In Pursuit of Local Correlation for Reduced-Scaling Electronic Structure Methods in Molecules and Periodic Solids

Marjory Carolena Clement

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Chemistry

Edward F. Valeev, Chair

Daniel Crawford

Mark P. Embree

Nicholas Mayhall

July 14, 2021

Blacksburg, Virginia

Keywords: Electronic Structure, Local Correlation, Periodic Solids, Reduced-Scaling, Orbital Localization

Copyright 2021, Marjory Carolena Clement
In Pursuit of Local Correlation for Reduced-Scaling Electronic Structure Methods in Molecules and Periodic Solids

Marjory Carolena Clement

(ABSTRACT)

Over the course of the last century, electronic structure theory (or, alternatively, computational quantum chemistry) has grown from being a fledgling field to being a “full partner with experiment” [Goddard Science 1985, 227 (4689), 917–923]. Numerous instances of theory matching experiment to very high accuracy abound, with one excellent example being the high-accuracy \textit{ab initio} thermochemical data laid out in the 2004 work of Tajti and co-workers [Tajti et al. J. Chem. Phys. 2004, 121, 11599] and another being the heats of formation and molecular structures computed by Feller and co-workers in 2008 [Feller et al. J. Chem. Phys. 2008, 129, 204105]. But as the authors of both studies point out, this very high accuracy comes at a very high cost. In fact, at this point in time, electronic structure theory does not suffer from an accuracy problem (as it did in its early days) but a cost problem; or, perhaps more precisely, it suffers from an accuracy-to-cost ratio problem. We can compute electronic energies to nearly any precision we like, \textit{as long as we are willing to pay the associated cost.}

And just what are these high computational costs? For the purposes of this work, we are primarily concerned with the way in which the computational cost of a given method scales with the system size; for notational purposes, we will often introduce a parameter, $N$, that
is proportional to the system size. In the case of Hartree-Fock, a one-body wavefunction-based method, the scaling is formally $N^4$, and post-Hartree-Fock methods fare even worse. The coupled cluster singles, doubles, and perturbative triples method [CCSD(T)], which is frequently referred to as the “gold standard” of quantum chemistry, has an $N^7$ scaling, making it inapplicable to many systems of real-world import.

If highly accurate correlated wavefunction methods are to be applied to larger systems of interest, it is crucial that we reduce their computational scaling. One very successful means of doing this relies on the fact that electron correlation is fundamentally a local phenomenon, and the recognition of this fact has led to the development of numerous local implementations of conventional many-body methods. One such method, the DLPNO-CCSD(T) method, was successfully used to calculate the energy of the protein crambin [Riplinger, et al. *J. Chem. Phys.* 2013, 139, 134101].

In the following work, we discuss how the local nature of electron correlation can be exploited, both in terms of the occupied orbitals and the unoccupied (or virtual) orbitals. In the case of the former, we highlight some of the historical developments in orbital localization before applying orbital localization robustly to infinite periodic crystalline systems [Clement, et al. 2021, *Submitted to J. Chem. Theory Comput.*]. In the case of the latter, we discuss a number of different ways in which the virtual space can be compressed before presenting our pioneering work in the area of iteratively-optimized pair natural orbitals (“iPNOs”) [Clement, et al. *J. Chem. Theory Comput.* 2018, 14 (9), 4581–4589].

Concerning the iPNOs, we were able to recover significant accuracy with respect to tradi-
tional PNOs (which are unchanged throughout the course of a correlated calculation) at a comparable truncation level, indicating that our improved PNOs are, in fact, an improved representation of the coupled cluster doubles amplitudes. For example, when studying the percent errors in the absolute correlation energies of a representative sample of weakly bound dimers chosen from the S66 test suite [Rezác, et al. J. Chem. Theory Comput. 2011, 7 (8), 2427–2438], we found that our iPNO-CCSD scheme outperformed the standard PNO-CCSD scheme at every truncation threshold ($\tau_{\text{PNO}}$) studied. Both PNO-based methods were compared to the canonical CCSD method, with the iPNO-CCSD method being, on average, 1.9 times better than the PNO-CCSD method at $\tau_{\text{PNO}} = 10^{-7}$ and more than an order of magnitude better for $\tau_{\text{PNO}} < 10^{-10}$ [Clement, et al. J. Chem. Theory Comput. 2018, 14 (9), 4581–4589]. When our improved PNOs are combined with the PNO-incompleteness correction proposed by Neese and co-workers [Neese, et al. J. Chem. Phys. 2009, 130, 114108; Neese, et al. J. Chem. Phys. 2009, 131, 064103], the results are truly astounding. For a truncation threshold of $\tau_{\text{PNO}} = 10^{-6}$, the mean average absolute error in binding energy for all 66 dimers from the S66 test set was 3 times smaller when the incompleteness-corrected iPNO-CCSD method was used relative to the incompleteness-corrected PNO-CCSD method [Clement, et al. J. Chem. Theory Comput. 2018, 14 (9), 4581–4589].

In the latter half of this work, we present our implementation of a limited-memory Broyden-Fletcher-Goldfarb-Shanno (BFGS) based Pipek-Mezey Wannier function (PMWF) solver [Clement, et al. 2021, Submitted to J. Chem. Theory Comput.]. Although orbital localization in the context of the linear combination of atomic orbitals (LCAO) representation
of periodic crystalline solids is not new [Marzari, et al. Rev. Mod. Phys. 2012, 84 (4), 1419–1475; Jónsson, et al. J. Chem. Theory Comput. 2017, 13 (2), 460–474], to our knowledge, this is the first implementation to be based on a BFGS solver. In addition, we are pleased to report that our novel BFGS-based solver is extremely robust in terms of the initial guess and the size of the history employed, with the final results and the time to solution, as measured in number of iterations required, being essentially independent of these initial choices. Furthermore, our BFGS-based solver converges much more quickly and consistently than either a steepest ascent (SA) or a non-linear conjugate gradient (CG) based solver, with this fact demonstrated for a number of 1-, 2-, and 3-dimensional systems. Armed with our real, localized Wannier functions, we are now in a position to pursue the application of local implementations of correlated many-body methods to the arena of periodic crystalline solids; a first step toward this goal will, most likely, be the study of PNOs, both conventional and iteratively-optimized, in this context.

This work was supported in part by the U.S. National Science Foundation and the Virginia Tech Institute for Critical Technology and Applied Science
Increasingly, the study of chemistry is moving from the traditional wet lab to the realm of computers. The physical laws that govern the behavior of chemical systems, along with the corresponding mathematical expressions, have long been known. Rapid growth in computational technology has made solving these equations, at least in an approximate manner, relatively easy for a large number of molecular and solid systems. That the equations must be solved approximately is an unfortunate fact of life, stemming from the mathematical structure of the equations themselves, and much effort has been poured into developing better and better approximations, each trying to balance an acceptable level of accuracy loss with a realistic level of computational cost and complexity.

But though there has been much progress in developing approximate computational chemistry methods, there is still great work to be done. Many chemical systems of real-world import (particularly biomolecules and potential pharmaceuticals) are simply too large to be treated with any methods that consistently deliver acceptable accuracy.

As an example of the difficulties that come with trying to apply accurate computational methods to systems of interest, consider the seminal 2013 work of Riplinger and co-workers [Riplinger, et al. J. Chem. Phys. 2013, 139, 134101]. In this paper, they present the results
of a calculation performed on the protein crambin. The method used was DLPNO-CCSD(T), an approximation to the “gold standard” computational method CCSD(T). The acronym DLPNO-CCSD(T) stands for “domain-based local pair natural orbital coupled cluster with singles, doubles, and perturbative triples.” In essence, this method exploits the fact that electron-electron interactions (“electron correlation”) are a short-range phenomenon in order to represent the system in a mathematically more compact way. This focus on the locality of electron correlation is a crucial piece in the effort to bring down computational cost.

When talking about computational cost, we will often talk about how the cost scales with the approximate system size $N$. In the case of CCSD(T), the cost scales as $N^7$. To see what this means, consider two chemical systems $A$ and $B$. If system $B$ is twice as large as system $A$, then the same calculation run on both systems will take $2^7 = 128$ times longer on system $B$ than on system $A$. The DLPNO-CCSD(T) method, on the other hand, scales linearly with the system size, provided the system is sufficiently large (we say that it is “asymptotically linearly scaling”), and so, for our example systems $A$ and $B$, the calculation run on system $B$ should only take twice as long as the calculation run on system $A$.

But despite the favorable scaling afforded by the DLPNO-CCSD(T) method, the time to solution is still prohibitive. In the case of crambin, a relatively small protein with 644 atoms, the calculation took a little over 30 days. Clearly, such timescales are unworkable for the field of biochemical research, where the focus is often on the interactions between multiple proteins or other large biomolecules and where many more data points are required.

In the work that follows, we discuss in more detail the genesis of the high costs that are
associated with highly accurate computational methods, as well as some of the approximation techniques that have already been employed, with an emphasis on local correlation techniques. We then build off this foundation to discuss our own work and how we have extended such approximation techniques in an attempt to further increase the possible accuracy to cost ratio. In particular, we discuss how iteratively-optimized pair natural orbitals (the PNOs of the DLPNO-CCSD(T) method) can provide a more accurate but also more compact mathematical representation of the system relative to static PNOs [Clement, et al. *J. Chem. Theory Comput.* 2018, 14 (9), 4581–4589]. Additionally, we turn our attention to the problem of periodic infinite crystalline systems, a class of materials less commonly studied in the field of computational chemistry, and discuss how the local correlation techniques that have already been applied with great success to molecular systems can potentially be applied in this domain as well [Clement, et al. 2021, *Submitted to J. Chem. Theory Comput.*].
To my husband, Sam, and our soon to be born child
Acknowledgments

I would like to thank the following people for their help and support throughout my time in graduate school:

• My advisor, Prof. Edward F. Valeev, for patiently teaching me nearly everything I know about electronic structure theory and for continuing to believe in me

• My committee members, Prof. T. Daniel Crawford, Prof. Mark Embree, and Prof. Nick Mayhall, for their helpful discussions and willingness to guide me throughout my time in graduate school

• The chemistry graduate program director, Prof. John Morris, and the chemistry graduate program coordinator, Joli Huynh, for keeping me on track to graduate and making sure I always had the right paperwork

• My friends and colleagues in the chemistry department, especially the Valeev group members, for their helpful discussions and their willingness to share what they had learned before. Special thanks to Fabijan Pavošević for helping me get started in the
area of pair natural orbitals and to Xiao Wang for answering my every question about the electronic structure theory of periodic solids

- My family, especially my parents, Will and Angie Clement, for their love and encouragement and their constant reminders to keep the big picture in view

- My husband, Sam Slattery, for his unconditional love and support
Attribution

Chapter 2 was co-authored by Jinmei Zhang, Cannada A. Lewis, Chao Yang, and Edward F. Valeev. Zhang and Yang provided pioneering work and useful discussions, while Lewis provided some foundational code and useful discussions. Valeev was the primary source of the research idea and served to guide the writing of the manuscript. I did much of the coding, generated nearly all of the data, and contributed to writing and editing.

Chapter 3 was co-authored by Xiao Wang and Edward F. Valeev. Wang provided foundational work and helpful discussions, while Valeev again served as the primary source of the research idea and an editor and guide in the writing process. I did much of the coding, generated all of the data, and contributed to writing and editing.
# Contents

1 Introduction ................................. 1

1.1 The Electronic Structure Problem .......... 1

1.2 Terminology and Notation .................. 3

1.3 One-Body Methods .......................... 5

1.3.1 Hartree-Fock ............................ 5

1.4 Many-Body Methods ....................... 8

1.4.1 Perturbation Theory ...................... 8

1.4.2 Configuration Interaction ............... 10

1.4.3 Coupled Cluster ......................... 12

1.5 Approximation Techniques .................. 15

1.5.1 Local Correlation ....................... 17
1.5.2 Density Fitting .............................................. 26

1.6 Extension to Periodic Solids ................................. 30

1.6.1 Terminology and Notation .............................. 31

1.6.2 Periodic Hartree-Fock ................................. 35

2 Optimized Pair Natural Orbitals for the Coupled Cluster Methods 39

2.1 Abstract ....................................................... 39

2.2 Introduction .................................................. 40

2.3 Theoretical background .................................... 43

2.4 Computational details ....................................... 49

2.5 Results and discussion ..................................... 51

2.5.1 PNO-CCSD vs. iPNO-CCSD .......................... 51

2.5.2 Perturbative Energy Correction for PNO Incompleteness‡ ................................. 52

2.5.3 Broader Assessment ...................................... 58

2.6 Summary and Perspective ................................... 62

3 Robust Pipek-Mezey Orbital Localization in Periodic Solids 66

3.1 Abstract ....................................................... 66

‡This is a typo in the published work. Clearly, the word should be “incompleteness.”
3.2 Introduction ............................................................. 67
3.3 Formalism ............................................................... 70
  3.3.1 Basic Definitions ................................................. 70
  3.3.2 PM Functional with Pseudoinverse Minimal Basis Atomic Charges . . 71
  3.3.3 PM Functional Maximization .................................... 76
3.4 Computational Details ................................................... 86
3.5 Results ................................................................. 88
3.6 Summary and Perspective ............................................... 96

4 Conclusions and Future Outlook ........................................ 98

Appendix A: Supporting Information for “Robust Pipek-Mezey Orbital Localization in Periodic Solids” ............................................. 101
  A.1 PM Functional Values Obtained with Different Solvers ...................... 101
  A.2 Number of Iterations to Solution vs. BFGS Parameters ....................... 104
  A.3 Initial Guess .......................................................... 106
  A.4 Minimal Basis Set Choice ............................................. 108

Appendix B: Publication List .................................................. 111
List of Figures

1.1 Estimated time to solution for systems consisting of increasing numbers of uracil bases. The blue line corresponds to a conventional one-body method (such as Hartree-Fock), while the red line corresponds to a conventional two-body method (such as CCSD). Figure courtesy of Edward F. Valeev . . . . . 16

1.2 Rank reduction of the virtual space via approximation with truncated PNOs for a system consisting of 10 helium atoms each separated by 1 Å . . . . . . 24

1.3 A cartoon example of a two-dimensional periodic system, illustrating the major features of this framework. Each small block is a unit cell, while the entire system as pictured is an example of a supercell. The central unit cell, highlighted by the two red arrows, is the reference cell. . . . . . . . . . . . . 31

2.1 Statistical analysis of dimer absolute correlation energy percent errors. . . . . 51

2.2 Ratio of the PNO-CCSD dimer correlation energy percent error to the iPNO-CCSD dimer correlation energy percent error for all systems. . . . . . . . . . 52
2.3 Average number of PNOs/pair for three $\tau_{PNO}$ values, compared across all three methods. .......................................................... 53

2.4 Statistical analysis of binding energy absolute errors. ......................... 54

2.5 Statistical analysis of dimer energy absolute errors and perturbative energy corrections. ................................................................. 56

2.6 Statistical analysis of binding energy absolute errors and perturbative energy corrections. ................................................................. 57

2.7 Comparison of the accuracy of the various corrected and uncorrected schemes for dimer correlation energies. ................................. 58

2.8 Comparison of the accuracy of the various corrected and uncorrected schemes for binding energies. .................................................. 59

3.1 Plots of the difference between the PM functional value $P$ and the converged value $P_0$ vs the iteration count. For each system represented, the shortest calculation in each category was chosen; $P_0$ was taken to be the greatest final $P$ value across the selected calculations for a given system. (Continued on next page) .................................................. 91
3.1 (Continued from previous page) Plots of the difference between the PM functional value $P$ and the converged value $P_0$ vs the iteration count. For each system represented, the shortest calculation in each category was chosen; $P_0$ was taken to be the greatest final $P$ value across the selected calculations for a given system.
List of Tables

2.1 Twelve-System Subset of the S66 Benchmark Data Set Utilized for Initial Assessment of the iPNO-CCSD Approach ........................................ 50

2.2 Statistical Analysis of the S66 Binding Energy Errors, the DARC Reaction Energy Errors, and the BH76 Reaction Barrier Errorsa ........................................ 61

2.3 PNO Truncation Errors in CCSD cc-pVTZ Energy of the Equidistant H10 Chain 62

3.1 Test systems used to assess the performance of PMWF solvers, along with the relevant computational details. .................................................. 87

3.2 Statistical analyses of the number of iterations to solution, broken down by system and solver type. “CG” denotes the aggregate data of the 3 CG solvers. Computations that failed to converge in 4000 iterations were deemed “failed to converge” and were not included in the statistics. ........................................ 89
3.3 Statistical analyses of the number of iterations to solution for each BFGS solver parameter set. Colored boxes identify the smallest and largest values in each column as follows: green (smallest), blue (second smallest), red (largest), orange (second largest). A “Min” value of 1 indicates that this solver was the fastest (required the fewest iterations) for at least 1 test system. A “Max” value of 1.5 means that, for every test system, this solver required at most 50% more iterations than the fastest solver for that system. A “Mean” value of 1.2 means that, for a given test system, this solver required on average 20% more iterations than the fastest solver for that system.

A.1 P Value Summary .......................... 102

A.2 The number of iterations to solution for every parameter pair of the BFGS solver. .......................... 105

A.3 A statistical analysis of initial $P$, final $P$, and the number of iterations to solution vs. the initial guess type .......................... 107

A.4 Comparison of initial $P$, final $P$, and the number of iterations to solution vs. the minimal basis set employed .......................... 109

A.5 Comparison of maximum overlap condition number, initial $P$, final $P$, and the number of iterations to solution vs. the minimal basis set employed . . . 110
Chapter 1

Introduction

1.1 The Electronic Structure Problem

The underlying physical laws necessary for the mathematical theory of a large part of physics and the whole of chemistry are thus completely known, and the difficulty is only that the exact application of these laws leads to equations much too complicated to be soluble. It therefore becomes desirable that approximate practical methods of applying quantum mechanics should be developed, which can lead to an explanation of the main features of complex atomic systems without too much computation.¹

These words, penned by Paul A. M. Dirac in 1929, serve to define the central research focus of today’s computational quantum chemists, much as they did for the computational quantum...
chemists of nearly a century ago. Quantum mechanics as applied to the chemistry of atoms and molecules is, on the surface, seemingly simple. The wavefunction of a quantum system encodes all of the information about that system; knowledge of a system’s wavefunction allows us to further calculate other properties, such as the total energy of the system and various vibrational frequencies. As an example, consider the time-independent, non-relativistic Schrödinger equation, arguably the central equation of quantum mechanics, which relates a system’s wavefunction to the corresponding energy via an eigenvalue problem:

\[ \hat{H} \Psi = E \Psi \]  

(Although Erwin Schrödinger, who published this equation in 1926, occupies a central place in the development of quantum mechanics, it is important to note that his work was an extension of Werner Heisenberg’s earlier algebraic formulation of quantum mechanics.) As is often the case, we choose to work within the Born-Oppenheimer approximation, which assumes that the heavier nuclei may be treated as stationary relative to the much lighter and faster electrons; under this assumption, the Hamiltonian operator used in the Schrödinger equation is the time-independent, non-relativistic electronic Hamiltonian, which is defined for a system with \( N \) electrons and \( M \) nuclei as

\[ \hat{H}_{\text{elec}} = - \sum_{i=1}^{N} \frac{1}{2} \nabla_i^2 - \sum_{i=1}^{N} \sum_{A=1}^{M} \frac{Z_A}{r_{iA}} + \sum_{i=1}^{N} \sum_{j>i}^{N} \frac{1}{r_{ij}} \]  

In the case of a system containing a single electron, such as the neutral hydrogen atom, it is possible to find an exact analytical solution to the Schrödinger equation. When more electrons are added, however, the Schrödinger equation becomes non-separable, due to the
presence of the electron-electron repulsion term in the electronic Hamiltonian, thus making
the many-electron problem analytically insoluble. But despite this difficulty, quantum me-
chanics remains the most accurate theoretical means of describing the behavior of electrons,
making the discovery of accurate but computationally tractable approximate solutions to the
Schrödinger equation of the utmost importance. Unsurprisingly, much labor and many pages
have been devoted to this subject over the years, and it is the goal of this current work to
explain the ways, however small, in which we have expanded this important field.

1.2 Terminology and Notation

In the case of a one-electron system, solution of the Schrödinger equation yields a one-particle
function. If this function depends on the electron’s spin and spatial coordinates, then it is
called a spin orbital; if it depends only on the spatial coordinates, however, it is termed a
spatial orbital. Spin orbitals are often denoted by the Greek letter $\chi$, while spatial orbitals
are represented by the Greek letter $\phi$. Atom-centered one-electron orbitals are known as
atomic orbitals.

For systems with more than one electron, one-particle functions are no longer sufficient. In
the case of a non-interacting $N$-electron system, a product of $N$ spin orbitals satisfies the
Schrödinger equation, but this form fails for realistic systems in which the electrons interact.
A commonly used alternative is the Slater determinant, which satisfies the fermionic anti-
symmetry requirement. For an $N$-electron system with spin orbitals $\{\chi_1, \chi_2, \ldots, \chi_N\}$, the
The corresponding Slater determinant is defined as:

$$\psi = \frac{1}{\sqrt{N!}} \begin{vmatrix} \chi_1(x_1) & \chi_2(x_1) & \cdots & \chi_N(x_1) \\ \chi_1(x_2) & \chi_2(x_2) & \cdots & \chi_N(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \chi_1(x_N) & \chi_2(x_N) & \cdots & \chi_N(x_N) \end{vmatrix}$$

Equivalently, the above can also be written in a short-hand notation as

$$|\psi\rangle = |\chi_1(x_1)\chi_2(x_2)\cdots\chi_N(x_N)\rangle$$

where only the diagonal elements of the determinant are written out, and normalization is included implicitly.

The structure $| \rangle$ is known as a ket, while $| \rangle^* = \langle |$ is known as a bra. These two elements together form the basis for Dirac’s bra-ket notation. This notation provides a useful shorthand for describing integrals. To see this, consider the most commonly used integrals in quantum chemistry.

The first is a type of one-electron integral, the overlap integral:

$$\langle \phi_i | \phi_j \rangle = \int \phi_i^*(r)\phi_j(r)dr$$

The second is a type of two-electron integral, called, generally, an electron repulsion integral (ERI):

$$\langle \phi_i \phi_j | \phi_k \phi_l \rangle = \int \int \phi_i^*(r_1)\phi_j^*(r_2)r_{12}^{-1}\phi_k(r_1)\phi_l(r_2)dr_1dr_2$$

Finally, an anti-symmetrized ERI is given by

$$\langle \phi_i \phi_j | | \phi_k \phi_l \rangle = \langle \phi_i \phi_j | \phi_k \phi_l \rangle - \langle \phi_i \phi_j | \phi_l \phi_k \rangle$$
Bra-ket notation is also known as physicist’s notation, to distinguish it from the related chemist’s notation. In certain situations, it will be useful to adopt the chemist’s notation in the description of two-electron integrals:

\[
(\phi_i \phi_j | \phi_k \phi_l) = \int \int \phi_i^*(r_1) \phi_j(r_1) r_1^{-1} \phi_k^*(r_2) \phi_l(r_2) dr_1 dr_2 = \langle \phi_i \phi_k | \phi_j \phi_l \rangle \tag{1.8}
\]

### 1.3 One-Body Methods

As a first approximation to the solution to the Schrödinger equation for the many-electron system, we consider the main “one-body method,” Hartree-Fock, which is a wavefunction-based method. Other one-body methods exist, notably Kohn-Sham density functional theory (KS DFT), but since these methods do not commonly serve as the foundation for the many-body methods that will be discussed later in this work, we will not discuss them herein.

#### 1.3.1 Hartree-Fock

The Hartree-Fock (HF) approximation to the Schrödinger equation is a wavefunction-based method that seeks to find approximate eigenfunctions of the exact Hamiltonian. The result of a Hartree-Fock calculation is the best single-determinant approximation of a ground-state wavefunction, assuming each electron moves under an average potential due to all other electrons present in the system. By variationally optimizing the expectation value of the Hamiltonian operator, our approximate ground-state wavefunction is guaranteed to yield a ground-state energy that is an upper bound to the system’s true ground-state energy.
In practice, a Hartree-Fock calculation for an \( N \)-electron system is undertaken by first introducing a finite basis set of \( M \geq N \) atomic orbitals (AOs). These AOs will then be taken in linear combinations to form \( M \) molecular orbitals (MOs):

\[
|\psi_p\rangle = \sum_{\mu} |\phi_{\mu}\rangle c_{\mu p}
\] (1.9)

The expansion coefficients in Equation (1.9) are found by solution of the Roothaan equations,\(^5,14\) which can be written as a single generalized eigenvalue equation:

\[
\mathbf{FC} = \mathbf{SC}\epsilon
\] (1.10)

where \( \mathbf{F} \) is the Fock operator and \( \mathbf{S} \) is the matrix of orbital overlap elements. The \( N \) lowest energy MOs, as determined by their eigenvalues \( \epsilon_p \), are then used to form the single Slater determinantal Hartree-Fock wavefunction.\(^5\) The MOs defined by solution of Equation (1.10) are the canonical Hartree-Fock orbitals, but the Hartree-Fock wavefunction and corresponding properties are invariant to a unitary transformation of these occupied molecular orbitals.\(^15,16\)

In addition to being variational, the HF method has a number of other benefits. In particular, it is simple to understand and implement, and it generally provides qualitatively correct results.\(^5\) Furthermore, HF is usually size-consistent, which means that for a system composed of two identical, non-interacting species (such as two hydrogen molecules separated by an infinite distance), the total HF energy of the system is equal to twice the HF energy of one of the species.\(^17\) For the example of the two non-interacting hydrogen molecules, this means
that

\[ E_{(H_2)_2} = 2E_{H_2} \]  

(1.11)

Size-consistency in the energy is an indicator of the physical reasonableness of the underlying wavefunction; as such, it will be used to judge the quality of later computational methods. Despite the benefits of the Hartree-Fock method, it possesses a number of drawbacks as well, most notably concerning the accuracy of the HF results. Although HF is frequently qualitatively correct, this is not always the case, as can be seen with the benzene dimer. Hartree-Fock calculations fail to predict the boundedness of the benzene dimer, contrary to experimental evidence. Furthermore, Hartree-Fock results, even when qualitatively correct, are rarely quantitatively accurate enough for chemical purposes. Both of these accuracy issues stem from the inability of the Hartree-Fock method to address the correlation of electrons of opposite spin. Electron correlation is defined as

\[ E_{\text{corr}} = E - E_{HF} \]  

(1.12)

where \( E \) is the exact eigenvalue of the Hamiltonian in question and \( E_{HF} \) is the corresponding Hartree-Fock approximation to that eigenvalue, assuming an infinite basis set has been used. In practice, however, we generally use the term “correlation energy” to mean “basis set correlation energy,” which is the difference between the “exact” eigenvalue of the Hamiltonian and the corresponding Hartree-Fock eigenvalue, both computed using a particular basis set. Correlation energy is very important for the accurate description of atomic and molecular systems, and the development of new methods to better estimate the correlation energy of various systems is an active area of ongoing research.
1.4 Many-Body Methods

The Hartree-Fock method yields a reasonable first approximation to the wavefunction of a chemical system, but it is generally not accurate enough. Fundamentally, HF fails to correlate electrons with opposite spins, though it does partially correlate those with parallel spins. Furthermore, depending on the electronic structure of the particular system involved, the HF method can vary wildly in quality, and basis set incompleteness errors are always present. The introduction of various post-Hartree-Fock or many-body methods is an attempt to address the first of these issues, and in many cases, such methods enjoy a reasonable level of success. However, it is important to understand the various methods and their strengths and weaknesses because the particulars of the situation will dictate which method is best, and even the most rigorous post-Hartree-Fock method will fail if the reference is poor.

1.4.1 Perturbation Theory

Perturbation theory is a general technique commonly used in physics. The basic idea in perturbation theory is that an operator whose exact eigenfunctions and eigenvalues are unknown can be expressed as a sum of two individual operators; one of these operators, whose eigenfunctions and eigenvalues are known, is assumed to contribute a majority share to the original operator, while the other operator, whose exact eigenfunctions and eigenvalues are unknown, is assumed to contribute only a small perturbation. The goal of perturbation theory is to express the eigenfunctions and eigenvalues of the total operator as a linear
combination of the known eigenfunctions and eigenvalues of the simpler operator. Because
the perturbation is assumed to be small, it is believed that the sum will converge quickly to
the exact result, making it possible to compute good approximations with only a few terms.\(^5\)

In the field of computational quantum chemistry, the most common implementation of per-
turbation theory originated in the work of Rayleigh\(^23\) and Schrödinger\(^24\) and was later put
into today’s well-known form by Møller and Plesset.\(^25\) In Møller-Plesset Perturbation Theory
(MPPT), the unperturbed operator is the Hartree-Fock Hamiltonian operator;\(^5\) in the most
common implementation, second-order MPPT (MP2), the Hartree-Fock energy is the sum
of the zeroth- and first-order energies, while the correlation energy is approximated by a
single term, the second-order correction. The MP2 energy correction expressed in terms of
the canonical Hartree-Fock orbitals can be summarized succinctly as\(^5\)

\[
E^{(2)}_0 \left( \frac{1}{4} \sum_{ijab} \frac{|\langle ab||ij \rangle|^2}{\epsilon_a + \epsilon_b - \epsilon_i - \epsilon_j} \right)
\]

where \(\epsilon_p\) is the energy of molecular orbital \(p\) as determined in the Hartree-Fock procedure.

In practice, the necessary integrals are generated in terms of atomic orbitals (AOs) and then
transformed to molecular orbitals (MOs) at a cost of \(N^5\), where \(N\) is the number of basis
functions. Like Hartree-Fock, many-body perturbation theory is size-consistent; it is not,
however, variational.\(^5\)
1.4.2 Configuration Interaction

Within the Hartree-Fock approximation to the ground-state wavefunction of an \( N \)-electron system, the \( N \) eigenvectors of the Fock operator with the lowest eigenvalues provide the expansion coefficients for the \( N \) molecular orbitals occupied in the ground state of the system.\(^5\) The remaining \( M - N \) eigenvectors define the unoccupied or virtual MOs. Although these orbitals are a poor approximation to the excited states of the system, they are useful for lowering the calculated energy of the ground state.\(^{20,22}\)

The ground-state wavefunction calculated in the HF method, \( \Psi_{HF} \), is represented by a Slater determinant consisting of the \( N \) lowest energy MOs, where each MO is itself a linear combination of the basis functions:\(^5\)

\[
|\psi_i\rangle = \sum_\mu |\phi_\mu\rangle c_{\mu i} \tag{1.14}
\]

\[
|\Psi_{HF}\rangle = |\psi_1\psi_2 \ldots \psi_N\rangle \tag{1.15}
\]

Clearly, though, Equation (1.15) is not the only Slater determinant that can be formed from the MOs generated by the HF process. We can form \( M - N \) new Slater determinants simply by systematically replacing the first orbital in \( |\Psi_{HF}\rangle \) with each of the \( M - N \) virtual orbitals:

\[
|\Psi_{1}^{N+1}\rangle = |\psi_{N+1}\psi_2 \ldots \psi_N\rangle
\]

\[
|\Psi_{1}^{N+2}\rangle = |\psi_{N+2}\psi_2 \ldots \psi_N\rangle
\]

\[
\vdots
\]

\[
|\Psi_{1}^{M}\rangle = |\psi_M\psi_2 \ldots \psi_N\rangle
\]
where the notation $|\Psi_{i}^{a}\rangle$ means that occupied MO $\psi_{i}$ has been replaced with unoccupied MO $\psi_{a}$, relative to the reference Slater determinant (in this case, the HF Slater determinant). The process of replacing one orbital with another within a Slater determinant is called a single replacement or a single excitation, and the resulting Slater determinant is known as a singly-excited Slater determinant. Furthermore, we can create even more Slater determinants by systematically replacing $K$ orbitals in the reference Slater determinant, where $K \in \{1, 2, \ldots, N\}$.\textsuperscript{20,22}

We can take advantage of these additional determinants to help lower the ground-state energy in the following manner. After expressing the ground-state wavefunction $|\Psi_{0}\rangle$ as a linear combination of the reference determinant and all excited determinants up to a chosen excitation level, we will then use a process very similar to that employed in the HF method to systematically vary the expansion coefficients and thereby variationally determine the best form for the ground-state wavefunction.\textsuperscript{22} This method is referred to generally as Configuration Interaction (CI). If all possible excitation levels are included, then we have a full CI (FCI) calculation; otherwise, we are performing a truncated CI calculation.\textsuperscript{20,22}

A full CI calculation is exact within the space spanned by the basis set, but, in practice, FCI calculations are impossible to perform in all but the smallest of cases. Instead, the excitations are truncated at a predetermined level, usually double excitations.\textsuperscript{22} A CI calculation that includes only single and double excitations is a CISD calculation. Double excitations can be shown to have the greatest effect on the ground-state energy, while single excitations prove important in accurately computing other ground-state properties.\textsuperscript{5} Higher excitations
do not contribute as much to the energy as double excitations do, and since singly-excited determinants do not interact directly with the ground-state reference wavefunction, CIS calculations are not useful for ground-state calculations.\textsuperscript{16}

The Configuration Interaction method is very important in its potential to generate exact results, making it an important benchmarking tool for subsequent quantum chemical methods.\textsuperscript{22} Unfortunately, truncated CI (regardless of the truncation level) suffers from a number of problems that rather diminish its usefulness. First of all, truncated CI is not size-consistent.\textsuperscript{16} Furthermore, as more and more non-interacting units are added to the system, the total energy calculated using truncated CI becomes a worse and worse approximation. This latter situation is an example of non-size-extensivity.\textsuperscript{16,17}

\subsection{1.4.3 Coupled Cluster}

In the previous section, we discussed the Configuration Interaction method, which is a conceptually simple means of recovering some of a system’s correlation energy. Unfortunately, however, full CI is very computationally intensive, and truncated CI performs poorly on large systems, which are often the focus of chemical problems of interest.\textsuperscript{16} In this section, we discuss an alternative to CI known as the Coupled Cluster (CC) method.\textsuperscript{26–28} Unlike CI, CC is not variational, but it is size-consistent, and depending on the variant used, CC can be extremely accurate.\textsuperscript{17}

Like CI, CC is based on generating excited Slater determinants and using these, along with
the reference determinant, to generate a more accurate approximation to the ground-state wavefunction and energy. To see what makes CC different, though, requires looking at the operator used to generate the excited configurations.

Within the mathematical formalism known as second quantization, any Slater determinantal wavefunction can be formed by the repeated application of appropriate creation and annihilation operators on a reference determinant.\(^{29}\) Creation operators, which take the form \(a^\dagger_p = a^p\), serve to add an electron to orbital \(p\) within a Slater determinant, while annihilation operators, which have the form \(a_p\), have the effect of removing an electron from orbital \(p\).\(^{29}\) In the case of the CC method, the reference determinant is nearly always the HF ground-state wavefunction, and the excitation operators have the following form:\(^{17}\)

\[
\hat{T}_n = \left( \frac{1}{n!} \right)^2 \sum_{i,j,...} t_{ij,...} a^\dagger_a a^\dagger_b ... a_j a_i
\]

where \(n\) refers to the particular excitation level (single, double, etc.). The \(t_{ij,...}\) are expansion coefficients (called “amplitudes” in this particular context).\(^{17}\)

In the full CC method as applied to an \(N\)-electron system, a \(\hat{T}\) operator is defined that is simply the sum of all possible \(\hat{T}_n\) operators:\(^{17}\)

\[
\hat{T} = \sum_{n=1}^{N} \hat{T}_n
\]

Up until this point, the CC method appears very similar to the CI method, which also defined an operator that was the sum of all excitation operators:\(^{17}\)

\[
\hat{C} = \sum_{n=1}^{N} \hat{C}_n
\]
The crucial difference between the two methods, however, is in how the excitation operator is applied to the reference wavefunction. In the case of CI, the CI wavefunction was formed directly via application of the CI cluster operator to the HF wavefunction, while in CC, it is an exponentiated operator that is applied to the HF wavefunction:

$$e^{\hat{T}} |\Psi_{HF}\rangle = |\Psi_{CC}\rangle$$  \hspace{1cm} (1.20)

The presence of this exponentiated operator is the genesis of the term *exponential ansatz*.

In order to see the benefits of an exponential operator, let us substitute the CC wavefunction into the Schrödinger equation, using $|\Psi\rangle = e^{\hat{T}} |\Psi_{HF}\rangle$:

$$\hat{H}e^{\hat{T}} |\Psi_{HF}\rangle = E e^{\hat{T}} |\Psi_{HF}\rangle$$  \hspace{1cm} (1.21)

Next, we will left multiply both sides of the above expression by $e^{-\hat{T}}$, which is simply the inverse of $e^{\hat{T}}$:

$$e^{-\hat{T}} \hat{H} e^{\hat{T}} |\Psi_{HF}\rangle = e^{-\hat{T}} E e^{\hat{T}} |\Psi_{HF}\rangle$$  \hspace{1cm} (1.22)

Then, because $E$ is a scalar, it can be moved out in front, which allows us to take advantage of the fact that $e^{-\hat{T}} e^{\hat{T}} = 1$:

$$e^{-\hat{T}} \hat{H} e^{\hat{T}} |\Psi_{HF}\rangle = E e^{-\hat{T}} e^{\hat{T}} |\Psi_{HF}\rangle = E |\Psi_{HF}\rangle$$  \hspace{1cm} (1.23)

Finally, left multiplying the preceding expression first by $\langle \Psi_{HF} |$ and then by various excited determinants of the form $\langle \Psi_{ij...}^{ab...} |$ yields the two central equations of CC theory:

$$\langle \Psi_{HF} | e^{-\hat{T}} \hat{H} e^{\hat{T}} |\Psi_{HF}\rangle = E \langle \Psi_{HF} |\Psi_{HF}\rangle = E$$  \hspace{1cm} (1.24)

$$\langle \Psi_{ij...}^{ab...} | e^{-\hat{T}} \hat{H} e^{\hat{T}} |\Psi_{HF}\rangle = E \langle \Psi_{ij...}^{ab...} |\Psi_{HF}\rangle = 0$$  \hspace{1cm} (1.25)
where the simplification in Equation (1.24) comes from the fact that \( \Psi_{HF} \) is normalized and the simplification in Equation (1.25) comes from the fact that \( \Psi_{HF} \) is orthogonal to all excited determinants formed from it.\(^{17}\)

The two central equations of CC theory given above are applicable to any level of excitation but, in practice, calculations involving triple and higher excitations are generally not performed except for benchmarking purposes.\(^{16,17,31–34}\) Rather, it is common to perturbatively estimate triple excitations, leading to the CCSD(T) method,\(^{35}\) which is frequently referred to as the gold standard of quantum chemistry for its high accuracy and reasonably widespread applicability.\(^{36}\)

### 1.5 Approximation Techniques

Throughout the preceding text, we have discussed a number of conventional many-body methods, alluding to their potential for great accuracy but also alluding to their potential for exorbitant computational cost, as measured in terms of scaling with approximate system size \( N \) (here, \( N \) does not mean the number of electrons, as it did earlier in this work). That conventional many-body methods quickly change from being merely unwieldy to patently unusable is evident in looking at Figure 1.1, which plots the estimated time to solution for a series of systems consisting of an increasing number of uracil bases. The blue line corresponds to a conventional one-body method (such as Hartree-Fock, which scales formally as \( N^4 \))\(^{37}\), while the red line corresponds to a conventional two-body method (such as CCSD, which
scales as $N^6$).\textsuperscript{38}

![Figure 1.1: Estimated time to solution for systems consisting of increasing numbers of uracil bases. The blue line corresponds to a conventional one-body method (such as Hartree-Fock), while the red line corresponds to a conventional two-body method (such as CCSD). Figure courtesy of Edward F. Valeev](image)

If we are ever to apply highly-accurate quantum chemical methods to real-life systems of interest, it is obvious that something must change. The answer, though, is not simply to use bigger computers (and more of them); our appetite for computation will always outpace our computational resources, and many, many systems will remain perpetually outside of our grasp.\textsuperscript{39} Rather, the key is to fundamentally address the costs associated with these methods, bringing their computational scaling down to more manageable levels. Such scaling-reduction
techniques are discussed in Section 1.5.1.

Another aspect of cost that is not directly illustrated by Figure 1.1 is that of computational prefactor. Although the computational prefactor of a method (or a step in a method) does not often get as much press as that same method’s computational scaling, it can nevertheless play a substantial role in determining how expensive a method is and to what systems a method can be applied. Though prefactor reduction is not directly addressed by the research reported herein, we include a discussion of density fitting (DF), an oft-used and very successful example of a technique for reducing a computational prefactor. This discussion, found in Section 1.5.2, is included not only for the sake of completeness, but also because the work reported in Chapter 3 relies heavily on an implementation of Hartree-Fock that itself owes much to density fitting.

1.5.1 Local Correlation

Though there are different tactics that have been taken to reduce the scaling of the many-body methods discussed herein, we have chosen to focus on the fact that, fundamentally, electron correlation is a primarily local phenomenon. Indeed, the high scaling associated with, say, CCSD ($N^6$) or CCSD(T) ($N^7$) is, in fact, unphysical, and it is our goal to realize correlated methods whose computational scaling is more in line with the physical nature of the problem. Again, we are here faced with a choice for how to proceed, and we have chosen what has been termed the “direct” local correlation path over that of local
correlation via fragments.\textsuperscript{44,48}

The work that follow discusses two major aspects of developing local correlated many-body methods: orbital localization and virtual space compression. These are not the only pieces of the puzzle, but they are the two we have chosen to focus on. Orbital localization is discussed in a cursory manner in Section 1.5.1 and then again in a more detailed manner in Chapter 3; virtual space compression is the focus of both Section 1.5.1 and Chapter 2.

**Orbital Localization**

The canonical molecular orbitals determined by a Hartree-Fock calculation are by nature delocalized over a large region of space.\textsuperscript{7,16} Because of the presence of $r^{-1}_{12}$ in the electron repulsion integrals, post-Hartree-Fock methods must take into account many integrals of fairly small magnitude.\textsuperscript{17} If, on the other hand, the MOs were each localized to a relatively small region of space, then many of the ERIs would become negligible and could be neglected \textit{a priori} using distance cutoffs.\textsuperscript{16,49} Furthermore, localized molecular orbitals correspond more closely with our intuitive picture of chemical bonding,\textsuperscript{50} and provided that the orbitals are localized via a unitary transformation of the canonical orbitals, the calculated energy of the system is unchanged.\textsuperscript{51}

Numerous methods for localizing orbitals have been proposed throughout the years. Some of the earliest work on localized orbitals comes from the 1940s, with Coulson\textsuperscript{52} using localized orbitals to better model the dipole moment of the carbon-hydrogen bond in methane and
Lennard-Jones and Pople\textsuperscript{53} arguing that their so-called “equivalent orbitals” allowed for a better qualitative understanding of the energy of a molecular system. Somewhat later schemes were proposed by Boys and Foster,\textsuperscript{54,55} Edmiston and Ruedenberg,\textsuperscript{15,56} and Pipek and Mezey,\textsuperscript{57} all of which continue to be used today.

The localization scheme of Boys and Foster seeks to physically separate the “centroids of charge”\textsuperscript{55} as much as possible and does so by maximizing the following quantity:\textsuperscript{50}

\[ J(\phi) = \sum_{i>j}^{n} \left[ \langle \phi_i | r | \phi_i \rangle - \langle \phi_j | r | \phi_j \rangle \right]^2 \]  

(1.26)

Edmiston and Ruedenberg, on the other hand, approached the problem from a somewhat different angle. In their localization scheme, they sought to maximize the sum of the orbital self-repulsion energies\textsuperscript{15,50,56}:

\[ J_s(\phi) = \sum_{i=1}^{n} \langle \phi_i \phi_i | \phi_i \phi_i \rangle \]  

(1.27)

Finally, Pipek and Mezey used Mulliken population analysis\textsuperscript{58,59} to define an expression for the number of atoms over which molecular orbital \(i\) extends:\textsuperscript{60}

\[ d_i = \left( \sum_{A=1}^{M} (Q^i_A)^2 \right)^{-1} \]  

(1.28)

The individual \(d_i\) are then used to form the quantity \(D^{-1}\):

\[ D^{-1} = \frac{1}{n} \sum_{i=1}^{n} d_i^{-1} \]  

(1.29)

The appropriate orbital transformation is then found by maximization of \(D^{-1}\) (or, equivalently, minimization of \(D\)).\textsuperscript{57}
Although localization is more commonly applied to occupied molecular orbitals, it is possible to localize virtual molecular orbitals as well. Jørgensen and co-workers applied the Pipek-Mezey scheme to the localization of virtual molecular orbitals with a modicum of success, deeming the resulting virtual orbitals semilocal and highly dependent upon the particulars of the system in question.

**Virtual Space Compression**

Whereas orbital localization applies primarily to occupied molecular orbitals, this next approximation technique considers mainly the virtual orbitals. Certain sets of virtual orbitals will lead to more rapidly converging calculations, which in turn means that fewer orbitals are required to reach a certain level of accuracy. By determining and employing these improved virtual orbitals, we can often appreciably decrease the number of matrix elements that must be computed and stored, making certain high-accuracy methods applicable to larger systems than was previously possible.

**Projected Atomic Orbitals**  Projected atomic orbitals (PAOs) provide a means of localizing the virtual space. As their name suggests, PAOs are formed by projecting the atomic orbitals against the reference MOs (for example, HF MOs). The AOs can be projected against both the occupied and the virtual MOs, with the appropriate projection operators given by, respectively

\[ \hat{P} = \sum_{i=1}^{n} |\psi_i\rangle \langle \psi_i| \]  

(1.30)
\[ \hat{Q} = \sum_{a=n+1}^{N} |\psi_a\rangle \langle \psi_a| \]  

Here, \( N \) refers to the total number of MOs. Projecting the AOs leads to the PAO basis, which consists of either \( N \) functions or \( 2N \) functions, depending on whether the AOs are projected against just the virtual MOs or against the occupied and virtual MOs. In either case, the result is a set of linearly-dependent, \( n \) non-orthogonal functions:

\[ \{ |\phi_K\rangle \} = \{ \hat{P} |\chi_\alpha\rangle \} \]  

\[ \{ |\phi_A\rangle \} = \{ \hat{Q} |\chi_\alpha\rangle \} \]  

Those PAOs that are formed by projecting against the virtual MOs are guaranteed to be orthogonal to the occupied MOs. In addition, each PAO will be localized to generally the same region of space as the corresponding AO from which it was formed.\(^{46}\) Compression of the virtual space comes about by only allowing each occupied orbital to correlate with virtual orbitals within a specified domain.\(^{47,63}\)

**Pair Natural Orbitals**  Although the groundwork for modern pair natural orbitals was laid in the late 1960s and early 1970s, it is only rather recently that they have become popular in the larger computational quantum chemistry community. Originally proposed by Edmiston and Krauss\(^{64,65}\) under the name pseudonatural orbitals and then expanded upon by Meyer,\(^{66,67}\) pair natural orbitals (PNOs) were put into their modern form by Neese et al.\(^{68}\) in 2009. Since then, there has been great interest in expanding the use of PNOs within numerous quantum chemical methods, in addition to growing the number of parallel implementations that involve PNOs.
Although the method by which the PNOs are formed has varied somewhat over the years, the one we use here follows the treatment of Yang et al. The identifying feature of this algorithm is its use of the MP1 amplitudes to form the PNOs. Specifically, the exchange integrals, $g_{ab}^{ij}$, are first computed

$$g_{ab}^{ij} = \langle ai | bj \rangle = \langle ab | ij \rangle$$

and these are then used to form the MP1 amplitudes, $T_{ab}^{ij}$:

$$T_{ab}^{ij} = \frac{-g_{ab}^{ij}}{\epsilon_a + \epsilon_b - \epsilon_i - \epsilon_j}$$

For a given $i, j$ combination, there exist two matrices, $T^{ij}$ and $T^{ji}$, with the elements indexed by $(a, b)$. Together, these two amplitude matrices are used to form another amplitude matrix, $\tilde{T}^{ij}$, which is the closed-shell, spin-free analog of an anti-symmetrized $\bar{T}^{ij}$ matrix:

$$\tilde{T}^{ij} = 2T^{ij} - T^{ji}$$

These amplitude matrices are then used to form a central quantity, the $D^{ij}$ density matrix:

$$D^{ij} = \frac{1}{1 + \delta_{ij}} \left( \tilde{T}^{ij\dagger} T^{ij} + \tilde{T}^{ij} T^{ij\dagger} \right)$$

Diagonalization of $D^{ij}$ yields the PNOs (the eigenvectors of $D^{ij}$) and the corresponding occupation numbers (eigenvalues):

$$D^{ij} \mathbf{V}^{ij} = \mathbf{V}^{ij} \Lambda^{ij}$$

For each $i, j$ pair, the corresponding matrix of PNOs can then be used to perform a similarity transformation on the virtual orbitals. If the full PNO matrix is used, then this similarity
transformation is also a unitary transformation, which serves to rotate the virtual space, and the final energy calculated using these rotated orbitals will not vary from the energy computed using the original orbitals.\textsuperscript{51,70} However, if the matrix of PNOs is truncated somehow, then the corresponding transformation of the virtual space serves to compress the space.

In order to take advantage of a compressed virtual space, it is necessary to truncate the PNO space in an intelligent manner. That is, we want to determine which PNOs contribute the least to the final answer, and it is these that we will discard. As it turns out, this is straightforward to accomplish. The occupation numbers provide a measure of the importance of a particular PNO: the greater the occupation number, the larger the effect that particular PNO will have on the final answer. In practice, a parameter, known by various titles in various sources\textsuperscript{71,72} but herein referred to as $\tau_{\text{PNO}}$, is chosen, and all PNOs whose occupation numbers are smaller than $\tau_{\text{PNO}}$ are discarded. As a striking illustration of the rank reduction that the PNOs afford, as well as the locality of the PNOs, consider Figure 1.2, which plots the number of PNOs per $i,j$ pair for a system of ten helium atoms in a row, each separated by one angstrom. Each small square corresponds to the full virtual space (130 elements by 130 elements) while the smaller red rectangles correspond to the number of PNOs kept after truncation. The rank reduction is especially impressive for occupied orbitals that are far apart spatially. For this particular case, $\tau_{\text{PNO}} = 1.0 \times 10^{-7}$.

In many cases, a sufficiently large amount of the standard CCSD correlation energy can be recovered with a significant reduction in the size of the virtual space. Please see Chapter 2 for more specific details.
Figure 1.2: Rank reduction of the virtual space via approximation with truncated PNOs for a system consisting of 10 helium atoms each separated by 1 Å

**Orbital Specific Virtuals** As their name suggests, pair natural orbitals are virtual orbitals specific to a pair of occupied orbital indices. An alternative approach is to define natural orbitals relative to a single occupied orbital index, which is the logic behind orbital specific virtuals (OSVs). First proposed by Yang et al., OSVs are formed in exactly the same manner as PNOs, with the only differences being that $i = j$ in all of the above PNO equations, and a different truncation threshold, $\tau_{OSV}$, is employed instead. PNOs and OSVs
are often employed in conjunction, as will be discussed later, and when this is the case, $\tau_{\text{OSV}}$ is generally chosen to be a couple of orders of magnitude smaller than $\tau_{\text{PNO}}$.\textsuperscript{73} It is noted that OSVs form somewhat of a middle ground between PAOs and PNOs: whereas PAOs define a single virtual space accessible to all occupied orbitals and PNOs define a virtual space for each pair of (correlated) occupied orbitals, OSVs define a virtual space unique to each occupied orbital.\textsuperscript{74}

**Frozen Natural Orbitals** Finally, we discuss a virtual space compression method that is closely related to that of PNOs and OSVs: frozen natural orbitals (FNOs). Like PNOs and OSVs, FNOs are descendents of Löwdin’s natural spin orbitals\textsuperscript{75} and are formed as the eigenvectors of an appropriate density matrix. The eigenvalues are the occupation numbers, and truncation based on occupation number makes it possible to select a suitable virtual space that is smaller than the full virtual space but that recovers most of the important information contained in the full virtual space (of course, “most” is subjective and will depend on the particulars of a given calculation).\textsuperscript{76,77} The word “frozen” refers to the fact that the occupied orbitals are left untouched and only the virtual-virtual block of the density matrix is diagonalized. In particular, the density matrix we are concerned with is the MBPT(2) unrelaxed density matrix, which is defined as\textsuperscript{76}

$$D_{ab}^{(2)} = \frac{1}{2} \sum_{cij} \frac{\langle cb \rvert ij \rangle \langle ij \rvert ca \rangle e_{ij}^{cb} e_{ij}^{ca}}{e_{ij}^{cb} e_{ij}^{ca}}$$ \hfill (1.39)
The indices $i$ and $j$ refer to occupied orbitals while $a$ and $b$ refer to virtual ones. The terms in the denominator are given by\textsuperscript{76}

$$
\epsilon_{ij}^{ab} = f_{ii} + f_{jj} - f_{aa} - f_{bb}
$$

(1.40)

where $f_{pp}$ is a diagonal element of the Hartree-Fock Fock matrix.

### 1.5.2 Density Fitting

Though applying the techniques of local correlation to conventional many-body methods has made possible much larger calculations than were ever thought possible (consider the oft-cited example of crambin\textsuperscript{78}), these techniques do not operate in a vacuum. As noted by Schütz and Manby in Ref. \textsuperscript{62}, when the local coupled cluster with singles and doubles (LCCSD) method is employed, the evaluation and transformation of the electron repulsion integrals becomes the computational bottleneck. Thankfully, though, density fitting provides a means of substantially decreasing the costs associated with these integrals while generally introducing only relatively small errors.\textsuperscript{62}

To understand how density fitting is applied, we begin by considering the generation of the four-index electron repulsion integrals. Such integrals, which take the form

$$
g_{\mu\nu,\rho\sigma} = (\mu\nu|\rho\sigma)
$$

(1.41)

are generated naïvely in terms of AOs with complexity $\mathcal{O}(N^4)$, where $N$ is the number of atom-centered basis functions. To transform these integrals from AOs to MOs is $\mathcal{O}(N^5)$ in complexity.\textsuperscript{16}
By using screening techniques,\textsuperscript{79} it is possible to decrease the number of such integrals that must be computed, but such techniques are not generally sufficient for large systems.\textsuperscript{40} Instead, we focus here on a method for decreasing the prefactor of the integral calculation cost; though we use the density fitting terminology in this work, it is important to note that this same technique is also sometimes referred to as resolution of the identity (RI).\textsuperscript{80}

Although there exist a number of variations of density fitting, the basic idea behind all is the same: to approximate a four-index integral as three- and two-index integrals. To see how this can be done, consider in more detail the four-index integral given above:

\[ g_{\mu\nu,\rho\sigma} = (\mu\nu|\rho\sigma) = \int \chi_\mu^*(r_1)\chi_\nu(r_1) \frac{1}{r_{12}} \chi_\rho^*(r_2)\chi_\sigma(r_2) dr_1 dr_2 \]  

(1.42)

The product $\rho\sigma$ can be expanded as a linear combination of functions chosen from a suitable auxiliary basis set.\textsuperscript{40}

\[ |\rho\sigma) = \sum_X C^X_{\rho\sigma}|X) \]  

(1.43)

If the set of auxiliary functions used in the expansion is complete, then the expansion is exact, but in practice, this is impossible. A more accurate representation of this expansion would thus be

\[ |\rho\sigma) \approx |\tilde{\rho}\tilde{\sigma}) \equiv \sum_X C^X_{\rho\sigma}|X) \]  

(1.44)

where the tilde indicates a density-fitted quantity.\textsuperscript{81} Substituting this expansion in for $|\rho\sigma)$ in our original four-index integral yields

\[ g_{\mu\nu,\rho\sigma} \approx (\mu\nu|\tilde{\rho}\tilde{\sigma}) = \sum_X C^X_{\rho\sigma}(\mu\nu|X) \]  

(1.45)
which is, of course, now a three-index integral. Similarly, expanding the quantity $\mu\nu$ leads to a two-index integral:

$$g_{\mu\nu,\rho\sigma} \approx (\tilde{\mu}\tilde{\nu}|\tilde{\rho}\tilde{\sigma}) = \sum_{XY} C^Y_{\mu\nu} (Y|X) C^X_{\rho\sigma} \quad (1.46)$$

In order to take advantage of these two- and three-index integrals, it is necessary to first determine the expansion coefficients used above. This can be done for each $\rho\sigma$ product by projecting by every member of the auxiliary basis:

$$(Y_i|\rho\sigma) = \sum_j C^{X_j}_{\rho\sigma} (Y_i|X_j) \quad (1.47)$$

and solving the resulting matrix equation:

$$
\begin{bmatrix}
(Y_1|X_1) & (Y_1|X_2) & \ldots & (Y_1|X_n) \\
(Y_2|X_1) & (Y_2|X_2) & \ldots & (Y_2|X_n) \\
\vdots & \vdots & \ddots & \vdots \\
(Y_n|X_1) & (Y_n|X_2) & \ldots & (Y_n|X_n)
\end{bmatrix}
\begin{bmatrix}
C^{X_1}_{\rho\sigma} \\
C^{X_2}_{\rho\sigma} \\
\vdots \\
C^{X_n}_{\rho\sigma}
\end{bmatrix}
= 
\begin{bmatrix}
(Y_1|\rho\sigma) \\
(Y_2|\rho\sigma) \\
\vdots \\
(Y_n|\rho\sigma)
\end{bmatrix} \quad (1.48)
$$

Because the matrix $(Y|X)$ is the same for every $\rho\sigma$ pair, it only needs to be decomposed
once and we can thus solve for all expansion coefficients simultaneously:

\[
\begin{bmatrix}
(Y_1|X_1) & (Y_1|X_2) & \ldots & (Y_1|X_n) \\
(Y_2|X_1) & (Y_2|X_2) & \ldots & (Y_2|X_n) \\
\vdots & \vdots & \ddots & \vdots \\
(Y_n|X_1) & (Y_n|X_2) & \ldots & (Y_n|X_n)
\end{bmatrix}
\begin{bmatrix}
C_{\rho_1\sigma_1}^{X_1} & C_{\rho_1\sigma_2}^{X_1} & \ldots & C_{\rho_n\sigma_n}^{X_1} \\
C_{\rho_1\sigma_1}^{X_2} & C_{\rho_1\sigma_2}^{X_2} & \ldots & C_{\rho_n\sigma_n}^{X_2} \\
\vdots & \vdots & \ddots & \vdots \\
C_{\rho_1\sigma_1}^{X_n} & C_{\rho_1\sigma_2}^{X_n} & \ldots & C_{\rho_n\sigma_n}^{X_n}
\end{bmatrix}
= 
\begin{bmatrix}
(Y_1|\rho_1\sigma_1) & (Y_1|\rho_1\sigma_2) & \ldots & (Y_1|\rho_n\sigma_n) \\
(Y_2|\rho_1\sigma_1) & (Y_2|\rho_1\sigma_2) & \ldots & (Y_2|\rho_n\sigma_n) \\
\vdots & \vdots & \ddots & \vdots \\
(Y_n|\rho_1\sigma_1) & (Y_n|\rho_1\sigma_2) & \ldots & (Y_n|\rho_n\sigma_n)
\end{bmatrix}
\]

(1.49)

In the above expressions, it is implied that the $r_{12}^{-1}$ integration kernel is being used. This is not, however, the only choice. Other possible fitting metrics include the overlap metric and the attenuated Coulomb metric, but both suffer from an increase in error relative to the Coulomb metric when employed in the above expressions.

Besides the choice of fitting metric, the other major aspect of density fitting that can be varied is the set of auxiliary basis functions used to fit the two-electron density. A number of different possibilities exist, including the use of auxiliary functions localized in some way relative to the product that is being approximated. Likewise, the size of the auxiliary basis necessary to achieve a desired level of accuracy will also differ from calculation to calculation, but an auxiliary basis set that is about three times the size of the orbital basis set is often
a reasonable choice.\textsuperscript{40,84,85}

1.6 Extension to Periodic Solids

Up until this point, we have focused our attention on the electronic structure of molecules, a natural starting point for the computational chemist. But even though the majority of the electronic structure literature is concerned with gas phase systems, this is by no means the whole picture. Increasingly, those in the quantum chemistry community are turning their attention to the study of condensed phases. Whereas, in the the past, density functional theory (DFT) was overwhelmingly seen as the method of choice for such systems,\textsuperscript{41,86} it is no longer the default, and much promising work on wavefunction-based methods for (periodic) solids has emerged lately. In the text that follows, we will discuss the vocabulary and conventions used in the study of the electronic structure of periodic solids and the extension of the Hartree-Fock method to such systems. This will pave the way for our detailed discussion of orbital localization in periodic solids, which can be found in Chapter 3. We note that, in writing Sections 1.6.1 and 1.6.2, we are deeply indebted to Lucjan Piela’s Ideas of Quantum Chemistry (Ref. 87), which contains one of the most thorough, yet also most clear, introductions to this topic that we have come across.
1.6.1 Terminology and Notation

We begin by considering an infinite periodic solid. That the system be periodic is of the utmost importance, as it is the periodicity that makes these formally infinite calculations possible.\(^8\) The system may be one-, two-, or three-dimensional. Figure 1.3 is a cartoon example of a two-dimensional system, which will serve to illustrate the major features of this framework.

![Figure 1.3: A cartoon example of a two-dimensional periodic system, illustrating the major features of this framework. Each small block is a unit cell, while the entire system as pictured is an example of a supercell. The central unit cell, highlighted by the two red arrows, is the reference cell.](image)

From Figure 1.3, we see that the entire system is made up of a series of unit cells translated through space. In this case, our unit cell is orthorhombic, which simply means that each
of the basis vectors, indicated by the red arrows labeled ‘$a_1$’ and ‘$a_2$’, are orthogonal to each other; the lengths of the basis vectors are collectively known as the lattice constants, and each basis vector defines one of the periodicity axes. The contents of a single unit cell is known as the motif. We consider a finite number of unit cells, which together form a supercell.

Each unit cell is defined by its translation vector relative to the reference unit cell (the central cell), where cell $R_j$ has translation vector

$$ R_j = \sum_{i=1}^{d} n_{ji} a_i $$

and the sum is over the dimensions of the lattice (in the case of our cartoon system, $d = 2$).

By convention, the reference unit cell is referred to as $R_0$ with each $n_{0i} = 0$. To travel from any point within a given unit cell to the same point within another unit cell is simply a matter of shifting by an integer multiple of each of the basis vectors. The collection of points defined by all translation vectors makes up the primitive lattice.

In order to apply wavefunction-based electronic structure methods to periodic solids, it is necessary to introduce a basis set, which, in our case, will simply be a standard set of atom-centered Gaussian functions. Each Gaussian will now be a function of electron position $r$ relative to the unit cell in which it is centered. (As we will see in Chapter 3, it will sometimes be convenient to indicate on which atom a particular function is located, but for the current discussion, that is unnecessary.) Thus, for example, basis function $\chi_\mu$ in the reference cell.

\footnote{We note that the distinction between $R_j$ and $R_j$ is not always rigorously made, with different sources following different conventions. The particular meaning of a given $R$ will generally be evident from context.}
will be given by \( \chi_\mu(\mathbf{r} - \mathbf{R}_0) \equiv \chi_\mu(\mathbf{r}) \), and basis function \( \chi_\mu \) in unit cell \( R_j \) will be given by \( \chi_\mu(\mathbf{r} - \mathbf{R}_j) \). The two functions \( \chi_\mu(\mathbf{r}) \) and \( \chi_\mu(\mathbf{r} - \mathbf{R}_j) \) are related by the translation operator \( \hat{T}(\mathbf{R}_j) \), whose behavior is defined for a function \( f(\mathbf{r} - \mathbf{R}_i) \) as

\[
\hat{T}(\mathbf{R}_j) f(\mathbf{r} - \mathbf{R}_i) = f(\mathbf{r} - \mathbf{R}_i - \mathbf{R}_j) \tag{1.51}
\]

The translation operators \( \{\hat{T}(\mathbf{R}_j)\} \) commute with the Hamiltonian operator \( \hat{H} \), so that the eigenfunctions of the Hamiltonian operator are also eigenfunctions of the translation operators. In particular, for some eigenfunction \( f(\mathbf{r} - \mathbf{R}_i) \), the eigenvalue is given by

\[
\hat{T}(\mathbf{R}_j) f(\mathbf{r} - \mathbf{R}_i) = e^{-i\mathbf{k}\cdot\mathbf{R}_j} f(\mathbf{r} - \mathbf{R}_i) \tag{1.52}
\]

where \( e^{-i\mathbf{k}\cdot\mathbf{R}_j} \) is a shorthand for \( e^{-i\mathbf{k}\cdot\mathbf{R}_j} \). (Throughout this work, we will use an italicized \( i \) as an index and a non-italicized \( i \) for the imaginary unit; thus, the eigenvalue in Eq. (1.52) is complex.) By combining Eqs. (1.51) and (1.52), we arrive at what is known as the Bloch theorem, arguably the central equation of wavefunction-based methods for periodic solids:

\[
f(\mathbf{r} - \mathbf{R}_i - \mathbf{R}_j) = e^{-i\mathbf{k}\cdot\mathbf{R}_j} f(\mathbf{r} - \mathbf{R}_i) \tag{1.53}
\]

For our purposes, the Bloch theorem will be most conveniently written with respect to a function centered in the reference unit cell, \( f(\mathbf{r} - \mathbf{R}_j) = e^{-i\mathbf{k}\cdot\mathbf{R}_j} f(\mathbf{r}) \), as its most common use is to relate a function centered in the reference cell with a function centered in some other cell. We note that this formulation of the Bloch theorem is not the only (or even the most common) formulation, but it is the one that is most amenable to our situation.
The vector $\mathbf{k}$ introduced in Eq. (1.52) is known as a wavevector, and it arises as a result of defining another lattice, biorthogonal to the primitive lattice, known as the inverse (or reciprocal) lattice. The inverse lattice is a Fourier transform of the primitive lattice. Whereas points in the primitive lattice were defined in terms of the primitive basis vectors $\{\mathbf{a}_i\}$, points in the inverse lattice will be defined in terms of the biorthogonal basis vectors $\{\mathbf{b}_i\}$. The biorthogonal basis vectors are related to the primitive basis vectors via

$$\mathbf{b}_i \cdot \mathbf{a}_j = 2\pi \delta_{ij} \quad (1.54)$$

and, in analogy to the translation vectors of the primitive lattice, we will define the inverse lattice vectors as

$$\mathbf{K}_j = \sum_{i=1}^{d} g_{ji} \mathbf{b}_i \quad (1.55)$$

Just as we defined the primitive lattice as being made up of all points defined by the translation vectors $\{\mathbf{R}_j\}$, so we will define the inverse lattice as being made up of all points defined by the inverse lattice vectors $\{\mathbf{K}_j\}$. And just as we have chosen to work with a primitive unit cell (the smallest volume unit cell that, when translated throughout space, reproduces the crystal), so we will also consider an analogous concept in the inverse lattice: the first Brillouin zone (FBZ). The FBZ is made up of the wavevectors $\{\mathbf{k}\}$ that label all of the irreducible representations of the translational symmetry group. There are no duplicate $\mathbf{k}$

---

†Here, as was the case with $\mathbf{R}_j$ vs $R_j$, we run into some ambiguity in the terminology. The term “wavenumber” will sometimes be used to refer to $\mathbf{k}$ itself; at other times, “wavenumber” will be used to refer to the index $k$ of the wavevector $\mathbf{k}$. Still another notational option is to refer to the set of $\mathbf{k}$ as $\mathbf{k}$-points. Again, context will generally make clear what is intended.
within the FBZ, and every vector \( \mathbf{k}' \) falling outside of the FBZ can be related to a vector \( \mathbf{k} \) within the FBZ via the relationship\(^{87}\)

\[
\mathbf{k}' = \mathbf{k} + \mathbf{K}_s
\]  

(1.56)

where \( \mathbf{K}_s \) is some inverse lattice vector. Like the primitive unit cell, the FBZ can be translated throughout space to recreate the entire inverse lattice.

We have noted that the wavevectors \( \{ \mathbf{k} \} \) label the irreducible representations of the translational symmetry group, so we can use these vectors to construct periodic basis functions having the same symmetry as the crystal.\(^{87}\) Such periodic basis functions are known as Bloch functions or Bloch orbitals, and they are formed as “translation-symmetry-adapted linear combinations”\(^ {41}\) of the AOs, which are themselves replicated in each of the unit cells:

\[
\phi_{\mu, \mathbf{k}}(\mathbf{r}) = \sum_j e^{i \mathbf{k} \cdot \mathbf{r}_j} \chi_{\mu}(\mathbf{r} - \mathbf{R}_j)
\]  

(1.57)

These periodic basis functions will serve as the foundation of the periodic Hartree-Fock procedure, which we discuss next.

### 1.6.2 Periodic Hartree-Fock

In the molecular context, the Hartree-Fock procedure serves to determine the best single-determinant wavefunction, in a variational sense, from a pre-selected set of one-electron basis functions. In the periodic crystalline case, the goal and overall procedure are the same, but there are a few important implementation differences.
We begin with a set of standard atom-centered Gaussian functions, replicated in each unit cell. These basis functions are then combined via Eq. (1.57) to form the Bloch functions (periodic basis functions), in terms of which the Hartree-Fock equations will be expressed. In the molecular case, the Hartree-Fock wavefunction is formed as a determinant created from the \( N \) lowest-energy MOs, which are themselves linear combinations of the AOs; the coefficients for expanding the MOs in terms of AOs are what are solved for in the Hartree-Fock procedure. Likewise, in the periodic crystalline case, we will solve for the expansion coefficients that will allow us to form crystalline orbitals (COs) from the Bloch functions.

The COs are formed from the Bloch functions via\(^{41}\)

\[
\psi_{\gamma,\mathbf{k}}(\mathbf{r}) = \sum_{\mu} C_{\gamma,\mathbf{k}}^\mu \phi_{\mu,\mathbf{k}}(\mathbf{r})
\]  

(1.58)

It is important to note that only Bloch functions of the same \( \mathbf{k} \) can be combined to form COs.\(^{87}\) The standard HF Roothaan equations are formed by projecting the basis functions throughout the crystal against the basis functions in the reference cell \((R_0)\) and then transforming each quantity to the \( \mathbf{k} \)-space. In the case of the Fock matrix, this looks like\(^{41}\)

\[
F_{\mu,\nu,\mathbf{k}} = \sum_{j} e^{i \mathbf{k} \cdot \mathbf{R}_j} F_{\mu,\nu,\mathbf{R}_j}
\]  

(1.60)

where

\[
|\mu\rangle \equiv |\chi_{\mu}(\mathbf{r} - \mathbf{R}_0)\rangle = |\chi_{\mu}(\mathbf{r})\rangle
\]  

(1.61)

and

\[
|\nu \mathbf{R}_j\rangle \equiv |\chi_{\nu}(\mathbf{r} - \mathbf{R}_j)\rangle
\]  

(1.62)
Since only Bloch functions of the same \( k \) can combine to form COs, we will have \( N_k \) matrix equations that must be solved simultaneously, where each matrix equation is just the HF Roothaan equation for that \( k \):

\[
F_k C_k = S_k C_k \epsilon_k
\]  

(1.63)

The Fock matrix for each \( k \) is kept finite by the fact that we first project onto just the AOs in the reference cell, and this leads to sums over unit cells that naturally decay (albeit at different rates). The number of \( k \) points we must consider is also kept finite by taking advantage of the fact that the properties that depend on the \( k \) points vary smoothly with \( k \) throughout the FBZ.\(^{86,87}\) Thus, we can safely limit ourselves to a finite set of \( k \) points, chosen as a uniform grid from the FBZ,\(^{41}\) with \( N_k \) equal to the number of unit cells in the supercell.

Though an in-depth discussion of the solution of the periodic Hartree-Fock equations is beyond the scope of this work, we would be remiss if we did not highlight a few points. In particular, let us note the definitions of the close-shell Fock matrix and the Hartree-Fock energy, both of which are expressed in terms of the AOs:\(^{41}\)

\[
F_{\mu,\nu R_j} = T_{\mu,\nu R_j} + V_{\mu,\nu R_j} + 2J_{\mu,\nu R_j} - K_{\mu,\nu R_j}
\]  

(1.64)

\[
E^{\text{HF}} = \sum_j \sum_{\mu,\nu} D_{\mu,\nu R_j}(F_{\mu,\nu R_j} + T_{\mu,\nu R_j} + V_{\mu,\nu R_j})
\]  

(1.65)

where the density matrix \( D_{\mu,\nu R_j} \) is given by\(^ {41}\)

\[
D_{\mu,\nu R_j} = \frac{1}{\Omega} \int dke^{i k R_j} \sum_i^{\text{occ}} C_{\mu,i,k}(C_{\nu,i,k}^*)
\]  

(1.66)
In both Eqs. (1.64) and (1.65), the matrices $T$, $V$, $J$, and $K$ have their usual meanings (kinetic energy, nuclear Coulomb potential, electronic Coulomb potential, and exchange), and in Eq. (1.66), $\Omega$ is the volume of the FBZ. The standard procedure is to form an initial estimate of the density matrix $D$, use this to form the constituent elements of $F$ (in the real/direct space, not the reciprocal/k-space), form $F$, transform $F$ to the reciprocal space, solve Eq. (1.63) for each $k$, and then update $D$, continuing until convergence is reached. The $V$, $J$, and $K$ matrices all involve sums over lattice vectors, and details on the truncation of these sums can be found in Ref. 41. In addition, in the particular implementation upon which the work of Chapter 3 is based, cost-reduction is realized by employing density fitting for the approximation of the Coulomb integrals and concentric atomic density fitting (CADF), a form of local DF, for the exchange integrals. Again, more details on this can be found in Ref. 41.

The periodic localization work discussed in Chapter 3 owes an enormous debt of gratitude to the periodic Hartree-Fock implementation described in Ref. 41 and realized in the developmental version of the Massively Parallel Quantum Chemistry (MPQC) package (version 4.0.0).
Chapter 2

Optimized Pair Natural Orbitals for the Coupled Cluster Methods


2.1 Abstract

We present the coupled-cluster singles and doubles method formulated in terms of truncated pair natural orbitals (PNO) that are optimized to minimize the effect of truncation. Compared to the standard ground-state PNO coupled-cluster approaches, in which truncated PNOs derived from first-order Møller-Plesset (MP1) amplitudes are used to compress the
CC wave operator, the iteratively optimized PNOs ("iPNOs") offer moderate improvement for small PNO ranks but rapidly increase their effectiveness for large PNO ranks. The error introduced by PNO truncation in the CCSD energy is reduced by orders of magnitude in the asymptotic regime, with an insignificant increase in PNO ranks. The effect of PNO optimization is particularly effective when combined with Neese’s perturbative correction for the PNO incompleteness of the CCSD energy. The use of the perturbative correction in combination with the PNO optimization procedure seems to produce the most precise approximation to the canonical CCSD energies for small and large PNO ranks. For the standard benchmark set of noncovalent binding energies, remarkable improvements with respect to the standard PNO approach range from a factor of 3 with PNO truncation threshold $\tau_{\text{PNO}} = 10^{-6}$ (with the maximum PNO truncation error in the binding energy of only 0.1 kcal/mol) to more than 2 orders of magnitude with $\tau_{\text{PNO}} = 10^{-9}$.

### 2.2 Introduction

First-principles many-body electronic structure methods can predict many molecular properties, including structure, spectra, thermodynamic data, and chemical reactivities. In particular, the coupled-cluster method can approach near-experimental accuracy for the prediction of the heats of formation of small molecules, with the hope that for heavier elements and larger systems theory will actually become more accurate than experiment. Unfortunately, application of even the simplest many-body methods, such as the coupled-cluster singles and
doubles method (CCSD),\textsuperscript{93} is restricted to small systems due to the $O(N^6)$ computational complexity of its standard LCAO (linear combination of atomic orbitals) implementation. Coupled-cluster methods that are capable of quantitative energetics, such as the coupled-cluster singles, doubles, and perturbative triples method [CCSD(T)],\textsuperscript{35} have an even higher computational complexity [in this particular case, $O(N^7)$]. Thus, significant recent effort has focused on reducing the complexity of many-body methods.

The complexity of many-body methods can be reduced from their naïve figures by several techniques. First, the use of (non-LCAO) numerical representations, e.g., real-space/reciprocal-space grids, that permit fast application of operators can be used to reduce scaling,\textsuperscript{94–97} albeit at the cost of increasing the verbosity of the representation. In the context of LCAO, complexity reduction calls for the use of low-rank representations (e.g. orbital localization, density fitting, and iterative subspace compression) and screening. Recently, by combining such techniques, practical reduced scaling implementations of LCAO coupled-cluster methods, capable of maintaining chemically-acceptable precision and achieving low-order (sometimes linear) scaling with the system size, have been demonstrated\textsuperscript{43,73,98–103}.

A key to recent advances has been the introduction of pair natural orbitals (PNOs)\textsuperscript{42,68} and other closely related concepts for block-wise compression.\textsuperscript{74,78,104} Just as the natural spin orbitals are the optimal basis (in the sense of wave function norm) for the CI expansion of a two-electron system,\textsuperscript{75} so PNOs provide an efficient (albeit not optimal) basis for encoding pair blocks of a wave operator. Although PNOs date back to the 1960s and 1970s and the work of Edmiston and Krauss,\textsuperscript{64,65} Meyer,\textsuperscript{66,67,105} Ahlrichs,\textsuperscript{106} and others, their recent
use was popularized by the work of Neese and co-workers,\textsuperscript{42,68} who showed that they can reduce the scaling and prefactor to a degree sufficient for early crossover with canonical methods. A combination of PNO-style compression with local formulations of coupled-cluster (already shown to be capable of linear scaling by Werner and co-workers\textsuperscript{62,107,108}) gives rise to reduced\textsuperscript{102,103} and even linear scaling\textsuperscript{39,73,78,98–101,109,110} variants of the PNO coupled-cluster methods, which achieve practical supremacy compared to the canonical coupled-cluster implementations for systems with 10-20 atoms.

The use of PNOs in any infinite-order method, such as configuration interaction, coupled-cluster, or Green’s function approaches, is predicated on access to guess two-body amplitudes of sufficient quality to construct accurate PNOs. All modern applications use (approximate) first-order Møller-Plesset (MP1) amplitudes\textsuperscript{42} to form the PNOs (these are often referred to as MP2 PNOs; to avoid confusion about the Møller-Plesset perturbation order of the amplitudes we will use the MP1 nomenclature); other methods of PNO construction have also been investigated.\textsuperscript{66,67} It is clear that such a choice may be suboptimal, such as for cases when correlation can introduce substantial relaxation effects (e.g. in anions) and in small-gap systems in general (conjugated organic molecules, semiconductor crystals). Here we propose to investigate how closely MP1-based PNOs approximate the optimal PNOs in the context of the CCSD. To construct optimal PNOs we have devised an iterative algorithm for refinement of the PNOs; the moniker “iPNO” will be used to distinguish these optimized PNOs from standard PNOs.

Our paper is organized as follows. In Section 2.3, we outline the construction and truncation
of PNOs. We then discuss the current method for iPNO construction in more detail and include a discussion of perturbative corrections for the PNO truncation errors that we investigated in this work. Section 2.5 details numerical performance of iPNO-CCSD vs standard PNO-CCSD and canonical CCSD. We summarize our findings and discuss potential for the use of iPNOs in a production-quality PNO coupled-cluster implementation in Section 2.6.

2.3 Theoretical background

The pair natural orbitals of pair $ij$ are the eigenvectors of the corresponding pair density $D_{ij}$:

$$D_{ij} = U_{ij} n_{ij},$$  \hspace{1cm} (2.1)

where $(U_{ij})_{ba} \equiv U_{ba,ij}$ is the $b$th expansion coefficient of PNO $a_{ij}$, and $(n_{ij})_{ab} \equiv n_{a_{ij}} \delta_{a_{ij}b_{ij}}$ is the associated PNO occupation number. (Following convention, we have used $i, j, \ldots; a, b, \ldots; p, q, \ldots$ for the occupied, virtual, and general orbitals in the Hartree-Fock (HF) basis, respectively.) The pair density matrix is defined by the two-body amplitudes $T_{ij}$:

$$D_{ij} = \frac{1}{1 + \delta_{ij}} \left( \tilde{T}_{ij}^d T_{ij} + \tilde{T}_{ij}^v T_{ij}^d \right),$$  \hspace{1cm} (2.2)

where $(T_{ij})_{ab} \equiv T_{ab}^{ij}$ and $\tilde{T}_{ab}^{ij} = 2T_{ab}^{ij} - T_{ba}^{ij}$. Transforming amplitudes to the full set of PNOs for each pair

$$\tilde{T}_{ij} \equiv U_{ij} T_{ij} U_{ij}^d,$$  \hspace{1cm} (2.3)
does not produce any computational savings but, rather, greatly increases the costs of computing Hamiltonian matrix elements in the LCAO representation, e.g. the order-4 tensor $g_{cd}^{ab} \equiv \langle cd | r_{12}^{-1} | ab \rangle$, traditionally computed at an $O(N^5)$ cost, becomes, in the PNO basis, $g_{ij}^{a_i b_i}$, which requires an $O(N^7)$ effort to compute. Computational savings are realizable if PNOs are truncated to include only those orbitals for which $n_{a_{ij}} \geq \tau_{\text{PNO}}$, where the truncation threshold $\tau_{\text{PNO}}$ is a user-defined model parameter (setting $\tau_{\text{PNO}} = 0$ makes the PNO-based representation exact). For any finite $\tau_{\text{PNO}}$, the number of PNOs per pair is independent of the system size for systems with nonzero gap, so parametrizing the wave operator in terms of the PNO basis amplitudes $(T_{ab}^{ij})_{a_{ij} b_{ij}}$ directly reduces the number of parameters from $O(N^4)$ to $O(N^2)$. One-body amplitudes $T^i$ are similarly expressed in the compressed basis of orbital-specific virtuals (OSVs) $U^i$, here obtained for orbital $i$ as the PNOs of pair $ii$ truncated with threshold $\tau_{\text{PNO}}/100$; this effectively makes the effect of OSV truncation completely negligible relative to that of PNO truncation.

The precision of the PNO representation is determined by the truncation parameter $\tau_{\text{PNO}}$ and the quality of the guess amplitudes used to compute the PNOs. Although a variety of types of guess amplitudes have been used to construct PNOs in the past, in recent work, the PNOs are usually computed from exact or approximate first-order Möller-Plesset (MP1) amplitudes, which, in the basis of canonical Hartree-Fock orbitals, are computed as

$$T_{ab}^{ij} = \frac{g_{ab}^{ij}}{f_i^i + f_j^j - f_a^a - f_b^b}, \quad (2.4)$$

where $f_p^q = \langle p | \hat{f} | q \rangle$ are the matrix elements of the Fock operator. If the occupied orbitals are localized, the amplitudes evaluated via Eq. (2.4) are referred to as semicanonical am-
plitudes. While semicanonical amplitudes are not the exact MP1 amplitudes, they are sufficiently accurate for the purpose of computing PNOs. This approach has also been generalized by Tew and co-workers in the context of explicitly correlated methods.

Here we propose exploring whether it is possible to improve MP1 PNOs in the context of iterative solvers like those in the coupled-cluster method. The idea is to update PNOs periodically using the current values of the CC doubles amplitudes. In cases where the correlation effects are not described well by perturbation theory and the MP1 amplitudes are a poor approximation to the exact CC doubles amplitudes, updating PNOs might produce substantial savings and/or higher accuracy at constant compression rank.

Because the definition of the pair density in Eq. (2.2) includes the amplitudes expressed in the full space of unoccupied orbitals, it would appear that updating PNOs is only possible if guess amplitudes can be periodically computed in the full space by, e.g., computing the residuals of the CC amplitude equations in the full space also. As we discuss later, it should be possible to update PNOs without ever constructing $T_{ij}$ in the full space of unoccupied orbital products. Since our goal here is to assess the performance of the PNOs optimized for the coupled-cluster family of methods, we utilize a canonical CCSD solver rather than a production PNO CCSD solver (preliminary testing of these ideas utilized a pilot PNO-CCD solver). This simulated implementation is an appropriate choice for testing the approach since coupled-cluster residuals in the full space of unoccupied states are directly available and the effect of PNO truncations can be cleanly studied. Specifically, production implementations of PNO-based coupled-cluster usually introduce additional ad hoc approximations to the
residual, which introduces additional (but controlled) errors; because our goal here is to study how PNO optimization affects the PNO truncation error of the coupled-cluster energy, we do not make any additional approximations other than PNO/OSV truncation (i.e., no pairs are neglected, neither PAO domains nor local density fitting domains are introduced, etc.). Note that simulation of a PNO CC solver using a canonical CC solver has been utilized before by Werner and co-workers\textsuperscript{112,113} and recently by us in the context of PNO-EOM-CCSD.\textsuperscript{114}

The iPNO-CCSD solver is described in Algorithm 1. The basic idea is to solve the CCSD amplitude equations in a given fixed PNO subspace (we refer to these iterations as micro-iterations) and iteratively update the subspace by reconstructing the amplitudes in the full space and recomputing the PNOs (these are macroiterations).
Algorithm 1 iPNO-CCSD simulation algorithm

**Input:** micro iteration convergence predicate $C_{\text{micro}}(R^{ij}, R^i)$;

macro iteration convergence predicate $C_{\text{macro}}(R^{ij}, R^i)$;

PNO truncation threshold $\tau_{\text{PNO}}$.

**Output:** Converged doubles $\{T^{ij}\}$ and singles $\{T^i\}$ CCSD amplitudes expressed in the basis of optimized PNOs $U^{ij}$ and OSVs $U^i$;

1: $k = 0$;

2: $\{T^i_{(0)}\} = 0$;

3: Initialize $\{T^{ij}_{(0)}\}$ to semicanonical MP1 amplitudes using Eq. (2.4);

4: Construct pair densities $\{D^{ij}\}$ as well as truncated PNOs, $\{U^{ij}\}$, and OSVs, $\{U^j\}$ using Eqs. (2.2) and (2.1);

5: repeat

6: repeat

7: Compute CCSD residuals $\{R^{ij}\}$ and $\{R^i\}$ using amplitudes $\{T^{ij}_{(k)}\}$ and $\{T^i_{(k)}\}$;

8: Transform 2-body residual to PNO basis: $\tilde{R}^{ij} = U^{ij\dagger}R^{ij}U^{ij}, \forall ij$;

9: Transform 1-body residual to OSV basis: $\tilde{R}^i = U^{i\dagger}R^iU^i, \forall i$;

10: Use $\{\tilde{R}^{ij}\}$ and $\{\tilde{R}^i\}$ to compute Jacobi/DIIS updates for 2-body PNO basis amplitudes, $\{\tilde{\Delta}^{ij}\}$, and 1-body OSV basis amplitudes, $\{\tilde{\Delta}^i\}$;
Algorithm 1 iPNO-CCSD simulation algorithm, cont.

11: Back transform $\Delta^{ij}$ and $\Delta^i$ to the full unoccupied space:

12: $\bar{\Delta}_{ij} = U^{ij} \Delta^{ij} U^{ij\dagger}, \forall ij$;\(^a\)

13: $\bar{\Delta}^i = U^i \Delta^i U^{i\dagger}, \forall i$;\(^b\)

14:

15: Update the amplitudes:

16: $T^{ij}_{(k+1)} = T^{ij}_{(k)} + \Delta^{ij}, \forall ij$;

17: $T^i_{(k+1)} = T^i_{(k)} + \Delta^i, \forall i$;\(^c\)

18: $k \leftarrow k + 1$;

19: until not $C_{\text{micro}}(\{R^{ij}\}, \{R^i\})$

20: Compute CCSD residuals $\{R^{ij}\}$ and $\{R^i\}$ using amplitudes $\{T^{ij}_{(k)}\}$ and $\{T^i_{(k)}\}$;

21: Use $\{R^{ij}\}$ and $\{R^i\}$ to compute Jacobi/DIIS updates for 2-body amplitudes, $\{\Delta^{ij}\}$, and 1-body amplitudes, $\{\Delta^i\}$;

22: Compute amplitudes for the PNO update:

23: $\hat{T}^{ij}_{(k)} = T^{ij}_{(k)} + \Delta^{ij}, \forall ij$;

24: $\hat{T}^i_{(k)} = T^i_{(k)} + \Delta^i, \forall i$;

25: Construct updated truncated PNOs, $\{U^{ij}\}$, and OSVs, $\{U^i\}$, using $\{\hat{T}^{ij}\}$;

\(^a\)There is a typo in the published work. This line should be $\Delta^{ij} = U^{ij} \Delta^{ij} U^{ij\dagger}, \forall ij$;

\(^b\)There is a typo in the published work. This line should be $\Delta^i = U^i \Delta^i U^{i\dagger}, \forall i$;

\(^c\)There is a typo in the published work. This line should be $T^{i}_{(k+1)} = T^i_{(k)} + \Delta^i, \forall i$;
Algorithm 1 iPNO-CCSD simulation algorithm, cont.

26: Project amplitudes \( \{ T_{ij}^{(k)} \} \) and \( \{ T_{i}^{(k)} \} \) to the updated PNO and OSV subspaces, respectively:

27: \( T_{ij}^{(k+1)} \leftarrow U_{ij} U_{ij}^{\dagger} T_{ij}^{(k)} U_{ij} U_{ij}^{\dagger}, \forall ij \),

28: \( T_{i}^{(k+1)} \leftarrow U_{i} U_{i}^{\dagger} T_{i}^{(k)} U_{i} U_{i}^{\dagger}, \forall i \);

29: \( k \leftarrow k + 1; \)

30: until not \( C_{\text{macro}}(\{ R_{ij} \}, \{ R_{i} \}) \)

2.4 Computational details

The iPNO-CCSD approach was implemented in a developmental version of the Massively Parallel Quantum Chemistry package (version 4).\(^{115}\) Initial assessment of the iPNO-CCSD approach utilized a representative 12-system subset (see Table 2.1) of the S66 data set,\(^{116}\) the geometries of which were taken from the Benchmark Energy and Geometry Database (BEGDB).\(^{117}\) The full S66 data set was used in the final comparison of iPNO-CCSD with standard PNO-CCSD. For most calculations, the cc-pVDZ-F12 basis set\(^{118}\) was employed, with all two-electron integrals approximated by density fitting in the aug-cc-pVDZ-RI basis; computations on BH76 and DARC benchmark sets utilized the aug-cc-pVDZ basis with the aug-cc-pVDZ-RI density fitting basis.\(^{119}\) Valence occupied orbitals were localized via the standard Foster-Boys scheme.\(^{55}\)
Table 2.1: Twelve-System Subset of the S66 Benchmark Data Set Utilized for Initial Assessment of the iPNO-CCSD Approach

<table>
<thead>
<tr>
<th>Index</th>
<th>Index in S66</th>
<th>System</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>water ... water</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>water ... MeOH</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>water ... MeNH₂</td>
</tr>
<tr>
<td>4</td>
<td>9</td>
<td>MeNH₂ ... MeOH</td>
</tr>
<tr>
<td>5</td>
<td>24</td>
<td>benzene ... benzene (π-π)</td>
</tr>
<tr>
<td>6</td>
<td>25</td>
<td>pyridine ... pyridine (π-π)</td>
</tr>
<tr>
<td>7</td>
<td>26</td>
<td>uracil ... uracil (π-π)</td>
</tr>
<tr>
<td>8</td>
<td>34</td>
<td>pentane ... pentane</td>
</tr>
<tr>
<td>9</td>
<td>47</td>
<td>benzene ... benzene (TS)</td>
</tr>
<tr>
<td>10</td>
<td>50</td>
<td>benzene ... ethyne (CH-π)</td>
</tr>
<tr>
<td>11</td>
<td>59</td>
<td>ethyne ... water (CH-O)</td>
</tr>
<tr>
<td>12</td>
<td>66</td>
<td>MeNH₂ ... pyridine</td>
</tr>
</tbody>
</table>
2.5 Results and discussion

2.5.1 PNO-CCSD vs. iPNO-CCSD

First we compare the performance of the PNO and iPNO methods for the absolute correlation energies of dimers. Figure 2.1 illustrates the max and mean absolute percent error in dimer absolute energy, relative to the canonical CCSD energy, as a function of $\tau_{\text{PNO}}$. It is clear that the iPNO-CCSD approach performs consistently better than the PNO-CCSD scheme; a modest average improvement of a factor of 1.9 at $\tau_{\text{PNO}} = 10^{-7}$ (the value used in routine application of PNO methods) becomes an improvement of more than an order of magnitude for $\tau_{\text{PNO}} < 10^{-10}$. Figure 2.2 illustrates this improvement in more detail.

![Graphs showing max and mean absolute error](image)

(a) Maximum absolute error (MAX)  
(b) Mean absolute error (MAE)

Figure 2.1: Statistical analysis of dimer absolute correlation energy percent errors.

The smaller truncation errors of the iPNO-CCSD correlation energies relative to their standard PNO-CCSD counterparts do not come at the cost of increased PNO ranks, as is demon-
Figure 2.2: Ratio of the PNO-CCSD dimer correlation energy percent error to the iPNO-CCSD dimer correlation energy percent error for all systems.

Of course, in chemistry, we are usually interested in differences of correlation energies. The performance of iPNO-CCSD for the binding energies of the dimers studied is compared to that of PNO-CCSD in Figure 2.4. It appears that the improved performance of iPNO-CCSD for the absolute correlation energies translates into improved performance for the binding energies as well.

2.5.2 Perturbative Energy Correction for PNO Incompleteness

As the preceding data indicate, PNO optimization decreases the PNO truncation error in the CCSD energy; in other words, for the same rank, the iPNO-CCSD energy is closer to the canonical CCSD energy than its standard PNO-CCSD counterpart. An interesting follow-up

\footnote{This is a typo in the published work. Clearly, the word should be “incompleteness.”}
question is whether the reduction in the PNO truncation error can be achieved in another way. Neese and co-workers proposed a perturbative correction for the PNO truncation,\textsuperscript{42,68} obtained as the difference between the (semicanonical) MP2 and PNO-MP2 energies, both easily available in the course of computing MP1 PNOs:

\[
\Delta_{\text{PNO-MP2}} \equiv E_{\text{MP2}} - E_{\text{PNO-MP2}}, \quad \text{where} \quad (2.5)
\]

\[
E_{\text{MP2}} = \sum_{ij} (2g^{ij}_{ab} - g^{ij}_{ba}) t_{ab}^{ij}, \quad \text{(2.6)}
\]

\[
E_{\text{PNO-MP2}} = \sum_{a_i b_i} \left( 2g^{ij}_{a_i b_i} - g^{ij}_{b_i a_i} \right) t_{a_i b_i}^{ij}, \quad \text{(2.7)}
\]
and where both standard and PNO unoccupied orbitals are assumed to be canonical, i.e. the Fock operator is diagonal in these spaces. An improved estimate of the canonical CCSD energy is then obtained as

\[
E_{\text{CCSD}} \approx E_{\text{PNO-CCSD}} + \Delta_{\text{PNO-MP2}}
\]  

(2.8)

If the PNO incompleteness errors of the PNO-MP2 and PNO-CCSD energies were identical, this correction would be exact; thus, the key assumption of this scheme is that the PNO incompleteness is not sensitive to the level of correlation treatment. It seems to be a good assumption in practice: Neese and co-workers observed\(^{42,68}\) that for the practical values of \(\tau_{\text{PNO}}\) the use of the perturbative correction significantly reduces the PNO incompleteness error.

Thus, we decided to investigate whether the observed reduction in the PNO incompleteness of the correlation energy due to the optimization of PNOs is accounted for by the \(\Delta_{\text{PNO-MP2}}\)
correction. Since the change in the PNO basis should be accommodated by the incompleteness correction, by analogy with $\Delta_{\text{PNO-MP2}}$, we proposed the use of the following correction for iPNO-CCSD energies:

$$\Delta_{\text{iPNO-MP2}} \equiv E_{\text{MP2}} - E_{\text{iPNO-MP2}}$$  \hspace{1cm} (2.9)

where $E_{\text{iPNO-MP2}}$ is evaluated exactly as $E_{\text{PNO-MP2}}$ but uses optimized CCSD PNOs as the basis. All quantities needed to compute semicanonical $E_{\text{PNO-MP2}}$ and $E_{\text{iPNO-MP2}}$ are readily available in the iPNO-CCSD code, and the implementation is straightforward.

The max and mean absolute errors of the PNO-CCSD and iPNO-CCSD correlation energies of the dimers relative to their canonical CCSD counterparts (denoted by $\Delta_{\text{PNO-CCSD}}$ and $\Delta_{\text{iPNO-CCSD}}$, respectively) are compared to the $\Delta_{\text{PNO-MP2}}$ and $\Delta_{\text{iPNO-MP2}}$ corrections in Figure 2.5, with the corresponding data for binding energies shown in Figure 2.6. It is clear that with coarsely truncated PNOs (large $\tau_{\text{PNO}}$) $\Delta_{\text{PNO-MP2}}$ is nearly indistinguishable from the error in PNO-CCSD, and hence, it effectively corrects for the PNO incompleteness error, as observed by Neese et al. However, as $\tau_{\text{PNO}}$ decreases, the quality of the correction decreases and already at $\tau_{\text{PNO}} = 10^{-8}$, $\Delta_{\text{PNO-MP2}}$ becomes ineffective.

Another observation is that the proposed $\Delta_{\text{iPNO-MP2}}$ correction is not effective for correcting iPNO-CCSD energies; for $\tau_{\text{PNO}} < 10^{-7}$, $\Delta_{\text{iPNO-MP2}}$ significantly overcorrects $E_{\text{iPNO-CCSD}}$. The most interesting suggestion drawn from the data in Figures 2.5 and 2.6 is that $\Delta_{\text{PNO-MP2}}$ seems to be an ideal PNO truncation correction for $E_{\text{iPNO-CCSD}}$ for all values of $\tau_{\text{PNO}}$. This observation is seemingly counterintuitive, since $\Delta_{\text{PNO-MP2}}$ is computed using MP1 PNOs,
whereas $E_{\text{iPNO-CCSD}}$ uses relaxed CCSD PNOs. While an in-depth investigation of this effect is outside the scope of this work, a possible line of inquiry to explain this observation goes as follows. The PNO truncation error in standard PNO-CCSD is driven by two effects: the error due to the use of suboptimal (i.e., MP1) PNOs and that due to the fixed rank error. It is clear that the use of MP1 PNOs in PNO-CCSD results in substantial errors due to the suboptimal PNOs; for high PNO ranks (small $\tau_{\text{PNO}}$), the PNO truncation error of standard PNO-CCSD is entirely dominated by the suboptimality of PNOs, as evidenced by the massive reduction of the PNO incompleteness by PNO optimization (compare iPNO-CCSD and PNO-CCSD errors). The use of optimal PNOs, i.e., MP1 PNOs for PNO-MP2 and CCSD PNOs for iPNO-CCSD, minimizes the errors in respective correlation energies for fixed PNO ranks and thus eliminates one component of the error. It seems that the second
contribution to the error, due to the fixed PNO rank, is almost identical for MP2 and CCSD energies; further investigation of this phenomenon is left to future studies.

Thus, the best way to correct the PNO incompleteness of iPNO-CCSD seems to be via $\Delta_{\text{PNO-MP2}}$. Figures 2.7 and 2.8 compare the performance of PNO-CCSD and iPNO-CCSD with their counterparts corrected with $\Delta_{\text{PNO-MP2}}$ for correlation energies and binding energies of dimers. For coarse truncations ($\tau_{\text{PNO}} \geq 10^{-7}$), corrected PNO-CCSD energies are more precise than bare (uncorrected) iPNO-CCSD energies. However, for tighter truncations ($\tau_{\text{PNO}} \leq 10^{-8}$), even uncorrected iPNO-CCSD outperforms corrected PNO-CCSD; this clearly suggests the dominant effect of suboptimal PNOs in the residual error of PNO-CCSD energy in the asymptotic regime ($\tau_{\text{PNO}} \to 0$). Most importantly, note that for $\tau_{\text{PNO}} \leq 10^{-6}$, corrected iPNO-CCSD energies are more precise than both corrected PNO-CCSD and un-
corrected iPNO-CCSD. In fact, the performance of corrected iPNO-CCSD is rather remarkable, e.g., max error in binding energy of less than 0.1 kcal/mol is obtained already with $\tau_{\text{PNO}} = 10^{-6}$!

\[
\begin{align*}
\text{(a) Maximum absolute error (MAX)} & \quad \text{(b) Mean absolute error (MAE)} \\
\end{align*}
\]

Figure 2.7: Comparison of the accuracy of the various corrected and uncorrected schemes for dimer correlation energies.

### 2.5.3 Broader Assessment

As a further illustration of the performance of our corrected iPNO-CCSD scheme, we used our corrected and uncorrected PNO-CCSD and iPNO-CCSD schemes to compute the binding energies of all 66 dimers in the S66 dataset, to compute the Diels-Alder reaction energies of a subset of the DARC dataset$^{120,121}$, and to compute the reaction barriers of a subset of the BH76 dataset$^{120,122,123}$ (error statistics are presented in Table 2.2). We also compared the performance of the PNO and iPNO approaches for a prototypical strongly correlated
system, a 10-atom hydrogen chain (Table 2.3). In all cases, $\tau_{\text{PNO}} = 10^{-6}$ was used.

Of the 66 dimers in the S66 data set, not a single one has an iPNO-CCSD $+$ $\Delta_{\text{PNO-MP2}}$ error above 0.1 kcal/mol, while the mean absolute error for this scheme is more than an order of magnitude smaller than the corresponding values for uncorrected PNO-CCSD and iPNO-CCSD. In addition, the mean absolute error is reduced relative to the standard PNO-CCSD $+$ $\Delta_{\text{PNO-MP2}}$ approach by a factor of 3.

The DARC dataset, first proposed by Johnson et al.\textsuperscript{121} and later given the moniker “DARC” by Goerigk and Grimme,\textsuperscript{120} comprises a number of Diels-Alder reactions. From these reactions, a subset consisting of reactions 1-7, 9, 11, and 13 was chosen, and their reaction energies and errors (relative to the canonical CCSD result) were computed. The general
trend for these errors, as can be seen in Table 2.2, is very similar to that of the S66 binding
energy errors, with the main differences being that the iPNO-CCSD + ∆_{iPNO−MP2} approach
actually outperforms the PNO-CCSD + ∆_{PNO−MP2} approach and the improvement of the
iPNO-CCSD + ∆_{PNO−MP2} approach over the iPNO-CCSD + ∆_{iPNO−MP2} approach is much
less pronounced.

We also considered a subset of reaction barriers from the BH76 test set, proposed by Goerigk
and Grimme\textsuperscript{120} and based on earlier work by Truhlar and coworkers\textsuperscript{122,123}. With the excep-
tion of reaction 38, for which the optimized geometry files were not available, all closed-shell
reactions (8, 10, 12 (forward), 12 (reverse), 14 (forward), 14 (reverse), 19 (forward), and 19
(reverse)) from this set were considered. The summary of the errors in the barrier heights
can, as before, be seen in Table 2.2. The relative ordering of the five different schemes differs
even more from that observed with S66 than the DARC ordering does, though the largest
errors still come from the PNO-CCSD method. The iPNO-CCSD + ∆_{PNO−MP2} method is
actually ever so slightly outperformed by the iPNO-CCSD + ∆_{iPNO−MP2} method, but both
schemes have maximum absolute (MAX) errors 2-5 times smaller than the MAX error in the
uncorrected iPNO-CCSD scheme.

Last, we assessed the iPNO-CCSD approach for the description of a chain of 10 hydrogen
atoms. This is a prototypical system that undergoes transition from the weak to the strong
correlation regime as the bond distance increases. Recently, this system was used for exten-
sive benchmarking of a number of established and emerging electronic structure methods,
including CCSD,\textsuperscript{124} with the reference equilibrium HH distance of \( R = 1.801a_0 \). One expects
Table 2.2: Statistical Analysis of the S66 Binding Energy Errors, the DARC Reaction Energy Errors, and the BH76 Reaction Barrier Errors

<table>
<thead>
<tr>
<th>Method</th>
<th>MAX</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PNO-CCSD</td>
<td>4.22</td>
<td>0.59</td>
</tr>
<tr>
<td>iPNO-CCSD</td>
<td>3.96</td>
<td>0.51</td>
</tr>
<tr>
<td>PNO-CCSD + Δ_{PNO-MP2}</td>
<td>0.24</td>
<td>0.09</td>
</tr>
<tr>
<td>iPNO-CCSD + Δ_{PNO-MP2}</td>
<td>0.09</td>
<td>0.03</td>
</tr>
<tr>
<td>iPNO-CCSD + Δ_{iPNO-MP2}</td>
<td>0.76</td>
<td>0.11</td>
</tr>
<tr>
<td>PNO-CCSD</td>
<td>2.13</td>
<td>1.67</td>
</tr>
<tr>
<td>iPNO-CCSD</td>
<td>1.85</td>
<td>1.33</td>
</tr>
<tr>
<td>PNO-CCSD + Δ_{PNO-MP2}</td>
<td>0.65</td>
<td>0.48</td>
</tr>
<tr>
<td>iPNO-CCSD + Δ_{PNO-MP2}</td>
<td>0.25</td>
<td>0.12</td>
</tr>
<tr>
<td>iPNO-CCSD + Δ_{iPNO-MP2}</td>
<td>0.27</td>
<td>0.16</td>
</tr>
<tr>
<td>PNO-CCSD</td>
<td>0.44</td>
<td>0.28</td>
</tr>
<tr>
<td>iPNO-CCSD</td>
<td>0.17</td>
<td>0.10</td>
</tr>
<tr>
<td>PNO-CCSD + Δ_{PNO-MP2}</td>
<td>0.25</td>
<td>0.17</td>
</tr>
<tr>
<td>iPNO-CCSD + Δ_{PNO-MP2}</td>
<td>0.07</td>
<td>0.03</td>
</tr>
<tr>
<td>iPNO-CCSD + Δ_{iPNO-MP2}</td>
<td>0.03</td>
<td>0.02</td>
</tr>
</tbody>
</table>

In all cases, \( \tau_{PNO} = 10^{-6} \) was used. All values are in kcal/mol.
Table 2.3: PNO Truncation Errors in CCSD cc-pVTZ Energy of the Equidistant H\textsubscript{10} Chain

<table>
<thead>
<tr>
<th>Method</th>
<th>$R = 1.8a_0^a$</th>
<th>$R = 3.2a_0^b$</th>
</tr>
</thead>
<tbody>
<tr>
<td>PNO-CCSD</td>
<td>0.744</td>
<td>2.271</td>
</tr>
<tr>
<td>iPNO-CCSD</td>
<td>0.503</td>
<td>0.129</td>
</tr>
<tr>
<td>PNO-CCSD + $\Delta\text{PNO-MP2}$</td>
<td>0.033</td>
<td>1.934</td>
</tr>
<tr>
<td>iPNO-CCSD + $\Delta\text{iPNO-MP2}$</td>
<td>$-0.207$</td>
<td>$-0.209$</td>
</tr>
<tr>
<td>iPNO-CCSD + $\Delta\text{iPNO-MP2}$</td>
<td>$-0.377$</td>
<td>$-0.846$</td>
</tr>
</tbody>
</table>

\(^a E_{\text{CCSD}} = -249.598mE_h^b E_{\text{CCSD}} = -367.280mE_h^b\)

that as the HH distance is increased and the system enters the strong correlation regime the
MP1 amplitudes will become an increasingly poor approximation to the CCSD amplitudes
and thus the differences between the PNO and iPNO approaches will increase. Indeed, as the
data in Table 2.3 indicates, near equilibrium the truncation errors of PNO-CCSD and iPNO-
CCSD are comparable, but at the stretched geometry the truncation error of iPNO-CCSD is
much smaller (by a factor of 20) than that of PNO-CCSD. Also note that the perturbative

correction for PNO incompleteness becomes less useful in the strong correlation regime.

2.6 Summary and Perspective

We have investigated the use of iteratively-optimized PNOs ("iPNOs") for a more robust
compression of the coupled-cluster wave operator in the context of CCSD. The performance is
compared to that of the standard PNO approach in which MP1 PNOs are used to compress the CC wave operator. PNO optimization offers moderate improvement relative to the standard PNO-CCSD for small PNO ranks but rapidly increases in effectiveness for large PNO ranks; the PNO incompleteness error of the CCSD energy is reduced by orders of magnitude in the asymptotic regime, with an insignificant increase in PNO ranks. This suggests that in applications that call for a precise representation of the wave operator, such as nonresonant response, the use of iPNO-CCSD may be readily warranted.

The effect of PNO optimization is not accounted for by Neese’s perturbative correction for the PNO incompleteness of the CCSD energy. In fact, in the asymptotic regime, the correction becomes ineffective, and the PNO truncation error of PNO-CCSD is entirely dominated by the use of suboptimal (MP1) PNOs. The use of the perturbative correction in combination with the PNO optimization procedure seems to produce the most precise approximation to the canonical CCSD result for small and large PNO ranks; remarkable improvements with respect to the standard PNO approach range from a factor of 3 with $\tau_{\text{PNO}} = 10^{-6}$ to more than 2 orders of magnitude with $\tau_{\text{PNO}} = 10^{-9}$. Thus, the use of the iPNO approach seems warranted even for computing energies. The remarkable performance of the perturbatively-corrected iPNO approach for the energies suggests that further investigation of how to correct the PNO incompleteness of the coupled-cluster wave operator (not just the energy) is worthwhile.

While promising, it is too early to tell whether the PNO optimization will be viable for mainstream applications due to the need to recompute integrals in the PNO basis each time
PNOs are updated. Production PNO-CCSD implementations reduce the cost of integral transformation by the use of (local) density fitting in conjunction with the use of OSVs or PAO domains to reduce the size of orbital-specific unoccupied space. Thus, PNO updates will require only the last stage of the transformation, from OSVs/PAOs to PNOs, to be repeated; this last step is usually less expensive than the preceding three-index integral transformation in the asymptotic regime (three-dimensional systems with hundreds of atoms). As an experiment we performed a DLPNO-CCSD computation with the ORCA program on a cluster of 20 waters in the aug-cc-pVTZ basis. We assumed that PNO optimization will allow us to set \( \tau_{\text{PNO}} = 10^{-6} \) instead of \( \tau_{\text{PNO}} = 10^{-8} \) with the conventional PNO approach; with the standard DLPNO approach, the former computation (using, on average, 13 PNOs/pair) took 4.5 h on 8 cores, whereas the latter (averaging 44 PNOs/pair) took more than 9 h. The bulk of the difference in the execution time was due to the evaluation of the CCSD residual (the more precise computation took more than 10 times longer per iteration); the differences in all other steps were minor. Most notably, the cost of PNO transformations did not differ significantly (5,300 vs 5,600 s) despite the significant differences in the number of PNOs, likely due to the significant overhead of disk I/O and memory transfers. Because, on average, 4-5 PNO updates are usually sufficient to converge to a few microhartrees even with our simplistic solver, it appears that the extra costs of the PNO transformation are unlikely to pay off with the current implementation of the DLPNO method. However, the efficiency of the PNO transformation may be improvable by batching or avoiding disk whatsoever. Furthermore, if properties other than the energy are required it may be necessary to
use even higher truncation parameters; hence, the iPNO approach may become viable even without additional improvements of the PNO transformation.

Note that the current pilot implementation of the PNO optimization procedure also has much room for improvement. There is, for example, no reuse of residuals from previous macro iterations (i.e., DIIS extrapolation is only performed within the current subspace). Second, we utilize the CCSD residual, which has two indices in the full space of unoccupied states (in the case of PAO or OSV-based methods, the residual would be evaluated in the full PAO/pair-OSV space of each pair). This should not be necessary and can be eliminated by switching to the gradient-based optimization where the coupled-cluster energy is written as a functional of PNOs and the PNO optimization utilizes the energy derivative with respect to the PNOs. Gradient-based optimization of PNOs is already used by one of us (E.F.V.) in the context of real-space correlation methods and seems to be a robust way to compute optimal PNOs without the need to represent the wave operator explicitly. The advantage of the gradient-based approach is that only 1 index of the gradient is expressed in the full space of unoccupied orbitals for each pair. Last, it may be possible to avoid PNO optimization altogether by incorporating the PNO gradient into the coupled-cluster Lagrangian and solving for the corresponding multipliers.
Chapter 3

Robust Pipek-Mezey Orbital Localization in Periodic Solids

Reprinted in part with permission from Clement, M. C.; Wang, X.; Valeev, E. F. 2021, 

3.1 Abstract

We describe a robust method for determining Pipek-Mezy (PM) Wannier functions (WF), 
recently introduced by Jónsson et al. (J. Chem. Theor. Chem. 2017, 13, 460), which provide 
some formal advantages over the more common Boys (also known as maximally-localized) 
Wannier functions. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) based PMWF solver is 
demonstrated to yield dramatically faster convergence compared to the alternatives (steepest
ascent and conjugate gradient) in a variety of 1-, 2-, and 3-dimensional solids (including some with vanishing gaps), and can be used to obtain Wannier functions robustly in supercells with thousands of atoms. Evaluation of the PM functional and its gradient in periodic LCAO representation used a particularly simple definition of atomic charges obtained by Moore-Penrose pseudoinverse projection onto the minimal atomic orbital basis. An automated “Canonicalize Phase then Randomize” (CPR) method for generating the initial guess for WFs contributes significantly to the robustness of the solver.

3.2 Introduction

Localized orbitals, by eliminating the artifacts of symmetry and accidental degeneracy, are valuable for qualitative interpretation of electronic states in terms of traditional concepts of chemical bonding and as a computational basis underpinning many reduced complexity algorithms in electronic structure. Localized orbitals are particularly relevant for periodic solids, where due to the lattice symmetry, the eigenstates of observables are delocalized over the entire lattice; such delocalization becomes counterproductive when the unit cell size significantly exceeds the lengthscale of the decay of the 1-particle reduced density matrix or when the focus is on localized features of the electronic structure (e.g., impurities and surface adsorbates).

For molecules, the most commonly used black-box localization methods are those due to Foster and Boys (FB)\textsuperscript{55,126} and Pipek and Mezey (PM);\textsuperscript{127} several historically-important meth-
ods like Edmiston-Ruedenberg (ER)\textsuperscript{15} and von Niessen (vN)\textsuperscript{128} are rarely used nowadays. All of these orbitals are defined as the stationary points of the corresponding functional. For example, the FB orbitals minimize the sum of squared position uncertainties of the orbitals. Pipek-Mezey orbitals maximize the sum of squares of atomic charges of each orbital; the original PM definition utilized Mulliken charges, which are meaningless for non-minimal basis sets, and other definitions of atomic charges are far more robust.\textsuperscript{129–133} Unlike the FB orbitals and like the ER and vN orbitals, the PM orbitals preserve the $\sigma - \pi$ separation, which is an important advantage when interpreting the electronic structure.\textsuperscript{†} Further enhancements of these functionals include the use of higher-than-second power analogs of the FB\textsuperscript{134} and PM\textsuperscript{131,132} functionals.

In periodic solids, the analogs of molecular localized orbitals are referred to as generalized Wannier functions (WF),\textsuperscript{135} which, in contrast to conventional Wannier functions\textsuperscript{136} obtained from a single band, mix Bloch orbitals from several bands.\textsuperscript{137} Marzari and Vanderbilt championed the use of the FB functional to determine the generalized Wannier functions, which they dubbed as maximally-localized (generalized) Wannier functions (MLWF).\textsuperscript{135} Since the implementation of an MLWF solver in the \texttt{Wannier90} package\textsuperscript{138} the use of MLWF has become popular\textsuperscript{137} for interpreting the electronic states and as a basis in a number of reduced-scaling many-body electronic structure methods.\textsuperscript{139,140} Note that the MLWF formalism is not unique; for example, the FB localization of periodic orbitals by Zicovich-Wilson et al.,\textsuperscript{141} as implemented in the \texttt{Crystal} package,\textsuperscript{142} in contrast to the approach of Marzari and Vander-

\textsuperscript{†}However, unlike the ER and vN orbitals, the PM orbitals lack the intra-atomic localization.
bilt, uses a purely real-space formulation. Furthermore, the choice of localization functional itself is not unique; MLWFs are just one of many plausible types of “maximally-localized” generalized Wannier functions.

Recently, Jónsson et al. introduced the use of Pipek-Mezey Wannier functions (PMWF), which, in contrast to the FB-based WFs of Refs. 135 and 141 and analogously to the molecular PM counterparts, do not mix the $\sigma$ and $\pi$ orbitals. Jónsson et al. evaluated PMWFs using charges defined via real-space partitioning; the atomic charges and PMWFs were found to be insensitive to the specifics of the atomic partitioning, just as molecular PM orbitals were found to depend weakly on the choice of atomic charges used in the PM functional. The maximum of the PM functional was located by the conjugate gradient (CG) method, which for PM optimization in molecules was found to be superior than the steepest ascent (SA) method. Unfortunately, the convergence of the CG solver for PMWFs, as documented by Jónsson et al. as well as revealed in our own experiments, can be fairly poor, requiring hundreds or even thousands of iterations. Also note that the CG is used to construct MLWFs in Wannier90.

As we discovered in our work, the performance of the SA and (nonlinear) CG solvers for PMWFs can be greatly improved by the Broyden-Fletcher-Goldfarb-Shanno (BFGS) solver. While BFGS as well as other quasi-Newton methods have been used to solve for localized orbitals in molecules, to our knowledge its use for Wannier function optimization has not been considered. Thus the main purpose of this manuscript is to document the implementation of the BFGS PMWF solver and compare its performance to that of SA
and CG. Here we also document the particularly simple definition of atomic charges that we devised via pseudoinverse projection on the minimal basis, which is very convenient when employing the linear combination of atomic orbitals (LCAO) representation of the periodic solid’s orbitals. The result of this work is a robust periodic localizer that has been successfully applied to one-, two-, and three-dimensional systems with large Born-von-Kármán unit cells, including some with vanishingly small band gaps.

3.3 Formalism

3.3.1 Basic Definitions

The objective of PM localization is to convert an input set of periodic (Bloch) orbitals (usually, Hartree-Fock or Kohn-Sham orbitals) to a set of localized orbitals. In our work, the orbitals are expressed in the LCAO representation, expanded in (contracted) Gaussian AOs, obtained by the reduced-scaling Hartree-Fock method recently reported by some of us.\textsuperscript{41}

The $j$th Bloch orbital $|\psi_{j,k}\rangle$ with crystal momentum wave vector $\mathbf{k}$ is a linear combination of Bloch AOs $\{\phi_{\nu,k}\}$:

$$|\psi_{j,k}\rangle = \sum_{\nu} C_{\nu,k}^j |\phi_{\nu,k}\rangle .$$  \hspace{1cm} (3.1)

Bloch AOs in turn are translation-symmetry-adapted linear combinations of AOs $\{\phi_{\nu,R}\}$:

$$|\phi_{\nu,k}\rangle = \frac{1}{\sqrt{N}} \sum_{\mathbf{R}} e^{i\mathbf{k}\cdot\mathbf{R}} |\phi_{\nu,R}\rangle ,$$  \hspace{1cm} (3.2)
with \( \mathbf{R} \) denoting the origin of each primitive unit cell in the Born-von-Kármán (BvK) unit cell ("supercell") composed of \( N \) primitive cells; by convention, \( \mathbf{R} = 0 \) corresponds to the reference unit cell. Please note the different normalization of Bloch AOs in Eq. (3.2) than is traditional in the periodic LCAO literature.\(^{41,152,153}\)

A generalized Wannier function centered in the unit cell at \( \mathbf{R} \) will be expressed as a linear combination of Bloch orbitals:

\[
\left| \tilde{\psi}_{i, \mathbf{R}} \right> = \frac{1}{\sqrt{N_k}} \sum_{j, \mathbf{k}} e^{-i \mathbf{k} \cdot \mathbf{R}} \left| \psi_{j, \mathbf{k}} \right> U_{i, j, \mathbf{k}},
\]

(3.3)

where \( i \) indexes the Wannier functions (localizing \( o \) Bloch orbitals will produce \( o \) WFs) and \( N_k \) is the number of \( \mathbf{k} \) points in the uniform (Monkhorst-Pack)\(^{154}\) quadrature used to integrate the first Brillouin zone corresponding to the supercell in Eq. (3.2) (in this work we impose \( N_k = N \)). Matrices \( \{ \mathbf{U}_\mathbf{k} \} \), defined as \( (\mathbf{U}_\mathbf{k})_j^i \equiv U_{i, j, \mathbf{k}}, \)\(^{1}\) are unitary to ensure mutual orthonormality of WFs, both associated with the same unit cell and between WFs associated with different unit cells; this also ensures that WFs span the same space as the Bloch orbitals.

No additional constraints, such as realness of the WFs, are imposed.

### 3.3.2 PM Functional with Pseudoinverse Minimal Basis Atomic Charges

The PM WFs are stationary points of the PM functional:

\[
P \equiv \sum_{\mathbf{R}} \sum_{\mathbf{A}} \sum_i |Q_{i}^{A \mathbf{R}}|^p
\]

(3.4)

\(^{1}\)The superscript and subscript indices on matrix elements refer to the columns and rows, respectively.
where $A$ indexes the atoms in the primitive unit cell, and $i$ indexes the Wannier functions. The atomic charge contribution $Q_{i}^{AR}$ is the charge associated with orbital $i$ on atom $A$ in unit cell $\mathbf{R}$, and the charge exponent, $p$, is equal to 2 (conventional PM functional) or 4 (fourth-order PM functional$^{131,132}$).

The original work by Pipek and Mezey used standard Mulliken charges,$^{127}$

$$Q_{i}^{AR} = \langle \bar{\psi}_{i,0} | \hat{P}_{AR} | \bar{\psi}_{i,0} \rangle,$$  \hspace{1cm} (3.5)

where

$$\hat{P}_{AR} \equiv \sum_{\mu \in A} | \phi_{\mu,R} \rangle \langle \phi_{\mu,R} |$$  \hspace{1cm} (3.6)

is the projector onto the AOs centered on atom $A_{R}$ defined via the biorthogonal AO basis as

$$\langle \hat{\phi}_{\mu,R} | \equiv \sum_{\nu,R'} (S^{-1})_{\mu,R'}^{\nu,R} \langle \phi_{\nu,R}' |,$$  \hspace{1cm} (3.7)

or its Bloch-AO equivalent,

$$\langle \hat{\phi}_{\mu,R} | \equiv \sum_{\nu,k} (S^{-1})_{\mu,R}^{\nu,k} \langle \phi_{\nu,k} |,$$  \hspace{1cm} (3.8)

and $S^{-1}$ denotes the inverse overlap matrix.

The key issue with the Mulliken charges as defined by Eq. (3.5) is their ill-defined nature for non-minimal basis sets. This impacts the robustness of the PM localization method by making the PM orbitals sensitive to variations in the orbital basis set (OBS) and geometry. Luckily, as noted by many$^{129–133}$, alternative charge definitions make the PM functional much
more robust (the interpretation of approaches in Refs. 129, 130 as PM with non-Mulliken charge definitions was pointed out in Ref. 132). To this end, Jónsson et al. utilized real-space partitioning of orbital charge densities to obtain robust atomic charges for use in the PM functional for periodic solids.\textsuperscript{143}

In this work, we defined the PM functional using atomic charges evaluated with the help of a pre-defined minimal basis set (MBS) of atomic orbitals. Such projection onto an MBS has been long employed to define OBS-independent atomic partitioning of densities and thus, due to its basis independent (intrinsic) nature, eliminates the undue sensitivity of charges to the orbital basis. The use of such MBS-projected charges, in the context of PM localization, has been frequently employed recently\textsuperscript{131–133,155} but the idea is much older: already Mulliken remarked that, “The ideal LCAO-MO population analysis would perhaps be in terms of free atom SCF AO’s”\textsuperscript{156} and goes back even earlier.

There is no unique method to use an MBS to define atomic charges, due to the arbitrariness of how to partition orbitals and/or their charge densities into atomic components. It is also far too easy to break desired invariants, such as the equality of the sum of atomic populations and the number of electrons. For example, replacing OBS AO $\phi_{\mu,R}$ by MBS AO $\tilde{\phi}_{\mu,R}$ in Eq. (3.6) (such replacement is only meaningful if the overlap inverse is defined, i.e., if the MBS is a subset of the OBS) violates such an invariant. Thus, existing approaches\textsuperscript{131,133,155,157,158} utilize the MBS indirectly, as a way to define an MBS-like subspace of the OBS. Here we use a simpler approach. Please note that, throughout the following discussion, quantities with an overbar are expressed in the MBS AO representation.
Consider the OBS AO representation of the Bloch orbitals, obtained by plugging Eq. (3.2) into Eq. (3.1):

$$|\psi_{i,k}\rangle = \sum_{\mu,R} C_{\mu,R}^{i,k} |\phi_{\mu,R}\rangle,$$

with $C_{\mu,R}^{i,k}$ defined as:

$$C_{\mu,R}^{i,k} \equiv \frac{1}{\sqrt{N}} C_{\mu,k}^{i} e^{i k \cdot R};$$

(3.10)

clearly, $C_{\nu,R}^{j,k} = C_{\nu,0}^{j,k} e^{i k \cdot R}$. Biorthogonal mapping of Bloch functions on the MBS,

$$|\bar{\psi}_{i,k}\rangle = \sum_{\mu,R} C_{\mu,R}^{j,k} |\bar{\phi}_{\mu,R}\rangle,$$

(3.11)
is trivially obtained by solving

$$\delta_{ij} = \langle \psi_{i,k} | \bar{\psi}_{j,k} \rangle = \sum_{\mu,R} \langle \psi_{j,k} | \bar{\phi}_{\mu,R} \rangle C_{\mu,R}^{j,k}$$

(3.12)

for coefficients $C_{\mu,R}^{j,k}$. A least-squares solution is produced by the Moore-Penrose pseudoinverse of the overlap matrix between the target set of Bloch orbitals for the given $k$ with the MBS AOs in the reference cell, $\bar{\phi}_{\mu,0}$:

$$(S)_{i,k}^{\mu} \equiv \langle \psi_{i,k} | \bar{\phi}_{\mu,0} \rangle = \sum_{\nu,R} \left( C_{\nu,R}^{i,k} \right)^* \langle \phi_{\nu,R} | \bar{\phi}_{\mu,0} \rangle.$$

(3.13)

Only simple Gaussian AO overlaps are needed to evaluate Eq. (3.13), with the lattice sum in Eq. (3.13) geometrically convergent. Evaluation in other numerical representations, such as plane waves (PWs), should be also straightforward. Extension of such pseudoinverse MBS mapping to the molecular case is obvious, where such procedure is related to how the corresponding orbitals are constructed. Note also that the pseudoinverse-mapped orbitals
are not orthonormal and so the pseudoinverse charges differ from the existing definitions for minimal-basis-derived charges;\textsuperscript{131,133,155,157,158} however, the relationship between the pseudoinverse MBS charges and other MBS-based charges is outside of the scope of this article and will be discussed elsewhere.

The OBS AO coefficients of the WF,

\begin{equation}
W_i^{\mu, R} \equiv \frac{1}{\sqrt{N_k}} \sum_{j,k} C_{j,k}^{\mu, R} U_{j,k}^i,
\end{equation}

are mapped straightforwardly to the MBS AO coefficients:

\begin{equation}
\overline{W}_i^{\mu, R} \equiv \frac{1}{\sqrt{N_k}} \sum_{j,k} \overline{C}_{j,k}^{\mu, R} U_{j,k}^i.
\end{equation}

The minimal-basis pseudoinverse charges are thus obtained by replacing the OBS with the MBS in Eq. (3.5) and replacing the corresponding OBS AO coefficient of the WF,\textsuperscript{79} \langle \bar{\psi}_{i,0} | \bar{\phi}_{\mu, R} \rangle \equiv W_i^{\mu, R}, with the corresponding MBS WF AO coefficient:

\begin{equation}
Q_i^{AR} = \frac{1}{2} \left[ \sum_{\mu \in A} \langle \bar{\psi}_{i,0} | \bar{\phi}_{\mu, R} \rangle \overline{W}_i^{\mu, R} + \text{h.c.} \right],
\end{equation}

where “h.c.” denotes the Hermitian conjugate. Evaluation of these charges in the LCAO representation leverages the Bloch-MBS overlaps (Eq. (3.13)) and is therefore completely straightforward. For a fixed MBS, these charges are expected to depend weakly on the OBS and have a well-defined basis set limit.

To avoid introducing a new symbol, \( P \) will henceforth denote the PM functional (Eq. (3.4)) defined with the MBS pseudoinverse charges.
3.3.3 PM Functional Maximization

Initial WF Guess

It is easy to see that the WFs, defined by the unitary matrices \( \{ U_k \} \) in Eq. (3.3), are in general not uniquely defined by the corresponding functional. The nonuniqueness stems from several factors. First, PM and other functionals defining WFs are invariant with respect to arbitrary permutations of the sequence of WFs. This may appear trivial, but in general means that comparing sets of WFs produced in two separate computations is not straightforward (see e.g. Ref. 143). Second, the WF functional is invariant with respect to all or some of the geometric transformations of the space group of the crystal (such as shifting a WF by a lattice vector). Third, the functionals defining WFs routinely have multiple maxima for a given system; hence, finding the global maximum is NP-hard. Thus, WF computation relies on heuristics to generate initial guesses for \( \{ U_k \} \); this initial guess and other solver details determine which functional maximum will be located.

Initial guesses for generalized WFs are typically obtained by projecting Bloch orbitals onto some trial functions. For example, Marzari and Vanderbilt utilized Gaussians located at expected centers of charge of Wannier functions, such as midbond centers;\textsuperscript{135} such user-controlled guess construction is also utilized by \textit{Wannier90}.\textsuperscript{138} A more automated approach was used by Zicovich-Wilson et al.,\textsuperscript{141} who approximately projected Bloch orbitals onto the reference cell’s AO basis (since they expanded the Bloch orbitals in an AO basis already, such projection was trivial); unfortunately, such a choice is not appropriate when covalent
bonds cross boundaries of the unit cell. A similar approach was used by Mustafa et al., who projected Bloch orbitals (expanded in PW basis) onto an appropriate set of AOs that spanned a space containing the target Wannier set. To account for the covalent bonds crossing the unit cell boundaries, the projection AO set must include AOs not only in the reference cell but also in its nearest periodic images.

Note that these projection-based approaches are still not robust enough to deal with multiple minima, since it is necessary to generate multiple initial guesses to probe the global optimality of the resulting WFs. Thus, Jónsson et al. simply generated guess WFs using randomly-generated \( \{ \mathbf{U}_k \} \) and ran multiple computations.

Here we have devised a novel automated “Canonicalize Phase then Randomize” (CPR) method for generating guess WFs that can be applied to arbitrary Bloch orbitals expressed in LCAO and non-LCAO representations. The first step in this method is motivated by the realization that, to produce localized Wannier functions even for bands composed of a single atomic orbital (e.g., core bands), it is helpful to canonicalize the phases of the AO coefficients at different \( k \) points. Such phase canonicalization can be viewed as removing the gauge freedom of the Bloch orbitals; once the arbitrariness of the gauge is removed, then the original Wannier prescription, Eq. (3.3) with \( \{ \mathbf{U}_k \} \) set to identity, will recover the maximally-localized state, namely, the atomic orbital in the reference cell. Of course, the generalized Wannier functions can compensate for the gauge freedom of the Bloch orbitals via the \( k \)-dependence of \( \{ \mathbf{U}_k \} \); thus, the generalized Wannier orbital for a single-AO band with arbitrary gauge transformation will still be a single AO (although it will not necessarily
be the reference cell AO. But by including the phase canonicalization of the input Bloch orbitals, the WF functional maximization becomes more robust by starting from a good initial guess.

Phase canonicalization in the CPR method proceeds as follows:

- The set of Bloch orbitals at the Γ point \((k = 0)\) is split into degenerate subsets (bands).

  A set of orbitals is considered degenerate if its eigenenergies are within a prescribed energy tolerance \(\epsilon\).

- For each band \(\alpha\), the phase-defining subset of AOs, \(\{\mu_\alpha\}\), includes AOs \(\mu\) that have the largest occupancies in the band:

  \[
  \rho^{\alpha,0}_{\mu,0} \equiv \sum_{i_\alpha} |C^{i_\alpha,0}_{\mu,0}|^2 ,
  \]
  \(3.17\)

  where \(i_\alpha\) are the Bloch orbitals in band \(\alpha\). The phase-defining AO set, \(\{\mu_\alpha\}\), thus includes the AOs with the greatest contribution to the band; it can include a single AO (most common) or multiple AOs (due to geometric symmetry and band degeneracy).

- The phase of every Bloch orbital, \(i\), at the Γ point is aligned so that the coefficient of the first phase-defining AO, \(\mu^{(0)}_{\alpha_i}\), for its band, \(\alpha_i\), is positive:

  \[
  C^{i}_{\mu,0} \rightarrow C^{i}_{\mu,0} \times \left( \frac{C^{i}_{\mu^{(0)}_{\alpha_i},0}}{C^{i}_{\mu^{(0)}_{\alpha_i},0}} \right) .
  \]
  \(3.18\)

  This step eliminates possible arbitrary phase factors that were introduced by the SCF solver and makes all AO coefficients at the Γ point real.
Bloch orbitals at every \( \mathbf{k} \) point are next mapped to the matching orbital at the \( \Gamma \) point. To perform such matching consider orbitals \( i \) and \( j \) at two neighboring points \( \mathbf{k} \) and \( \mathbf{k}' \), respectively. Their reference cell MBS “overlap” is defined as the overlap of Bloch orbital \( \psi_{i,k} \) with the Bloch orbital \( \psi_{j,k'} \) pseudoinverse-projected (see Section 3.3.2) onto the reference cell’s MBS AO basis:

\[
\bar{S}_{j,k'}^{i,k} = \sum_{\mu} \langle \psi_{i,k}|\tilde{\phi}_{\mu,0} \rangle \overline{C}_{\mu,0}^{j,k'}.
\] (3.19)

All matrix elements in this equation were already evaluated when computing the MBS AO projections of the Bloch orbitals for the purpose of computing atomic charges. For every \( i \in [0,o) \), where \( o \) is the number of Bloch orbitals being localized, \( \psi_{i,k} \) matches orbital \( \psi_{j,k'} \) point if \( |\bar{S}_{j,k'}^{i,k}| \) is the largest among all \( j \); if the match candidate \( \psi_{j,k'} \) has been declared a match for another orbital \( i' < i \), the next best candidate is chosen.

If orbital \( \psi_{i,k} \) was matched to orbital \( \psi_{j,k'} \), its phase is canonicalized such that its reference-cell MBS overlap is real:

\[
C_{\mu,k}^{i} \rightarrow C_{\mu,k}^{i} \times \left( \frac{\bar{S}_{j,k'}^{i,k}}{\bar{S}_{j,k'}^{0,k}} \right).
\] (3.20)

Since the goal is to align the phases of bands at all \( \mathbf{k} \) points to the bands at the \( \Gamma \) point, the band matching is performed using sequences of \( \mathbf{k} \) points originating from the \( \Gamma \) point that span the entire first Brillouin zone mesh. Assuming that points in a uniform mesh of \( \mathbf{k} \) points are indexed by triplets \( (i,j,k) \), with \( i,j,k = 0, \pm 1, \pm 2... \pm \lfloor N_k/2 \rfloor \), for 1-dimensional structures bands at the mesh points \((1,0,0)\) and \((-1,0,0)\) are matched to the canonicalized bands at \((0,0,0)\) (\( \Gamma \) point), then bands at mesh points \((2,0,0)\) and
\((-2, 0, 0)\) are matched to the canonicalized bands at \((1, 0, 0)\) and \((-1, 0, 0)\), respectively, and so on. Similarly, for a 2-dimensional structure, the bands at \((i, 1, 0)\) and \((i, -1, 0)\) are matched to the canonicalized bands at \((i, 0, 0)\), etc.

- To account for band crossings, bands at point \(k\) are sorted to appear in the same order as their matching bands at point \(k'\).

Despite its relative simplicity, the phase canonicalization is fairly robust and significantly improves the quality of the trial Wannier functions (see Supporting Information). However, the described algorithm is not perfect: (1) it requires a dense Brillouin zone mesh to track high-dispersion bands across the Brillouin zone reliably, (2) it relies on an ad hoc way of matching bands, and (3) it does not account for arbitrary rotations among the degenerate bands. Work to address these shortcomings is underway and will be presented elsewhere.

Performing the phase canonicalization ensures that using identity for \(\{U_k\}\) produces well-localized WFs for many bands. Note, however, that the “intra-cell” localization is not assisted by the phase canonicalization; thus, it alone will not be sufficient for systems with large unit cells. To be able to locate the global maxima of the PMWF functional by sampling the initial guesses, by default we initialize \(\{U_k\}\) with a (quasi)random unitary matrix generated from a user-supplied seed. Note that the same unitary matrix is used for every \(k\) in order to preserve the benefit of phase canonicalization.

All computations reported in the manuscript used the CPR guess generated with the same (default) seed value. In the Supporting Information, we report additional computations.
that, after the phase canonicalization, initialized \( \{ U_k \} \) with identity matrices as well as with random \( \{ U_k \} \) generated nonuniformly across the first Brillouin zone (i.e., a different quasirandom matrix was used for every \( k \) quadrature point, thereby canceling the benefit of phase canonicalization). The performance of the default CPR and “identity” CPR (CP) was found to be similar, whereas using the random nonuniform guess required significantly more iterations to reach convergence; however, the final PM functional value was found to be the same for all initial guesses.

Clearly, since the CPR method is defined intrinsically without any reference to the LCAO representation, it can be utilized in the context of non-LCAO (PW and other) representations as long as the Bloch orbitals can be projected onto the MBS AOs. As discussed above, projection on localized states is common in preparing trial Wannier functions,\textsuperscript{135,141,160} the use of a full minimal AO basis for projection makes the CPR guess (1) well-defined even in the limit of a complete orbital basis set (in contrast to the approach of Ref. 141) and (2) more black-box by eliminating the need to guess positions or composition of target Wannier functions (in contrast to the approaches of Refs. 135, 160).
PM Functional Gradient

To find a maximum of the PM functional, we will use global gradient-based optimization. The gradient of $P$ with respect to $\{U_k\}$ is expressed straightforwardly:

$$\frac{\partial P}{\partial (U_{i,k}^*)} = \frac{p}{\sqrt{N_k}} \sum_{A,R} |Q_i^A|^p \sum_{\mu \in A} \left[ \langle \psi_{j,k} | \bar{\phi}_{\mu,R} | W_{\mu,R}^i \rangle + \left( \overline{C_{\mu,R}^{j,k}} \right)^* \langle \bar{\phi}_{\mu,R} | \tilde{\psi}_{i,0} \rangle \right]$$

$$= \frac{p}{\sqrt{N_k}} \sum_{A,R} |Q_i^A|^p e^{-ik \cdot R} \sum_{\mu \in A} \left[ \langle \psi_{j,k} | \bar{\phi}_{\mu,0} | W_{\mu,R}^i \rangle + \left( \overline{C_{\mu,0}^{j,k}} \right)^* \langle \bar{\phi}_{\mu,0} | \tilde{\psi}_{i,0} \rangle \right] \quad (3.21)$$

Eq. (3.21) uses the standard complex-valued form of the derivative of a real-valued function of complex-valued parameters, $(\partial f/\partial z^*) \equiv (\partial f/\partial \text{Re } z) + i (\partial f/\partial \text{Im } z)$, which makes the notation more compact. The evaluation of the gradient again leverages the Bloch-MBS overlaps (Eq. (3.13)) and is straightforward.

It is of course more convenient to express the PMWF functional in terms of nonredundant variables by introducing the standard exponential parametrization of a unitary matrix, $U \equiv \exp(\kappa - \kappa^\dagger)$, where $\kappa$ is a complex triangular matrix. It is straightforward to convert the “Euclidean” gradient arranged as a matrix for each $k$,

$$(\Gamma_k)^i_j \equiv \frac{\partial P}{\partial (U_{i,k}^*)} \quad (3.22)$$

to its “curvilinear” counterpart,

$$(G_k)^i_j \equiv \frac{\partial P}{\partial (\kappa_{i,j,k}^*)} \quad (3.23)$$

as follows:

$$G_k = (U_k)^\dagger \Gamma_k - (\Gamma_k)^\dagger U_k. \quad (3.24)$$
Note that $G_k$ is antihermitian, just like $\kappa_k - \kappa_k^\dagger$. Maximization of the PMWF functional expressed in terms of $\kappa_k$ is a standard (unconstrained) nonlinear optimization problem; its solution is described in the following section.

**Direction Choice: SA, CG, BFGS**

The main focus on our work is how to solve for the PMWFs robustly. As the reference methods for locating the PM functional maxima, we will use the steepest ascent (SA) and (nonlinear) conjugate gradient (CG) methods. In particular, we have chosen three specific varieties of CG for comparison: the Polak-Ribière formulation (CGPR),\textsuperscript{164} the Fletcher-Reeves formulation (CGFR),\textsuperscript{165} and the Hestenes-Stiefel formulation (CGHS).\textsuperscript{166} Due to the well-known nature of SA and nonlinear CG (see, for example, any textbook on numerical optimization), we will not discuss their implementation details here, except to note that, for each of the three different CG variants considered, we also varied the number of SA steps taken before beginning CG. The numbers of initial SA steps considered in this work were 1, 2, 5, 10, and 15, meaning that, for each system, we ran a total of 15 different CG calculations. Also note that the optimization problem of the real-valued PMWF functional $P$ was recast (as usual) in terms of real and imaginary components of the complex-valued parameters $U_i^{j,k}$, i.e., henceforth the gradient and other vectors will consist of $o(o - 1)N_k$
real numbers only, where \( o \) is the number of Bloch orbitals being localized;\(^4\) complex-valued formulations of the optimization problem\(^{167} \) were not considered here.

The BFGS method, though also well-known, warrants a bit of discussion. In particular, we have employed the “two-loop recursion” form of the limited-memory BFGS\(^{168} \) (L-BFGS; henceforth we will omit the “L-” prefix unless this algorithmic detail is relevant) algorithm for updating the estimated inverse Hessian; the initial estimate of the inverse Hessian was chosen to be an identity matrix. Because each BFGS iteration depends on some number of prior iterations (the “history”) to generate an updated estimate of the inverse Hessian, it is necessary to select the size of this history (i.e., the number of iterations kept). In addition, regardless of the history size, the first update must be necessarily be steepest ascent since the history does not yet exist. Of course, it is also possible to perform any number of SA steps before beginning the BFGS procedure, and it is these two parameters (the history size and the number of initial SA steps) that define the BFGS algorithm as implemented here. For all systems, we used five different initial SA values and five different history sizes, for a total of 25 BFGS solver setups. The initial SA values considered were 1, 2, 5, 10, and 15; the history sizes considered were also 1, 2, 5, 10, and 15. In future discussion, BFGS parameters are indicated as \( \langle \text{No. of SA steps}, \text{history size} \rangle \).

\(^4\)In practice the implementation uses pairs of real square matrices to represent \( \kappa_k, G_k \), thus using \( 2o^2N_k \) (instead of \( o(o-1)N_k \)) parameters. This is due to the lack of support for the (anti)symmetric matrix format in the Eigen library used for numerical manipulations in the PM solver.
Line Search

Regardless of how the direction was chosen (SA, CG, BFGS), the line search was performed in the same manner, using a low-order polynomial approximation of the objective function along the trial direction. First, the proposed direction is checked to point uphill (if not, the trial direction is reversed). Then, given a fitting range upper bound $T_\mu$ (see below) and the polynomial order $n = 4$, the PM functional is evaluated at $n + 1$ evenly-spaced points, \{\mu_0 \equiv 0, \mu_1 \equiv T_\mu/n, \mu_2 \equiv 2T_\mu/n \ldots \},$ in $[0, T_\mu]$. The \{P(\mu_i)\} set is then used to construct a polynomial fit, $f(\mu_i)$, and the bisection method is used to find a root of $f'(\mu_i)$. The fitting range and polynomial orders are determined as follows.

- **Iteration 0:** $T_\mu$ is estimated from the shortest orbital rotation frequency $\omega_{\text{max}}$ along the given direction via Eq. (15) of Ref. 163 (the largest $\omega_{\text{max}}$ is chosen among all $k$ points).

- **Iteration $i$:** The upper bound from the previous iteration is used as a trial upper bound. If $P(\mu_1)$ is less than $P(0)$ then $T_\mu$ is reset to $\mu_1$, else $P(\mu_k)$ are evaluated for increasing $k$ until $P(\mu_k) < P(\mu_{k-1})$ is found. If such $k$ is found, values \{P(\mu_i)\}, $i \in [0, k]$ are fitted to a polynomial of order $k - 1$; else $T_\mu$ is reset to $5T_\mu$ and the process is repeated.

Finally, if we fail to find an acceptable upper bound along a chosen direction, we will reset to the SA direction. If the upper bound is not located in the SA direction or the root finder fails, then the upper bound is recomputed as done at the start. These resets are rarely necessary when performing BFGS.
Note that, in addition to resetting the CG direction to the SA direction whenever an acceptable upper bound cannot be found, it is also necessary to reset the CG direction every \( n \) iterations, where \( n \) is the number of orbitals being localized. This is because a system with \( n \) variables can only have \( n \) conjugate directions.

### 3.4 Computational Details

All calculations were carried out in the developmental version of the Massively Parallel Quantum Chemistry (MPQC) package (version 4.0.0).\(^{90}\) All computations were performed on the Cascades commodity cluster at Virginia Tech.\(^{169}\)

Hartree-Fock computations were carried out using the reduced-scaling LCAO formalism described in Ref. 41. The Coulomb potential was evaluated using multipole-accelerated real-space summation and density-fitting approximation, whereas the exchange potential was evaluated using concentric atomic density fitting.\(^{41}\) Table 3.1 lists the test systems as well as the corresponding orbital basis set, the Monkhorst-Pack mesh size, and the PM convergence threshold employed for each. In all cases, the def2-SVP-J basis set\(^ {170}\) was used as the density fitting basis. The pseudoinverse atomic charges were evaluated using the Huzinaga MINI basis set\(^ {171–173}\) as the minimal AO basis. All occupied orbitals (including core) were localized. The PM functional with \( p = 4 \) (see Eq. (3.4)) was utilized throughout. Note that the convergence thresholds that we use in this work are significantly tighter than the typical threshold of \( 10^{-5} \) utilized in the comparable studies.\(^ {144,151}\) A slightly looser con-
vergence in bulk Si was due to the greater range of the lengthscales of the Wannier functions in that system; localizing core and valence orbitals separately would alleviate these issues, but was not pursued in order to keep the solver assessment protocol as uniform and as stringent as possible.

Table 3.1: Test systems used to assess the performance of PMWF solvers, along with the relevant computational details.

<table>
<thead>
<tr>
<th>System</th>
<th>OBS</th>
<th>Monkhorst-Pack mesh size$^a$</th>
<th>PM gradient norm</th>
</tr>
</thead>
<tbody>
<tr>
<td>trans-(C$_2$H$<em>2$)$</em>\infty$</td>
<td>6-31G*</td>
<td>101</td>
<td>$10^{-8}$</td>
</tr>
<tr>
<td>(C$_2$H$<em>4$)$</em>\infty$</td>
<td>6-31G*</td>
<td>101</td>
<td>$10^{-8}$</td>
</tr>
<tr>
<td>(4,0) nanotube</td>
<td>6-31G*</td>
<td>51</td>
<td>$10^{-8}$</td>
</tr>
<tr>
<td>Graphene</td>
<td>6-31G</td>
<td>$21 \times 21$</td>
<td>$10^{-8}$</td>
</tr>
<tr>
<td>h-BN</td>
<td>6-31G</td>
<td>$21 \times 21$</td>
<td>$10^{-8}$</td>
</tr>
<tr>
<td>LiH</td>
<td>CR-cc-pVDZ$^{74}$</td>
<td>$11 \times 11 \times 11$</td>
<td>$10^{-8}$</td>
</tr>
<tr>
<td>Diamond</td>
<td>6-31G*</td>
<td>$11 \times 11 \times 11$</td>
<td>$10^{-8}$</td>
</tr>
<tr>
<td>Silicon</td>
<td>6-31G*</td>
<td>$11 \times 11 \times 11$</td>
<td>$10^{-7}$</td>
</tr>
</tbody>
</table>

$^a$ All systems except h-BN utilized primitive unit cells; an orthorhombic non-primitive unit cell was used for h-BN.
3.5 Results

As discussed in Section 3.3.3, the PM functional does not specify WFs uniquely; thus, to compare computed WFs, we compare the corresponding values of the PM functional. Two WF sets will be referred to as equivalent if they correspond to the same value of the PM functional up to a target precision. For every test system, all PMWF solvers considered in this work (BFGS with 25 parameter settings, CG with 15 parameter settings, and SA; see Section 3.3.3), when converged successfully, produced WFs that were practically equivalent; only for silicon and the carbon nanotube did the variance of the final $P$ value exceeded $10^{-8}$ (see the Supporting Information for more details). Limited testing also indicated that the use of the standard (CPR) and nonstandard guesses produced equivalent sets of Wannier functions.

Although all solvers produced equivalent sets of PMWFs, the number of iterations needed to locate the maximum of the PM functional differed dramatically between the different classes of solvers. Column “Min” in Table 3.2 lists the minimum number of iterations needed to arrive at the solution, broken down by the system and solver class. In all cases except the zero-gap system, the BFGS solver converged in fewer than 60 iterations in the best-case scenario, and only two systems required more than 100 iterations in the worst-case scenario. Each of the three CG variants managed to converge in under 100 iterations for at least one system, but even CGFR, which saw the most success in converging in less than 100 iterations, only did so for three systems. In other cases, the various CG variants could take thousands
Table 3.2: Statistical analyses of the number of iterations to solution, broken down by system
and solver type. “CG” denotes the aggregate data of the 3 CG solvers. Computations that
failed to converge in 4000 iterations were deemed “failed to converge” and were not included
in the statistics.

<table>
<thead>
<tr>
<th>System</th>
<th>Solver</th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>St. Dev.</th>
<th>System</th>
<th>Solver</th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>St. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_2H_2$</td>
<td>BFGS</td>
<td>31</td>
<td>47</td>
<td>40.5</td>
<td>3.5</td>
<td>BFGS</td>
<td>45</td>
<td>57</td>
<td>50.2</td>
<td>3.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>1398</td>
<td>2534</td>
<td>2007.5</td>
<td>389.1</td>
<td>CG</td>
<td>174</td>
<td>808</td>
<td>260.8</td>
<td>160.9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CGPR</td>
<td>1398</td>
<td>1934</td>
<td>1708.6</td>
<td>209.0</td>
<td>h-BN</td>
<td>CGPRc</td>
<td>174</td>
<td>808</td>
<td>339.3</td>
<td>312.7</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>1752</td>
<td>1832</td>
<td>1801.0</td>
<td>35.5</td>
<td>CGFR</td>
<td>174</td>
<td>250</td>
<td>210.6</td>
<td>35.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CGHS</td>
<td>2483</td>
<td>2534</td>
<td>2513.0</td>
<td>22.5</td>
<td>CGHS</td>
<td>235</td>
<td>257</td>
<td>248.2</td>
<td>8.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SA</td>
<td>1745</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>SA</td>
<td>175</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>$C_2H_4$</td>
<td>BFGS</td>
<td>24</td>
<td>37</td>
<td>29.0</td>
<td>3.2</td>
<td>BFGS</td>
<td>10</td>
<td>22</td>
<td>14.5</td>
<td>4.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>66</td>
<td>729</td>
<td>230.7</td>
<td>209.8</td>
<td>CG</td>
<td>14</td>
<td>1441</td>
<td>783.2</td>
<td>607.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CGPR</td>
<td>69</td>
<td>425</td>
<td>215.0</td>
<td>133.2</td>
<td>LiH</td>
<td>CGPR</td>
<td>786</td>
<td>1062</td>
<td>900.6</td>
<td>122.7</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>109</td>
<td>187</td>
<td>158.6</td>
<td>33.8</td>
<td>CGFR</td>
<td>14</td>
<td>23</td>
<td>17.6</td>
<td>3.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CGHS</td>
<td>66</td>
<td>729</td>
<td>318.4</td>
<td>344.6</td>
<td>CGHS</td>
<td>1395</td>
<td>1441</td>
<td>1431.4</td>
<td>20.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SA</td>
<td>52</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>SA</td>
<td>1457</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>Diamond</td>
<td>BFGS</td>
<td>23</td>
<td>28</td>
<td>24.7</td>
<td>1.4</td>
<td>BFGS</td>
<td>75</td>
<td>101</td>
<td>86.1</td>
<td>8.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>39</td>
<td>212</td>
<td>89.6</td>
<td>65.3</td>
<td>CG</td>
<td>713</td>
<td>2275</td>
<td>1174.5</td>
<td>471.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CGPR</td>
<td>43</td>
<td>212</td>
<td>133.0</td>
<td>69.6</td>
<td>Nanotube</td>
<td>CGPR</td>
<td>935</td>
<td>2275</td>
<td>1309.0</td>
<td>581.3</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>39</td>
<td>51</td>
<td>46.2</td>
<td>5.1</td>
<td>CGFR</td>
<td>1015</td>
<td>1903</td>
<td>1447.8</td>
<td>334.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CGHS</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>CGHS</td>
<td>713</td>
<td>853</td>
<td>766.8</td>
<td>55.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SA</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>SA</td>
<td>485</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>Graphene</td>
<td>BFGS</td>
<td>53</td>
<td>186</td>
<td>86.0</td>
<td>42.0</td>
<td>BFGS</td>
<td>21</td>
<td>29</td>
<td>24.1</td>
<td>2.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>2657</td>
<td>3666</td>
<td>3296.8</td>
<td>275.8</td>
<td>CG</td>
<td>32</td>
<td>2975</td>
<td>954.9</td>
<td>1240.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CGPRc</td>
<td>2657</td>
<td>3666</td>
<td>3234.8</td>
<td>439.4</td>
<td>Silicon</td>
<td>CGPR</td>
<td>113</td>
<td>221</td>
<td>176.4</td>
<td>44.3</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>3321</td>
<td>3376</td>
<td>3346.4</td>
<td>21.6</td>
<td>CGFR</td>
<td>32</td>
<td>83</td>
<td>55.4</td>
<td>19.9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CGHS</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>CGHS</td>
<td>2243</td>
<td>2975</td>
<td>2633.0</td>
<td>310.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SA</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>SA</td>
<td>3171</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
</tbody>
</table>

a All five calculations failed to converge.

b Calculation failed to converge.

c One calculation failed to converge.
of iterations to converge, and CGHS failed to converge in 4,000 iterations for both diamond and graphene, regardless of the starting number of steepest ascent steps. Depending on the system, SA could converge in under 50 iterations or take hundreds or thousands to converge; in two cases, it failed to converge at all within 4,000 iterations. Even in the best-case scenario (i.e., minimum iterations to convergence), BFGS was always superior to CG and SA, though the latter two could sometimes come close. But even when CG and SA were nearly comparable to BFGS in terms of number of iterations to solution, their convergence behavior could not be counted on, as is evidenced by the much larger calculation length standard deviations for these solvers compared with the same metric for BFGS. Overall, the performance of the SA and CG solvers can, at best, be characterized as unreliable; in contrast, BFGS is reliably rapid.

Due to the significant variation in the performance of the CG and SA solvers for different systems, it is difficult to pinpoint the origin of their struggles. Figure 3.1 illustrates the convergence patterns observed for representative 1-d, 2-d, and 3-d test systems. The superior performance of BFGS compared to CG and SA is plainly visible. Also note the extended plateau exhibited by the CG and SA solvers in the 1-d system, which is a typical CG convergence pattern and which suggests a high condition number of the PM Hessian in this system. The BFGS solver also exhibits a plateau in this system, but it is much shorter. Lastly, the quality of the initial guess can vary greatly from system to system: in the 1-d system, the initial guess is clearly significantly worse than in the 2-d and 3-d systems, as indicated by significantly larger initial deviations.
Figure 3.1: Plots of the difference between the PM functional value $P$ and the converged value $P_0$ vs the iteration count. For each system represented, the shortest calculation in each category was chosen; $P_0$ was taken to be the greatest final $P$ value across the selected calculations for a given system. 

(Continued on next page)
Figure 3.1: (Continued from previous page) Plots of the difference between the PM functional value $P$ and the converged value $P_0$ vs the iteration count. For each system represented, the shortest calculation in each category was chosen; $P_0$ was taken to be the greatest final $P$ value across the selected calculations for a given system.
The performance of BFGS is also relatively insensitive to the choice of its parameters, namely the number of SA steps at the start and the history size, as illustrated in Table 3.2. The small standard deviation of the BFGS solvers’ performance (< 10 for all systems other than graphene) indicates that approximately the same number of iterations is needed to locate the maximum regardless of the BFGS parameter values. In contrast, the CG solvers’ performance can depend strongly on the number of starting SA steps. This correlates with the unreliable performance of the CG solvers that we noted previously.

In addition, the number of iterations needed for the CG solver also correlates as expected with the findings of Jónsson et al., whose implementation of CG required hundreds of iterations in periodic systems. Cases where our CG implementation converged in fewer iterations may be a result of differences in initial guess (CPR in our work, random guess in their work), while situations in which our CG implementation took longer may be due to the tighter convergence thresholds employed herein. Furthermore, the only 3-d system that Jónsson et al. considered was a benzene crystal, which, like other molecular crystals, would have low-dispersion band structure and for which it should be easier to generate a localized initial guess (see our discussion in Section 3.3.3). We focused on the more challenging ionic and covalent 3-d systems.
Table 3.3: Statistical analyses of the number of iterations to solution for each BFGS solver parameter set. Colored boxes identify the smallest and largest values in each column as follows: green (smallest), blue (second smallest), red (largest), orange (second largest). A “Min” value of 1 indicates that this solver was the fastest (required the fewest iterations) for at least 1 test system. A “Max” value of 1.5 means that, for every test system, this solver required at most 50% more iterations than the fastest solver for that system. A “Mean” value of 1.2 means that, for a given test system, this solver required on average 20% more iterations than the fastest solver for that system.

<table>
<thead>
<tr>
<th>⟨# of SA, History⟩</th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>St. Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>⟨1, 1⟩</td>
<td>1.00000</td>
<td>2.83019</td>
<td>1.33181</td>
<td>0.61693</td>
</tr>
<tr>
<td>⟨1, 2⟩</td>
<td>1.00000</td>
<td>1.81132</td>
<td>1.19016</td>
<td>0.27206</td>
</tr>
<tr>
<td>⟨1, 5⟩</td>
<td>1.04348</td>
<td>1.35849</td>
<td>1.20877</td>
<td>0.11738</td>
</tr>
<tr>
<td>⟨1, 10⟩</td>
<td>1.03774</td>
<td>1.41935</td>
<td>1.19625</td>
<td>0.14074</td>
</tr>
<tr>
<td>⟨1, 15⟩</td>
<td>1.01887</td>
<td>1.38095</td>
<td>1.18609</td>
<td>0.12330</td>
</tr>
<tr>
<td>⟨2, 1⟩</td>
<td>1.00000</td>
<td>3.50943</td>
<td>1.41816</td>
<td>0.85422</td>
</tr>
<tr>
<td>⟨2, 2⟩</td>
<td>1.00000</td>
<td>1.43396</td>
<td>1.14882</td>
<td>0.13846</td>
</tr>
<tr>
<td>⟨2, 5⟩</td>
<td>1.04348</td>
<td>1.22581</td>
<td>1.13897</td>
<td>0.06151</td>
</tr>
<tr>
<td>⟨2, 10⟩</td>
<td>1.03774</td>
<td>1.41935</td>
<td>1.14482</td>
<td>0.12524</td>
</tr>
<tr>
<td>⟨2, 15⟩</td>
<td>1.01887</td>
<td>1.19355</td>
<td>1.11134</td>
<td>0.05189</td>
</tr>
</tbody>
</table>

Continued on next page
Although the BFGS-based PMWF solver is already highly robust, some parameter choices are systematically better than others. Thus, we analyzed the distribution of the number
of iterations needed to locate the solution for a given system with the given BFGS solver parameter values relative to the smallest number of iterations needed for that system; the results of this analysis are listed in Table 3.3 (see the Supporting Information for the raw number of iterations for each system). We highlighted the smallest and largest values in each column to make it easier to locate the fastest and slowest solvers. The \( \langle 2,15 \rangle \) BFGS solver is overall the fastest, both on average and in the worst-case scenario, and thus is the recommended default choice.

### 3.6 Summary and Perspective

We described a robust BFGS-based solver to obtain (generalized) Pipek-Mezey Wannier functions in periodic solids whose use was pioneered recently by Jónsson et al.\textsuperscript{143} The PM functional used in this work utilized atomic charges using a simple pseudoinverse projection onto a pre-defined minimal AO basis, thus making its evaluation convenient in a periodic LCAO representation. An essential contributor to the robustness of the solver is the novel automated “Canonicalize Phase then Randomize” (CPR) method for generating the initial guess. The limited-memory BFGS solver converged very tightly in fewer than 60 iterations in 1-, 2-, and 3-dimensional systems featuring a variety of bonding patterns (covalent, ionic) and gaps, even in systems with very large BvK unit cells (thousands of atoms). The sole exception was one system with a vanishing gap where \( \sim 80 \) iterations were needed. This is a significant improvement on the more traditional SA-based solver that can require hundreds
or thousands of iterations, or the nonlinear CG solvers that often converge faster than SA, but can unpredictably converge very slowly or even fail to converge at all. Although the performance of the solver was relatively insensitive to the BFGS history size, the near-optimal choice of history size was determined to be 1, making the BFGS solver notionally similar in the operation and storage costs to that of CG.

Clearly, the BFGS-based solver should be robustly usable for computing other generalized Wannier functions, such as the Boys (maximally-localized) WFs. Although here we only explored localization of occupied states, the solver should be also applicable to the unoccupied states with valence character. The automated CPR method for generating initial WF guesses could be used in conjunction with PW-based representations of Bloch orbitals, potentially improving on the existing approaches\cite{135,160}. Lastly, it is also worthwhile to assess the efficacy of BFGS-based solvers for other challenging orbital optimization problems in molecules and solids, such as localization of unoccupied (virtual) orbitals\cite{144,151} and for the orbital optimization in the context of Perdew-Zunger self-interaction-corrected DFT (notably, some limited use of BFGS in this context was recently reported by Lehtola et al.\cite{176}).
Chapter 4

Conclusions and Future Outlook

Throughout the course of this work, we have discussed the cost problem associated with wavefunction-based computational quantum chemistry, particularly the high non-physical scaling from which conventional many-body methods suffer. Noting that computational scaling can often be decreased by exploiting the local nature of electron correlation, we have considered some of the ways that this has already been done – namely, via (occupied) orbital localization and virtual space compression – and we have discussed how our own work has served to expand this rather important field.

In particular, we have implemented an iteratively-optimized pair natural orbital (“iPNO”) approach in the context of the coupled cluster singles and doubles (CCSD) method. Such iPNOs were found to provide superior accuracy relative to conventional PNOs, which are generated once at the beginning of a correlated calculation. Although iPNOs allow us to
recover a greater percentage of the canonical CCSD energy relative to traditional PNOs at the same truncation threshold, it is unclear if or when the use of iPNOs will be warranted, especially when compared with the performance of the very successful family of domain-localized pair natural orbital (DLPNO) based methods.\textsuperscript{39,43,78} However, we stress that our pilot implementation of the iPNO-CCSD solver was merely that: a pilot implementation. The iPNO-CCSD results were simulated within the context of a production-level canonical CCSD solver, instead of being generated via a \textit{bona fide} production-level PNO-CC solver. This simulation purposely did not take advantage of a number of additional potential approximations (such as PAO domains or local density fitting domains) so that we could, as far as possible, determine how much of the error associated with truncated PNOs is a result of MP1 PNOs being a suboptimal representation of the CC doubles amplitudes. A production-level iPNO-CC solver would, of course, introduce these approximation techniques, as well as other means of increasing efficiency, such as DIIS extrapolation between macroiterations or the use of a gradient-based optimization of the PNOs. And whether or not iPNOs become competitive with traditional PNOs for computations on molecular systems, the discussion is still worthwhile because it highlights the inherent discrepancies between the MP1 amplitudes and the CC doubles amplitudes.

Though the work on iPNOs was undertaken in the molecular context, there is no reason why it cannot be extended to the realm of periodic crystalline solids; or, if iPNOs are determined to be important primarily as a proof of concept and not as a viable reduced-scaling tool, it is equally true that we would like to take advantage of the myriad of reduced-scaling
local formulations of highly-accurate many-body methods that currently exist for molecular systems. Though this work did not explore applying such methods to periodic crystalline systems, we have described the crucial first step: generating a set of real, localized Wannier functions that can be used in much the same way as localized molecular orbitals are used in the molecular case. We stress that, although the periodic Hartree-Fock procedure must be undertaken in the space of $N_k$ sets of complex basis functions (Bloch orbitals), as opposed to the molecular case where we just have a single set of (usually real) basis functions, the localization procedure restores the realness to the representation of the wavefunction. That is, whereas the crystalline orbitals (COs) that resulted from the periodic Hartree-Fock procedure were complex, the Wannier functions that come from our periodic localization procedure are again real. And even though we formally have $N_R$ sets of Wannier functions, since each set is just a translation of those Wanniers centered in the reference cell, we can select just the reference Wanniers for use in post-Hartree-Fock calculations. This correspondence between the Wannier functions and the localized MOs should make application of any molecular many-body method to the crystalline case relatively trivial, and it is our hope that the Massively Parallel Quantum Chemistry package will soon boast a competitive library of highly-accurate, reduced-scaling many-body methods for both molecules and periodic crystalline solids.
Appendix A

Supporting Information for “Robust Pipek-Mezey Orbital Localization in Periodic Solids”

A.1 PM Functional Values Obtained with Different Solvers

For each system and solver class, we have determined the minimum, maximum, and mean final PM functional values, along with the standard deviation. Calculations that failed to converge in 4,000 iterations were excluded, though only when all calculations in a particular solver class failed to converge is this noted in this document. More details about the number and type of calculations that failed to converge can be found in the footnotes to Table 3.2 in the main text.
Table A.1: P Value Summary

<table>
<thead>
<tr>
<th>System</th>
<th>Solver</th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>St. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>trans-(C₂H₂)₅</td>
<td>L-BFGS</td>
<td>4.56799</td>
<td>4.56799</td>
<td>4.56799</td>
<td>1.28433 × 10⁻¹²</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>4.56799</td>
<td>4.56799</td>
<td>4.56799</td>
<td>2.24573 × 10⁻¹²</td>
</tr>
<tr>
<td></td>
<td>CGPR</td>
<td>4.56799</td>
<td>4.56799</td>
<td>4.56799</td>
<td>6.66975 × 10⁻¹³</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>4.56799</td>
<td>4.56799</td>
<td>4.56799</td>
<td>1.63555 × 10⁻¹²</td>
</tr>
<tr>
<td></td>
<td>CGHS</td>
<td>4.56799</td>
<td>4.56799</td>
<td>4.56799</td>
<td>7.75747 × 10⁻¹³</td>
</tr>
<tr>
<td>(C₂H₄)₅</td>
<td>L-BFGS</td>
<td>5.12161</td>
<td>5.12161</td>
<td>5.12161</td>
<td>9.31529 × 10⁻¹⁵</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>5.12161</td>
<td>5.12161</td>
<td>5.12161</td>
<td>6.96696 × 10⁻¹³</td>
</tr>
<tr>
<td></td>
<td>CGPR</td>
<td>5.12161</td>
<td>5.12161</td>
<td>5.12161</td>
<td>1.00829 × 10⁻¹²</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>5.12161</td>
<td>5.12161</td>
<td>5.12161</td>
<td>7.07957 × 10⁻¹⁴</td>
</tr>
<tr>
<td></td>
<td>CGHS</td>
<td>5.12161</td>
<td>5.12161</td>
<td>5.12161</td>
<td>6.02599 × 10⁻¹³</td>
</tr>
<tr>
<td>(4,0) nanotube</td>
<td>L-BFGS</td>
<td>31.19107</td>
<td>31.19176</td>
<td>31.19167</td>
<td>1.93878 × 10⁻⁴</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>31.19107</td>
<td>31.19176</td>
<td>31.19169</td>
<td>1.98526 × 10⁻⁴</td>
</tr>
<tr>
<td></td>
<td>CGPR</td>
<td>31.19176</td>
<td>31.19176</td>
<td>31.19176</td>
<td>9.44383 × 10⁻¹¹</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>31.19107</td>
<td>31.19176</td>
<td>31.19154</td>
<td>3.14052 × 10⁻⁴</td>
</tr>
<tr>
<td></td>
<td>CGHS</td>
<td>31.19176</td>
<td>31.19176</td>
<td>31.19176</td>
<td>8.10078 × 10⁻¹²</td>
</tr>
</tbody>
</table>

Continued on next page
Table A.1 – *Continued from previous page*

<table>
<thead>
<tr>
<th>System</th>
<th>Solver</th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>St. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>L-BFGS</td>
<td>3.89326</td>
<td>3.89326</td>
<td>3.89326</td>
<td>$7.04501 \times 10^{-11}$</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>3.89326</td>
<td>3.89326</td>
<td>3.89326</td>
<td>$4.48737 \times 10^{-10}$</td>
</tr>
<tr>
<td>Graphene</td>
<td>CGPR</td>
<td>3.89326</td>
<td>3.89326</td>
<td>3.89326</td>
<td>$4.93482 \times 10^{-10}$</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>3.89326</td>
<td>3.89326</td>
<td>3.89326</td>
<td>$8.32480 \times 10^{-11}$</td>
</tr>
<tr>
<td></td>
<td>CGHS$^a$</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>L-BFGS</td>
<td>8.25633</td>
<td>8.25633</td>
<td>8.25633</td>
<td>$9.34466 \times 10^{-13}$</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>8.25633</td>
<td>8.25633</td>
<td>8.25633</td>
<td>$3.82389 \times 10^{-12}$</td>
</tr>
<tr>
<td>h-BN</td>
<td>CGPR</td>
<td>8.25633</td>
<td>8.25633</td>
<td>8.25633</td>
<td>$2.21896 \times 10^{-12}$</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>8.25633</td>
<td>8.25633</td>
<td>8.25633</td>
<td>$9.41174 \times 10^{-13}$</td>
</tr>
<tr>
<td></td>
<td>CGHS</td>
<td>8.25633</td>
<td>8.25633</td>
<td>8.25633</td>
<td>$1.06368 \times 10^{-12}$</td>
</tr>
<tr>
<td></td>
<td>L-BFGS</td>
<td>1.69834</td>
<td>1.69834</td>
<td>1.69834</td>
<td>$1.36862 \times 10^{-13}$</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>1.69834</td>
<td>1.69834</td>
<td>1.69834</td>
<td>$1.51313 \times 10^{-9}$</td>
</tr>
<tr>
<td>LiH</td>
<td>CGPR</td>
<td>1.69834</td>
<td>1.69834</td>
<td>1.69834</td>
<td>$1.09185 \times 10^{-9}$</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>1.69834</td>
<td>1.69834</td>
<td>1.69834</td>
<td>$8.55046 \times 10^{-13}$</td>
</tr>
<tr>
<td></td>
<td>CGHS</td>
<td>1.69834</td>
<td>1.69834</td>
<td>1.69834</td>
<td>$6.92907 \times 10^{-10}$</td>
</tr>
</tbody>
</table>

*Continued on next page*
Table A.1 – *Continued from previous page*

<table>
<thead>
<tr>
<th>System</th>
<th>Solver</th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>St. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diamond</td>
<td>L-BFGS</td>
<td>4.13437</td>
<td>4.13437</td>
<td>4.13437</td>
<td>1.82115 × 10^{-12}</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>4.13437</td>
<td>4.13437</td>
<td>4.13437</td>
<td>2.66713 × 10^{-12}</td>
</tr>
<tr>
<td></td>
<td>CGPR</td>
<td>4.13437</td>
<td>4.13437</td>
<td>4.13437</td>
<td>2.22468 × 10^{-12}</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>4.13437</td>
<td>4.13437</td>
<td>4.13437</td>
<td>3.01420 × 10^{-12}</td>
</tr>
<tr>
<td></td>
<td>CGHS^a</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Silicon</td>
<td>L-BFGS</td>
<td>12.17338</td>
<td>12.17338</td>
<td>12.17338</td>
<td>1.20008 × 10^{-6}</td>
</tr>
<tr>
<td></td>
<td>CG</td>
<td>12.17338</td>
<td>12.17338</td>
<td>12.17338</td>
<td>7.02816 × 10^{-7}</td>
</tr>
<tr>
<td></td>
<td>CGPR</td>
<td>12.17338</td>
<td>12.17338</td>
<td>12.17338</td>
<td>1.70569 × 10^{-7}</td>
</tr>
<tr>
<td></td>
<td>CGFR</td>
<td>12.17338</td>
<td>12.17338</td>
<td>12.17338</td>
<td>9.91283 × 10^{-8}</td>
</tr>
<tr>
<td></td>
<td>CGHS</td>
<td>12.17338</td>
<td>12.17338</td>
<td>12.17338</td>
<td>2.38640 × 10^{-7}</td>
</tr>
</tbody>
</table>

^a All five calculations failed to converge.

**A.2 Number of Iterations to Solution vs. BFGS Parameters**

For each system, we have tabulated the number of iterations to solution for every BFGS solver; this data is organized by BFGS parameter pair. This should allow the interested
reader to recreate the analysis in Table 3.3 of the manuscript.

Table A.2: The number of iterations to solution for every parameter pair of the BFGS solver.

<table>
<thead>
<tr>
<th>⟨ # of SA, History ⟩</th>
<th>trans-(C₂H₄)∞</th>
<th>(C₂H₄)∞</th>
<th>(4,0) nanotube</th>
<th>Graphene</th>
<th>h-BN</th>
<th>LiH</th>
<th>Diamond</th>
<th>Silicon</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1, 1)</td>
<td>40</td>
<td>24</td>
<td>96</td>
<td>150</td>
<td>50</td>
<td>10</td>
<td>23</td>
<td>24</td>
</tr>
<tr>
<td>(1, 2)</td>
<td>41</td>
<td>30</td>
<td>101</td>
<td>96</td>
<td>57</td>
<td>11</td>
<td>24</td>
<td>22</td>
</tr>
<tr>
<td>(1, 5)</td>
<td>38</td>
<td>26</td>
<td>98</td>
<td>72</td>
<td>57</td>
<td>11</td>
<td>24</td>
<td>27</td>
</tr>
<tr>
<td>(1, 10)</td>
<td>44</td>
<td>27</td>
<td>95</td>
<td>55</td>
<td>56</td>
<td>11</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>(1, 15)</td>
<td>38</td>
<td>27</td>
<td>98</td>
<td>56</td>
<td>55</td>
<td>11</td>
<td>25</td>
<td>29</td>
</tr>
<tr>
<td>(2, 1)</td>
<td>40</td>
<td>26</td>
<td>98</td>
<td>186</td>
<td>52</td>
<td>10</td>
<td>23</td>
<td>21</td>
</tr>
<tr>
<td>(2, 2)</td>
<td>37</td>
<td>26</td>
<td>93</td>
<td>76</td>
<td>47</td>
<td>11</td>
<td>23</td>
<td>23</td>
</tr>
<tr>
<td>(2, 5)</td>
<td>38</td>
<td>26</td>
<td>86</td>
<td>63</td>
<td>51</td>
<td>11</td>
<td>24</td>
<td>25</td>
</tr>
<tr>
<td>(2, 10)</td>
<td>44</td>
<td>27</td>
<td>92</td>
<td>55</td>
<td>50</td>
<td>11</td>
<td>24</td>
<td>23</td>
</tr>
<tr>
<td>(2, 15)</td>
<td>37</td>
<td>27</td>
<td>87</td>
<td>54</td>
<td>50</td>
<td>11</td>
<td>25</td>
<td>23</td>
</tr>
<tr>
<td>(5, 1)</td>
<td>41</td>
<td>30</td>
<td>77</td>
<td>180</td>
<td>56</td>
<td>12</td>
<td>23</td>
<td>22</td>
</tr>
<tr>
<td>(5, 2)</td>
<td>38</td>
<td>27</td>
<td>76</td>
<td>109</td>
<td>52</td>
<td>12</td>
<td>23</td>
<td>22</td>
</tr>
<tr>
<td>(5, 5)</td>
<td>39</td>
<td>27</td>
<td>78</td>
<td>75</td>
<td>51</td>
<td>13</td>
<td>24</td>
<td>22</td>
</tr>
<tr>
<td>(5, 10)</td>
<td>44</td>
<td>28</td>
<td>81</td>
<td>56</td>
<td>51</td>
<td>13</td>
<td>25</td>
<td>24</td>
</tr>
<tr>
<td>(5, 15)</td>
<td>38</td>
<td>29</td>
<td>80</td>
<td>54</td>
<td>51</td>
<td>13</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>(10, 1)</td>
<td>45</td>
<td>30</td>
<td>78</td>
<td>86</td>
<td>45</td>
<td>17</td>
<td>24</td>
<td>22</td>
</tr>
<tr>
<td>(10, 2)</td>
<td>39</td>
<td>31</td>
<td>75</td>
<td>100</td>
<td>47</td>
<td>17</td>
<td>24</td>
<td>23</td>
</tr>
<tr>
<td>(10, 5)</td>
<td>46</td>
<td>31</td>
<td>77</td>
<td>53</td>
<td>45</td>
<td>17</td>
<td>25</td>
<td>23</td>
</tr>
<tr>
<td>(10, 10)</td>
<td>45</td>
<td>31</td>
<td>82</td>
<td>54</td>
<td>48</td>
<td>17</td>
<td>25</td>
<td>24</td>
</tr>
<tr>
<td>(10, 15)</td>
<td>41</td>
<td>32</td>
<td>80</td>
<td>55</td>
<td>48</td>
<td>18</td>
<td>27</td>
<td>25</td>
</tr>
<tr>
<td>(15, 1)</td>
<td>47</td>
<td>37</td>
<td>93</td>
<td>160</td>
<td>47</td>
<td>21</td>
<td>27</td>
<td>25</td>
</tr>
<tr>
<td>(15, 2)</td>
<td>41</td>
<td>34</td>
<td>83</td>
<td>122</td>
<td>50</td>
<td>21</td>
<td>26</td>
<td>24</td>
</tr>
<tr>
<td>(15, 5)</td>
<td>40</td>
<td>32</td>
<td>81</td>
<td>74</td>
<td>48</td>
<td>21</td>
<td>26</td>
<td>25</td>
</tr>
<tr>
<td>(15, 10)</td>
<td>41</td>
<td>32</td>
<td>83</td>
<td>56</td>
<td>49</td>
<td>21</td>
<td>27</td>
<td>25</td>
</tr>
<tr>
<td>(15, 15)</td>
<td>41</td>
<td>33</td>
<td>85</td>
<td>56</td>
<td>50</td>
<td>22</td>
<td>28</td>
<td>26</td>
</tr>
</tbody>
</table>
A.3 Initial Guess

To probe the sensitivity of the solver to the initial guess, the ⟨2,15⟩ BFGS PM solver was run on trans-(C₂H₂)∞ with 25 different initial guess scenarios:

1. 12 calculations using the CPR guess, each with a different seed.

2. 1 calculation using the “CP” guess (CPR without Randomization), in which, after phase canonicalization, an identity matrix was used to initialize every Uₖ matrix.

3. 12 calculations where each Uₖ matrix was initialized with a different seed.

Scenario (1) is what was used throughout the main text. Clearly, the phase canonicalization used in scenarios (1) and (2) leads to much greater starting values of P, thereby requiring fewer iterations to converge. The “k-nonuniform” random guesses [scenario (3)] were of much lower quality (smaller value of P) and resulted in slower convergence. However, the final value of P was the same, irrespective of the initial guess.
Table A.3: A statistical analysis of initial $P$, final $P$, and the number of iterations to solution vs. the initial guess type

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>St. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial $P$</td>
<td>1.20267281</td>
<td>1.35176477</td>
<td>1.27035531</td>
<td>$4.81965353 \times 10^{-2}$</td>
</tr>
<tr>
<td>CPR Final $P$</td>
<td>4.56798693</td>
<td>4.56798693</td>
<td>4.56798693</td>
<td>$1.48958067 \times 10^{-14}$</td>
</tr>
<tr>
<td>No. iters</td>
<td>37</td>
<td>42</td>
<td>39.2</td>
<td>1.90</td>
</tr>
<tr>
<td>Initial $P$</td>
<td>1.23188514</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>CP Final $P$</td>
<td>4.56798693</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>No. iters</td>
<td>37</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Initial $P$</td>
<td>0.17368364</td>
<td>0.19225600</td>
<td>0.18193346</td>
<td>$6.26283073 \times 10^{-3}$</td>
</tr>
<tr>
<td>Random Final $P$</td>
<td>4.56798693</td>
<td>4.56798693</td>
<td>4.56798693</td>
<td>$4.09819998 \times 10^{-13}$</td>
</tr>
<tr>
<td>No. iters</td>
<td>67</td>
<td>82</td>
<td>75.2</td>
<td>4.69</td>
</tr>
</tbody>
</table>
A.4 Minimal Basis Set Choice

We are also interested in how sensitive the solver is to the choice of minimal basis set used to compute the atomic charges. To this end, we have tabulated the initial $P$ value, the final $P$ value, and the number of iterations to solution for 16 different calculations. For each of the eight systems studied herein, we ran the $\langle 2,15 \rangle$ BFGS PM solver twice, once using the Huzinaga MINI basis set$^{171-173}$ for the minimal basis set and once using Knizia’s cc-pVTZ-derived MINAO basis$^{131}$. In all cases, the CPR guess with a seed of 1.0 was used.

In all cases, the MINI minimal basis set produced a slightly higher initial $P$ value than did MINAO, and in all but three cases (carbon nanotube, h-BN, and diamond), the MINI-based calculation converged in the same number or fewer iterations than the one that used the MINAO basis. The final $P$ value was slightly higher for the MINI calculations in all cases except for h-BN and diamond. Though the initial $P$ value is much less sensitive to the choice of the minimal basis than it is to the choice of the initial guess generating logic (CPR vs. CP vs. Random), the final $P$ value varies a fair bit more with the variation in minimal basis than it does with the variation in initial guess logic.

In looking at Table A.4, we see that LiH had the largest difference in initial $P$ value (though only the second smallest absolute difference in final $P$ value) with variation in the minimal basis. This led us to explore how the BFGS solver as applied to LiH would respond to the use of other minimal bases. In particular, we employed STO-3G and a variant of the MINAO basis where we maintained the $2p$ orbitals on Li found in the cc-pVTZ basis set$^{173,177}$ (we
Table A.4: Comparison of initial $P$, final $P$, and the number of iterations to solution vs. the minimal basis set employed

<table>
<thead>
<tr>
<th>System</th>
<th>Minimal Basis</th>
<th>Initial $P$</th>
<th>Final $P$</th>
<th>No. iters</th>
</tr>
</thead>
<tbody>
<tr>
<td>trans-$(\text{C}_2\text{H}<em>2)</em>{\infty}$</td>
<td>MINI</td>
<td>1.35176</td>
<td>4.56799</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>MINAO</td>
<td>1.34997</td>
<td>4.55664</td>
<td>38</td>
</tr>
<tr>
<td>$(\text{C}_2\text{H}<em>4)</em>{\infty}$</td>
<td>MINI</td>
<td>1.65401</td>
<td>5.12161</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td>MINAO</td>
<td>1.64906</td>
<td>5.12157</td>
<td>27</td>
</tr>
<tr>
<td>(4,0) nanotube</td>
<td>MINI</td>
<td>1.99298</td>
<td>31.19176</td>
<td>87</td>
</tr>
<tr>
<td></td>
<td>MINAO</td>
<td>1.98765</td>
<td>31.17120</td>
<td>84</td>
</tr>
<tr>
<td>Graphene</td>
<td>MINI</td>
<td>1.26844</td>
<td>3.89326</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td>MINAO</td>
<td>1.25791</td>
<td>3.88672</td>
<td>65</td>
</tr>
<tr>
<td>h-BN</td>
<td>MINI</td>
<td>1.06602</td>
<td>8.25633</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>MINAO</td>
<td>1.06308</td>
<td>8.28429</td>
<td>47</td>
</tr>
<tr>
<td>LiH</td>
<td>MINI</td>
<td>0.93803</td>
<td>1.69834</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>MINAO</td>
<td>0.89262</td>
<td>1.69277</td>
<td>11</td>
</tr>
<tr>
<td>Diamond</td>
<td>MINI</td>
<td>0.63132</td>
<td>4.13437</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>MINAO</td>
<td>0.62478</td>
<td>4.14208</td>
<td>24</td>
</tr>
<tr>
<td>Silicon</td>
<td>MINI</td>
<td>1.39184</td>
<td>12.17338</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>MINAO</td>
<td>1.38606</td>
<td>12.08619</td>
<td>24</td>
</tr>
</tbody>
</table>
are calling this “MINAO w/ 2p”). We compare the maximum overlap condition number, the initial $P$ value, the final $P$ value, and the number of iterations to solution for the four minimal bases (MINI, MINAO, MINAO w/ 2p, and STO-3G). This data can be found in Table A.5. The presence of the unoccupied valence $p$ orbital, far more than the maximum overlap condition number, seems to be the indicator of poor BFGS convergence behavior.

Table A.5: Comparison of maximum overlap condition number, initial $P$, final $P$, and the number of iterations to solution vs. the minimal basis set employed

<table>
<thead>
<tr>
<th>Minimal Basis</th>
<th>Max Condition Number</th>
<th>Initial $P$</th>
<th>Final $P$</th>
<th>No. iters</th>
</tr>
</thead>
<tbody>
<tr>
<td>STO-3G</td>
<td>114.53596</td>
<td>0.84045</td>
<td>1.21610</td>
<td>20</td>
</tr>
<tr>
<td>MINAO w/ 2p</td>
<td>3233.44358</td>
<td>0.73236</td>
<td>1.27476</td>
<td>19</td>
</tr>
<tr>
<td>MINAO</td>
<td>96.46795</td>
<td>0.89262</td>
<td>1.69277</td>
<td>11</td>
</tr>
<tr>
<td>MINI</td>
<td>102.88753</td>
<td>0.93803</td>
<td>1.69834</td>
<td>11</td>
</tr>
</tbody>
</table>
Appendix B

Publication List

Published:


Submitted:

Bibliography

(1) Dirac, P. A. M. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* 1929, 123, 714–733.


(23) Rayleigh, J., The Theory of Sound; The Theory of Sound v. 1; Macmillan: 1894.


(114) Peng, C.; Clement, M. C.; Valeev, E. F. Exploration of Reduced Scaling Formulation of Equation of Motion Coupled-Cluster Singles and Doubles Based on State-Averaged Pair Natural Orbitals, arXiv:1802.06738.
(115) Valeev, E. F.; Peng, C.; Lewis, C. A.; Calvin, J. A. The Massively Parallel Quantum chemistry Program (MPQC), Version 4.0.0.


(171) Van Duijneveldt, F. B. *Gaussian basis sets for the atoms H-Ne for use in molecular calculations*; tech. rep. RJ 945; IBM Research Laboratory, 1971.


