appended to the multiplication form in the C function, the product of such numbers is added to the result matrices. Then we can close the curly brackets for the for loops. The "program by side effects" scheme was used in our designs of these C functions. It is to pass in an empty array of the same size to hold the results instead of returning a pointer to a newly allocated array that generated inside the C function. There are pros and cons for different programming schemes but this method was chosen for its simplicity since C-style arrays are contiguous in computer memory. The indexing scheme for contiguous blocks of two-dimensional array (matrix) is very straightforward. At the end of every C file, there is a wrapper function that is invoking all the equations above. This acts like an interface to the python caller so that every shared library is called only once.

Parallelism was used to speed up the executions due to the computations of more than two thousand equations every time the BCH expansion is computed. The OpenMP library was used in our implementation. The details of how OpenMP pragma works with regular C code are discussed in the appendices. It is worth noting that the race condition could occur if multiple write requests are sent to the same memory location. The atomic or critical pragma is used to ensure the same memory location is only accessible for only one thread at a time. OpenMP library is designed for single computing node with many available threads, which meets our need well since the excitonic CC code is more resources-hungry than power-hungry.

The C header files are also composed for these C functions. The first reason is traditional. The C function names should be declared before being invoked in the
C code. Specially for a large project, these declarations should be moved to header files according to the principle of separation of complexity. Secondly, a python caller needs to be composed at the end of autocode generation. Having all function definitions in separated header files largely reduced the work loads than those if all function definitions were to be read from C source files. The use of header guards is always necessary for complicated projects. It is implemented in our header files as well using the `#ifndef ... #define ... #endif` syntax. By doing so the C preprocessor will not throw an error if the same header file is included in more than one location since they are already guarded from multiple include.

These C source files and header files will be compiled and built as shared libraries with `.so` extension in Linux operating systems. Now we need to generate the python caller. This is done by scanning through all header files for C function definition. The data types and variable names are analyzed to be composed into corresponding python-to-ctypes conversions, such as python integer to C `int` or C `long long` and python `float` to C `double`. After laying down all the type conversions, library handlers using `ctypes. cdll.LoadLibrary` objects are instantiated and used to load the C shared libraries we just built and pass all necessary values and matrices to the C code. Empty copies of matrices were made at the beginning and used to hold computation results.

The automatically generated code is being invoked each time the BCH expansion is calculated. When the calculations terminated, we are able to collect the results in `numpy` array format simply from the python side, which allow researchers to focus on method development rather than to get bogged down by programming details and debugging. Thus the whole excitonic CC code is written in python except for the autocode.
APPENDIX C: DIRECT INVERSION OF ITERATIVE SUBSPACE ACCELERATION

The direct inversion of iterative subspace (DIIS) method can be applied to many iterative problems such as the self-consistent field procedure. Here we introduce the DIIS procedure for CC iterations. It is essentially choosing a subspace of latest several, e.g., five, iterations and calculating and assigning weights for each of the iterations, then forming a much more robust weighted change of amplitude \( \Delta t^{\text{avg}} \) for the upcoming iteration to speed up the convergence. That is

\[
\Delta t^{\text{avg}} = \sum_{k=1}^{n} w_k \Delta t^{(k-1)}
\]

(6.26)

and these weight must sum to unity

\[
\sum_{k=1}^{n} w_k = 1
\]

(6.27)

Therefore the new \( t^{(n)} \) at the upcoming iteration \( n \) is interpolated by

\[
t^{(n)} = \sum_{k=1}^{n-1} w_k t^{(k)} + w_n \left( t^{(n-1)} + \Delta t^{(n-1)} \right)
\]

(6.28)

Here the subspace includes all previous iterations in which the early iterations can pose toxic effects on the convergence because the converged amplitudes typically make non-trivial changes compared to the starting set. Thus adding weighted amplitudes from early iterations can slow down the convergence. Usually a small subspace size accelerates CC method very well.

Weights are determined by solving this set of linear equations of dimension \( N_{\text{subspace}} + 1 \) in eq. (6.29) where the equation of obtaining \( \Omega^{(0)} \) can be found in the
CC introduction section. The weights of $w_1$ through $w_N$ can be plugged into the equations above to calculate the $\Delta^\text{avg}_t$.

\[
\begin{pmatrix}
\Omega_1^{(0)} \cdot \Omega_1^{(0)} & \cdots & \Omega_1^{(0)} \cdot \Omega_N^{(0)} & 1 \\
\vdots & \ddots & \vdots & \vdots \\
\Omega_N^{(0)} \cdot \Omega_1^{(0)} & \cdots & \Omega_N^{(0)} \cdot \Omega_N^{(0)} & 1 \\
1 & \cdots & 1 & 0
\end{pmatrix}
\begin{pmatrix}
w_1 \\
\vdots \\
w_N \\
\lambda
\end{pmatrix}
= 
\begin{pmatrix}
0 \\
\vdots \\
0 \\
1
\end{pmatrix}
\]

(6.29)
APPENDIX D: PYTHON AND C INTERFACE

There are two ways of passing information between python and C programs. The simplest approach is to build dynamic linking libraries written in C and load such shared library from python ctypes module. The other way to accomplish this requires the knowledge of coding up a Cython module in C using python application programming interface (API). Both will be demonstrated here but the first way is preferred due to its simplicity.

A simple C code example, add.c, is provided for a trivial function of adding two double precision numbers.

```c
double addtwo(double x, double y)
{
    return x + y;
}
```

A shared C Library can be compiled then built with gcc compiler using the following commands,

```
gcc -c -fPIC add.c -o add.o

gcc -shared add.o -o libadd.so
```

The "-c" flag means "compile only and no linking" and the "-fPIC" indicates this C code is position independent, which is essential for building shared libraries. "-o" is the flag for feeding in the output file name. The first step is for compiling the C code into an object file with no linking (shown in the first line). Here the linking is the process that the linker program tells the user programs where to find the actual library files if the user’s program is calling the functions from standard or
installed libraries. We say your executable or library is linked against certain li-

brary files. The second step is the command that does the linking for the new li-

brary, libadd.so, using the `-shared` flag. The `.so` here means "shared objects".

The linking dependencies can be shown with the following command `ldd`.

```
ldd libadd.so
    linux-vdso.so.1 => (0x00007fff53da2000 )
    libc.so.6 => /lib64/libc.so.6 (0x00007f3023e43000 )
    /lib64/ld-linux-x86-64.so.2 (0x00007f30243d9000 )
```

These system libraries are linked against in this case, and notably the second li-

brary is the GNU standard C library file, libc.so.

Calling a C library function from python is quite simple. The following code

is showing an example of adding two python float types using a C shared library

function call.

```
import ctypes
x = 100.0
y = 200.0
x_as_ctype = ctypes.c_double(x)
y_as_ctype = ctypes.c_double(y)
libraryHandler = ctypes.cdll.LoadLibrary('./libadd.so')
z = libraryHandler.addtwo(x_as_ctype, y_as_ctype)
print('x + y =', z)
```

This piece of python code is first initializing two floating point numbers x and y

with the values of 100.0 and 200.0, respectively. Then they are converted to their

c_type counterpart the `c_double` (C double data type represented in python) ob-

jects. Now the library handler is created using the `ctypes.cdll.LoadLibrary`

class. This python handler to the libadd.so library is containing all available
functions from it. By calling the `addtwo` function like a method call from this `libraryHandler` object with a dot syntax, one is able to use a C function directly from python. The printout of this piece of code should be:

\[
x + y = 300.0
\]

Now we discuss the second way of communicating between python and C programs, that is to write a python module (Cython module to be exact). This is much more complicated as it involves properly passing list of python objects to the C functions but the python API functions are quite robust and flexible, a quick implementation is still possible for not too complicated functionality. Here we are showing one example of python module function, source file name as `transformVMat.c`, that transforms electronic repulsion matrix with the unitary matrix obtained from the diagonalization of the Fock matrix.

```c
PyObject* transformV(PyObject *self, PyObject *args)
{
    PyObject* Py_Mat;
    PyObject* Py_U;

    if (!PyArg_ParseTuple(args, "OO", &Py_Mat, &Py_U))
    {
        puts("PyArg_ParseTuple Function Failed.");
        return NULL;
    }

    if ((Py_Mat == NULL || Py_U == NULL))
    {
        puts("Failed to Parse the Input Matrices.");
    }
```
return NULL;

PyObject* Np_Mat = PyArray_FROM_OTF(Py_Mat, NPY_DOUBLE, NPY_ARRAY_DEFAULT);
PyObject* Np_U = PyArray_FROM_OTF(Py_U, NPY_DOUBLE, NPY_ARRAY_DEFAULT);

long long mat_dim = (long long)PyArray_DIM(Py_U, 0);

if (Np_Mat == NULL || Np_U == NULL)
{

    // decrease reference counts for these...
    PyArray_XDECREF((PyArrayObject*)Np_Mat);
    PyArray_XDECREF((PyArrayObject*)Np_U);
    puts("Failed to Convert to Numpy Array Types.");
    return NULL;
}

// Now Get the raw data in C arrays.

double* Mat = (double*)PyArray_DATA(Np_Mat);
double* U = (double*)PyArray_DATA(Np_U);

if (Mat == NULL || U == NULL)
{

    PyArray_XDECREF((PyArrayObject*)Np_Mat);
    PyArray_XDECREF((PyArrayObject*)Np_U);
    puts("Retrieving Mat or U Data Failed.");
}
return NULL;
}

// Calling real transformation function
double* Ret_Mat = transform_V(Mat, U, mat_dim);

npy_intp Ret_dims[2] = {mat_dim*mat_dim, mat_dim*mat_dim};
PyObject* Np_Ret_Mat = PyArray_SimpleNewFromData(2, Ret_dims, NPY_DOUBLE, Ret_Mat);

// Cleaning Up (decrease reference count)
PyArray_XDECREF((PyArrayObject*) Np_Mat);
PyArray_XDECREF((PyArrayObject*) Np_U);

// Build a Numpy Matrix with Ret_Mat
return Np_Ret_Mat;
}

// Module's Method Table
static PyMethodDef TransformVMethods[] = {
    {"transformV", transformV, METH_VARARGS, "Transform the electronic repulsion matrix."},
    {NULL, NULL, 0, NULL} /* Sentinel */
};
The function name is called `transformV` and it takes two PyObject pointers, one is `self` and the other is `args` (argument list). First we initialize two PyObject pointers for receiving the repulsion matrix python object and the unitary matrix python object from the `args`. Then we call the `PyArg_ParseTuple` API function to parse the `args` into two PyObjects denoted by "OO" and point `Py_Mat` and `Py_U` to the real objects, respectively. Checking for NULL pointer is always desired in case of failure and this makes the code safe to execute. Now we use the `PyArray_FROM_OTF` to convert the `Py_Mat` and the `Py_U` into numpy objects since numpy array was the original data type being passed in as the arguments. Here we finished receiving PyObjects and restoring them back to original numpy array format. Numpy array are stored contiguously in C style in memory. We check again
for NULL pointers and would decrease the reference count if this step fails. (This allow garbage collector to work properly in failure cases.) The retrieval of the raw double precision data array is quite straightforward. By using `PyArray_DATA` API, we can get a double type pointer `Mat` directly pointing to the raw data array of `Np_Mat` so is the `Np_U`. We check again for NULL pointers just in case. The real transformation is done by low level help function `transform_V` and the result data array has a pointer named `Ret_Mat`. The following lines are to construct a numpy 2D array from the raw data pointed to by `Ret_Mat`. Before exiting, we need to decrease the reference count again as calling the `PyArray_XDECREF` API. Finally the function return a numpy object which is also a PyObject.

The rest of the code is setting up some module method table with function name, physical C function name, condition of whether take arguments and short description in the `PyMethodDef` array. Then the array is passed to the `PyModuleDef` structure. Finally the `PyMODINIT_FUNC` is called to inform the compiler that the module initializes here with the template-like name `PyInt_` plus user-defined module name, `transformVMat` in this case. The `import_array()` function is importing the numpy API functions. The `PyInt_transformVMat` returns a the module object.

Here is an sample python `setup.py` script for building python modules.

```python
from distutils.core import setup, Extension
import numpy.distutils.misc_util

setup(
    name = 'transformVMat',
    version = '1.0',
    ext_modules=[ Extension("transformVMat", sources=["transformVMat.c"], extra_compile_args=['-std=c99']) ],
```
include_dirs=numpy.distutils.misc_util.
get_numpy_include_dirs()}

The distutils is the distribution utility module that helps one to build a python extension module without the need to install it into the site-package library folder. All the detailed information is passed to the setup constructor, such as module name, version number, module source file, compilation flags, and proper way of import numpy APIs.

And this setup.py can be ran by

```bash
gnome3setup.py build_ext --inplace
```

Now we can call upon this transformVMat.transformV function directly in python without any installation like the following.

```python
import transformVMat
transformed_mat = transformVMat.transformV( elec_repul_mat, U_mat )
```

The most compiling reason to pass information between the two languages is that the interpreted language like python is lacking of speed when the code is solely doing number crunching but the C code can run the calculations extremely fast. The reasons are many. C language is a strongly typed and compiled language while python relies on its interpreter reading in line by line during executions. Compiled languages are in general much faster than scripting languages due to the direct executions of the binary machine code. But the strongly typed languages tend to have complex syntax and long development periods which is not favored by today’s standards. Quick prototyping is very important nowadays for both industrial engineers and scientific researchers. The interpreted languages like python has very
flexible and user-friendly syntax. This allows quick developments of complicated algorithms though at a compromised performance which can be alleviated later by an increase of hardware resources or by an integration with compiled languages like C for speed-limiting steps.
APPENDIX E: PARALLEL COMPUTING WITH PYTHON AND OPENMP

One major difference between multi-threading and multi-processing is whether you have shared resources between individual workers. Threading technique uses the shared resources that are available to all threads but at a risk of race conditions, that is, multiple workers trying to write to the same resource at the same time. The race condition can cause the shared resources to go to unpredictable states and ultimately crash a program or return useless results. A lock is needed to handle this situation but is discouraged due to the growing complexity over time. In contrast, the multi-processing is a much cleaner scheme that separate work loads into non-communicating processes. The implementation is much easier and when certain process is not needed, it can be simply killed to release computer hardware. The threading library in python does not allow a CPU usage of more than one physical core due to the presence of the "global interpreter lock". The threading library is mostly designed for quick responsiveness of the program if multiple threads are available to address the current needs. Therefore, the multiprocessing module is used in our work to deliver the computing power of multi-core CPUs.

The following python code is a simple example of using 10 processes to double every number in the original array.

```python
from multiprocessing import Pool

def double_func( a_number ):
    return 2 * a_number
```
an_array = [ i for i in range(100) ]

with Pool(10) as myPool:
    doubled_array = myPool.map(double_func, an_array)
print(doubled_array)

Here we use the work Pool rather than create processes one by one. For simple computations, the Pool class is much simpler to use. A double_func is defined to multiply every input argument by 2 and having the result returned. an_array is the original number array which contains the all the integer number ranging from 0 to 99. The with ... as ... syntax is automatically garbage-collected thus it is favored here. A pool of 10 worker (processes) is assigned to myPool to wait for computations. By using the map member function of Pool class, the whole work load of 100 numbers are passed to 10 processes with a target of applying such double_func to each number. The results are saved in another list called doubled_array. Now we have already demonstrated how easily the multiprocessing can parallelize your existing code with only several lines of code.

Python multiprocessing module indeed largely increases the performance of the code in our work. This simple "a pool of workers" method can be readily extended to any complex data structures or algorithms, which makes it very ideal for speeding up some not-too-simple number crunching steps in python.

On the other hand, we have the OpenMP library now being a standard part of the GNU C/C++ library. The OpenMP stands for open source message passing library which is a thread-based parallel computing implementation mainly designed for C/C++ and Fortran. The OpenMP library provides us a set of simple pragma clauses to easily modify your C/C++ code into a multi-threading powerful code without the need for detail threading knowledge. The following code is a simple
example for parallelizing a "for " loop in C with user-defined number of threads to work with.

```c
#include <omp.h>

int main()
{
    // set the number of threads to 10 in this case
    omp_set_dynamic(0);
    omp_set_num_threads(10);

    double array[100], doubled_array[100] = {0.0}, {0.0};
    for (int i=0; i<100; i++)
    {
        array[i] = (double) i;
    }

    #pragma omp parallel for
    for (int i=0; i<100; i++)
    {
        #pragma omp atomic
        doubled_array[i] += 2.0 * array[i];
    }

    return 0;
}
```

The "omp.h" header file should be included as shown in the first line. In the main function, the `omp_set_dynamic` function removes the predefined environment vari-
able \texttt{OMP\_NUM\_THREADS} in order to take control here. The \texttt{omp\_set\_num\_threads} function is telling the program to use 10 CPU threads in later omp computations. Then we make two double precision arrays named \texttt{array} and \texttt{doubled\_array} as the original and result array, respectively. Now we fill up the \texttt{array} with numbers ranging from 0 to 99.0. Then we issue the OpenMP pragma for parallel for loops and also include a atomic operation pragma for locking \texttt{doubled\_array} while one thread is writing to it. This can also be done with the \texttt{critical} clause but the latter is more computationally intense. The lower level \texttt{atomic} is fully capable in this case. Then numbers are being doubled and saved to the new array.

The OpenMP will implement all the complicated internals for you as a matter of fact the OpenMP is using the GNU pthread library in the backend. The threads are still being generated one by one but automatically. OpenMP works like a robust facade of the pthread library. With the aid of OpenMP library, one can easily speed up the limiting steps of your code with little programming efforts.