INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

HARVARD UNIVERSITY

THE GRADUATE SCHOOL OF ARTS AND SCIENCES

THESIS ACCEPTANCE CERTIFICATE

The undersigned, appointed by the

Division

Department of Physics

Committee

have examined a thesis entitled

"Adaptive Basis Approaches to Quantum Spin and Electronic Systems Using Parallel Computers"

presented by Normand Arthur Modine

candidate for the degree of Doctor of Philosophy and hereby certify that it is worthy of acceptance.

Signature .....

Typed name Efthimios Kaxiras, Chair

Signature .....

Typed name Bertrand I. Halperin

Signature .....

Typed name Daniel S. Fisher

Date June 12, 1996

Adaptive Basis Approaches to Quantum Spin and Electronic Systems Using Parallel Computers

A thesis presented

by

Normand Arthur Modine

to

The Department of Physics

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in the subject of

Physics

Harvard University

Cambridge, Massachusetts

June 1996

Abstract

Most computational methods do not adapt to the particular system that is being studied. We discuss the addition of adaptability to simulations of quantum spin models and electronic systems. First, we study the spin-$\frac{1}{2}$ Heisenberg antiferromagnet on a series of finite-size clusters with and without frustration. We use exact diagonalization combined with a truncation method in which only the most important basis states of the Hilbert space are retained. We describe an efficient variational method for finding an optimal truncation which minimizes the error in the ground state energy. Ground state energies and spin-spin correlations are obtained for clusters with up to thirty-two sites without recourse to symmetry. Next, we develop adaptive coordinate real-space electronic structure computations. A regular real-space mesh produces a sparse, local, and structured Hamiltonian, which enables effective use of iterative algorithms and parallel computers. However, a regular real space mesh can not be adapted to a particular physical system. To remedy this inefficiency without losing the advantages of a regular mesh, we use a *regular* mesh in *curvilinear* space, which is mapped by a change of coordinates to an *adaptive* mesh in *real* space. We report all-electron calculations for atoms and molecules with $1s$ and $2p$ valence electrons, and pseudopotential calculations for molecules and solids. Then, we present an eigensolver based on inverse iteration that efficiently finds a few eigenvalues and eigenvectors of a large sparse matrix. The core of the method is an algorithm that converges to eigenstates located within a given energy range, while strongly suppressing effects from states outside the energy range. The algorithm avoids global orthogonalization and has a convergence rate that does not depend on the width of the spectrum. We discuss implementation in the context of large scale electronic structure computations. Finally, we discuss adaptive coordinate real-space electronic structure computations for the Si surface. Parallel computers, which have sufficient power to allow the study of systems previously beyond the reach of simulation, are used for all computations, and details are included when unusual techniques are required.

Acknowledgments

Foremost, I would like to thank Tim Kaxiras, whose inexhaustible patience and endless enthusiasm pulled me kicking and screaming from the depths of despair through to a completed thesis.

A great deal of credit should also go to my collaborator Gil Zumbach. Without him, the work on ACRES and inverse iteration would have been impossible. I would also like to thank him for teaching me to write respectable computer code instead of unreadable garbage.

I would like to acknowledge Bert Halperin and Daniel Fisher for guiding me through my first several years at Harvard and for serving on my committee. They were willing to make the tough choices that ultimately proved best for me.

I would like to thank the numerous scientists who have spent hours of their time patiently explaining their work, their insights, and their ideas to me in the hallways at conferences and through their talks and papers. Some of these people are acknowledged in the individual chapters that make up this thesis, but there are many more who also deserve credit.

Equally, I would like to credit my fellow students at Harvard, both in Lyman 422 and in the Kaxiras group. I think that I learned more physics from them than from anyone else.

Finally, I would like to thank my mother and father, Linda and Frank Modine. Without their unwavering support, I would have given up long ago.

Citations to Previously Published Work

Much of Chapters 2 and 3 has previously appeared in the following published papers:

Chapter 2: N. A. Modine and E. Kaxiras, Phys. Rev. B, **53**, 2546 (1996).

Chapter 3: N. A. Modine, G. Zumbach, and E. Kaxiras, in *Materials Research Society Proceedings*, edited by E. Kaxiras, J. Joannopoulos, P. Vashishta, and R. K. Kalia (Materials Research Society, Pittsburgh, 1996), Vol. 408., p. 139

"Adaptive Coordinate, Real-space Electronic Structure Calculations On Parallel Computers", G. Zumbach, N. A. Modine, and E. Kaxiras, Solid State Comm. (in press).

Contents

Title Page 1

Abstract 2

Acknowledgments 3

Citations to Previously Published Work 4

Table of Contents 5

List of Figures 7

1 Introduction **15**

1.1 Introduction 15

1.2 Variational Hilbert Space Truncation 16

1.3 ACRES: Adaptive Coordinate, Real-space Electronic Structure 18

1.4 Inverse Iteration 21

1.5 Applications 23

2 Variational Hilbert Space Truncation Approach to Quantum Heisenberg Antiferromagnets on Frustrated Clusters **24**

2.1 Introduction 24

2.2 Choice of Truncation 30

2.3 Implementation of the Method in a Massively Parallel Architecture . . . 39

2.3.1 Use of previous results 40

2.3.2 Load balancing 41

2.3.3 Sorts instead of searches 41
2.4 Results 42
2.5 Conclusion 47

3 ACRES: Adaptive Coordinate, Real-Space Electronic Structure Calculations for Atoms, Molecules, and Solids **49**
3.1 Introduction 49
3.1.1 Desirable properties 50
3.1.2 Real space methods 54
3.1.3 ACRES 57
3.2 Theory 58
3.3 Implementation 59
3.3.1 The Laplacian 60
3.3.2 The Coulomb potential 64
3.3.3 The coordinate transformation 68
3.3.4 Optimization of the change of coordinates 72
3.3.5 The forces 84
3.3.6 Band structure calculations 87
3.3.7 The algorithms 88
3.4 Results 91
3.5 Conclusion 97
3.6 Acknowledgments 97

4 Inverse iteration eigensolver with applications to efficient Kohn-Sham electronic structure computations **98**
4.1 Introduction 98
4.2 Inverse Iteration 101
4.3 Modified Inverse Iteration 103

4.3.1 Scanning for the spectrum 103
4.3.2 Solving the linear system 107
4.4 Implementation Details 109
4.5 Optimized Inverse Iteration 111
4.5.1 Multigrid preconditioned conjugate gradient 111
4.5.2 Subspace diagonalization 113
4.5.3 Subspace preconditioning 115
4.6 Results 115
4.7 Conclusions 118

5 Applications **120**

5.1 Introduction 120
5.2 Reconstruction of the Silicon Surface 121
5.3 Oxygen Absorption on the Silicon Surface 123
5.4 ACRES Results 124
5.4.1 Symmetric Dimer 124
5.4.2 Buckled Dimer 126
5.4.3 Oxygenated Dimer 129
5.5 Discussion 129

6 Conclusion and Prospects **133**

Bibliography **136**

List of Figures

2.1 (a) 20-site, (b) 24-site, (c) 26-site, (d) 28-site, and (e) 32-site frustrated structures with spherical topology. The 3-dimensional structures are shown projected on a plane, which introduces a distortion of relative distances. Therefore, the figures only indicate the connectivity of the the structures, and the apparent lengths of the bonds are not to be interpreted literally. 26

2.2 (a) 18-site, (b) 24-site, and (c) 26-site unfrustrated structures derived from the honeycomb lattice by applying periodic boundary conditions along the thinner solid lines. 28

2.3 Error in the ground state energy resulting from truncation of the Hilbert space for (a) some frustrated structures and (b) some unfrustrated structures. Lines indicate the results of truncating the Hilbert space based on the weights of states in the full space solution. Individual points indicate results from the variational method and correspond to the same structure as the line immediately above them. The stars indicate results for the 28 site system, where results from truncating based on the full-space solution are unavailable. 35

2.4 Spin-spin correlations for the unfrustrated structures based on the honeycomb lattice as a function of the fraction of states retained in the truncated Hilbert space. Lines indicate results from truncating the Hilbert space based on the weights of states in the full-space solution. Individual points indicate results from the variational method. The groups of lines are labeled by nearest neighbor distances. The points near zero abscissa correspond to small, but finite, truncations. 37

2.5 Fractional error in the smallest spin-spin correlations on the frustrated structures as a function of the fraction of states retained in truncated Hilbert space. Lines indicate results from truncating the Hilbert space based on the weights of states in the full-space solution. Individual points indicate results from the variational method. 38

3.1 Schematic of a local refinement enhancement of the resolution in a difficult region near the center of the box. The intersections between the extra lines could indicate an added level of wavelets with a wavelet method, additional finite elements nodes in with finite element approach, or literally extra grid points with a multigrid method. 56

3.2 Schematic of the ACRES idea. The computations are done on a regular mesh in curvilinear coordinates, which is mapped by a change of variables to an adaptive mesh in real space. 57

3.3 Convergence of the total energy as a function of the amount of adaptation for the first, second, and third order discretized derivatives. Results are from an all-electron calculation for a H atom using a grid with $32 \times 32 \times 32$ points. 62

3.4 Convergence of the total energy as a function of the amount of adaptation for the first, second, and third order discretized derivatives. Results are from a pseudopotential calculation for a H atom using a grid with $32 \times 32 \times 32$ points. 63

3.5 Error in the total energy of a hydrogen atom as a function of the amount of adaptation for all-electron calculations using a Gaussian delta function with various values of σ. The points marked by circles result from placing the atom directly on a grid point (zero offset), while the points marked by squares result from placing the atom as far as possible from a grid point (maximum offset). The supercell is 12 a.u. on each side, the grid has 64 points in each direction, no backdrop is used, and the adaptation radius is 0.5 a.u. 67

3.6 Backdrop used for a pseudopotential calculation for an O_2 molecule in a $12 \times 12 \times 24$ a.u. box., in a horizontal cross-section through the atoms (every fourth line shown). The small dots indicate the locations of the atoms. The backdrop parameters are $\bar{x}_1 = 6$ a.u., $\bar{x}_2 = \bar{x}_3 = 3$ a.u., and $1/\bar{J}_i = 2$ for $i = 1 - 3$. 69

3.7 Grid used for a pseudopotential calculation for an O_2 molecule in a $12 \times 12 \times 24$ a.u. box., in a horizontal cross-section through the atoms (every fourth line shown). Notice the effect of the global backdrop (crosslike region with many grid points) and the local adaptation around each atom. The backdrop parameters are $\bar{x}_1 = 6$ a.u., $\bar{x}_2 = \bar{x}_3 = 3$ a.u., and $1/\bar{J}_i = 2$ for $i = 1 - 3$. For each atom, the local adaptation parameters are $1/|J|_\nu = 32$ and $\kappa_\nu = 1.4$ a.u. The spacing between the atoms is 2.28 a.u. 73

3.8 The error in the total energy of an O atom as a function of the amount of adaptation for three different adaptation radii κ. The results are from a pseudopotential calculation using a $48 \times 48 \times 48$ grid and a box 6 a.u. on each side. No backdrop was used. The exact large N limit of the energy was approximated using a $96 \times 96 \times 96$ grid. 74

3.9 The contributions to an integral from an elementary region of space evaluated using a regular grid with $N/2$ points and a regular grid with N points. $\Delta\xi$ indicates the grid spacing. 77

3.10 Merit functional $m(f; P)$ as a function of the amount of adaptation for various arguments f. The adaptation radius κ is fixed at 0.4 a.u. The results are from a pseudopotential calculation for an O atom in a $6 \times 6 \times 6$ a.u. box using a $48 \times 48 \times 48$ grid. The functions f are: e_{band}, the band energy density; e_{pot}, the potential energy density; e_{psp}, the pseudopotential energy density; ρ, the electronic charge density; $|\Psi_s|^2$, a (squared) s wavefunction; and $|\Psi_p|^2$, a (squared) p wavefunction. $m(e_{pot})$ has been divided by a factor of 20 in order to fit it on the scale of the other functionals. 80

3.11 Merit functional $m(f; P)$ as a function of the amount of adaptation for various arguments f. The adaptation radius κ is fixed at 1.2 a.u. The results are from a pseudopotential calculation for an O atom in a $6 \times 6 \times 6$ a.u. box using a $48 \times 48 \times 48$ grid. The functions f are: e_{band}, the band energy density; e_{pot}, the potential energy density; e_{psp}, the pseudopotential energy density; ρ, the electronic charge density; $|\Psi_s|^2$, a (squared) s wavefunction; and $|\Psi_p|^2$, a (squared) p wavefunction. $m(e_{pot})$ has been divided by a factor of 20 in order to fit it on the scale of the other functionals. 81

3.12 The forces from an all-electron computation for H_2 with and without Pulay corrections. The derivative of the energy is included for comparison. The box size was $24 \times 12 \times 12$ a.u. and a $64 \times 32 \times 32$ grid was used. The backdrop parameters were $\bar{x}_1 = 4$ a.u., $\bar{x}_2 = \bar{x}_3 = 2$ a.u., and $1/\bar{J}_i = 4$ for $i = 1 - 3$. For each atom, the local adaptation parameters are $1/|J|_\nu = 64$ and $\kappa_\nu = 0.5$ a.u. A Gaussian delta with $\sigma = 0.6$ was used. 86

3.13 The bands structure of diamond structure Si unfolded from the results for a 4 atom unit cell. 30 inequivalent k-points were used to sample the Brillouin zone. Once self-consistency was achieved, calculations with a fixed density were done for the Brillouin zone locations indicated in the figure. A $32 \times 32 \times 48$ grid was used without a backdrop. For each atom, the local adaptation parameters were $1/|J|_\nu = 8$ and $\kappa_\nu = 0.9$ a.u. . . . 89

3.14 Occupied wave functions of the O_2 molecule, along a line through the centers of the atoms. The π bonding and anti-bonding wave functions collapse onto the horizontal axis (they have nodes through the atomic centers). The $1s$ bonding and anti-bonding states were scaled by a factor of $1/3$ so they could be displayed on the same scale. Points on the curves indicate values at actual grid points used in the calculation. 93

3.15 The convergence with increasing grid size of physical properties of a H_2 molecule. A $12 \times 12 \times 24$ a.u. box was used. The grids have the indicated number of points in their short directions and twice as many in their long direction. The results from the $128 \times 128 \times 256$ grid were taken to be exact. 96

5.1 The charge density for a symmetric dimer reconstruction of the Si (100) surface shown in a cross-section through the center of the dimer bond. The size of the unit cell has been doubled in the direction parallel to the dimer in order to make it easier to see the dimer. 8 layers of Si atoms and 1 layer of surface terminating H atoms are included in the calculation, but only 4 layers of Si atoms and the H atoms can be seen in this cross-section. There are 2 additional rows of Si atoms located out of the plane of the picture between each of the sets of rows that are visible. The H atoms appear as bright spots because they have a much higher charge density than the Si atoms. 125

5.2 The charge density for a buckled dimer reconstruction of the Si (100) surface shown in a cross-section through the center of the dimer bond. We have concentrated on the interesting region near the surface by including only 5 layers of Si atoms and 1 layer of surface terminating H atoms in the calculation. Only 3 layers of Si atoms can be seen in this cross-section. There are 2 additional rows of Si atoms located out of the plane of the picture between the dimer and the top rows. The four H atoms that terminate the bonds on the bottom row of atoms are also located out of the plane with two atoms in front of the plane of the figure and and two behind. 128

5.3 The charge density for a symmetric dimer reconstruction of the Si (100) surface with a half monolayer of oxygen inserted into the dimer bonds. The figure shows a cross-section through the center of the dimer bond. The size of the unit cell has been doubled in the direction parallel to the dimer in order to make it easier to see the dimer. The oxygen, 8 layers of Si atoms and 1 layer of surface terminating H atoms are included in the calculation. There are 2 rows of Si atoms located out of the plane of the picture between each of the sets of rows that are visible. The O atom appears as a very bright spot due to its very large charge density relative to other atoms, even the H atoms. 130

5.4 The grid used for the dimer with oxygen shown in cross-section through the center of the dimer bond. The size of the unit cell has been doubled in the direction parallel to the dimer in order to make it easier to see the dimer. 131

Chapter 1

Introduction

1.1 Introduction

The human view of the physical world is intrinsically adaptive. Details are rarely noticed unless they are important or unusual. A person looking at a curve instantly notices any sharp cusps without having to examine the entire line closely, even when the unusual behavior only extends over a very short distance. More abstractly, an object is observed to have a definite position even though its wavefunction includes the possibility that it just tunneled through the floor. Human observation automatically truncates unimportant regions of the wavefunction. Even understanding, in general, and the pursuit of physics, in particular, could be viewed as nothing but the abstraction of general patterns from a sea of unimportant details. However, no matter how easily the human mind picks out important information from a constant deluge of input, it is not easy to convince a computer to do the same thing. The power of computers and the complexity of the operations that they perform has been consistently growing at an exponential rate with a doubling time of about 18 months. A significant step in this continuous evolution has been the recent development of massively parallel computers. Like a human brain, these computers have many processors, which work cooperatively

on a given task. As is the case in many fields, the growing capabilities of computers have led to their growing use in physics research. As computers become an increasingly important part of physics research, their inability to select the relevant variables for a particular situation presents an important obstacle. In the following thesis, we show how a degree of adaptability can be added to the computational treatment of quantum spin systems and electronic structure calculations. First, we discuss a variational Hilbert space truncation approach to quantum magnets. Then, we report an implementation of adaptive coordinate, real-space electronic structure calculations. This will be followed by a description of an inverse iteration eigensolver with applications to efficient electronic structure calculations. Finally, we will discuss some applications of the last two topics to the study of semiconductor surface reconstructions and impurities.

1.2 Variational Hilbert Space Truncation

Quantum lattice models such as the Heisenberg antiferromagnet have enjoyed long lasting popularity with theoretical physicists. At a fundamental level, it is hoped that an understanding of such simplified models will improve understanding of the general behavior of systems with many interacting quantum degrees of freedom. At a practical level, it is believed that the low energy, long wavelength behavior of some models may be identical to the low energy, long wavelength behavior of real systems such as magnetic materials, superconductors, or heavy fermion compounds. The fullerenes are spherical shells of threefold coordinated carbon atoms arranged in pentagonal and hexagonal rings. Doped fullerene crystals are observed to become superconducting at an unusually high temperature. Each carbon atom of a fullerene has a dangling bond occupied by a single electron. If we limit our degrees of freedom to these dangling bonds and their electrons, and if we keep only the shortest range interactions, we obtain an effective model known as the Hubbard model. The Hubbard model combines a tight binding like hopping between nearest neighbor orbitals with an on-site interaction between electrons

located in the same orbital. If we assume that that the on-site interaction is repulsive and large compared to the hopping matrix element, we can further simplify the model by limiting the considered space to states with a minimal number of doubly occupied orbitals. This results in the $t - J$ model, which consists of an antiferromagnetic background plus a fixed number of holes or doubly occupied sites that move through the background by a nearest neighbor hopping. The antiferromagnetic background arises because an antiferromagnetic arrangement of the electronic spins allows virtual hops of each electron to the neighboring sites, and thus lowers the energy relative to a ferromagnetic arrangement. In the case, of half-filling (one electron per orbital), there are no holes or doubly occupied orbitals, and the system acts like a pure antiferromagnet. The charge degrees of freedom are frozen out by the repulsive on-site interaction, and the only remaining degrees of freedom are the electronic spins.

We study the spin-$\frac{1}{2}$ Heisenberg antiferromagnet on a series of finite-size clusters with features inspired by the fullerenes. The pentagonal rings prevent each spin from pointing opposite to its neighbors. This frustration makes such structures challenging in the context of quantum Monte-Carlo methods. The growing power of computers has made practical the exact diagonalization of the Hamiltonian of systems with up to a few dozen degrees of freedom. Exact diagonalization provides important checks on other approaches and useful clues about unknown physics because its results are not biased toward any particular outcome. However, the size of the systems that can be handled is limited because the size of the Hilbert space (and thus the size of the matrix that must be diagonalized) grows exponentially in the size of the system. Therefore, the goal is to develop an adaptive method that includes only important states in the calculation without biasing the answer. Although human observation of a real system manages to select the important parts of a wavefunction by some means that is not fully understood, typically there is no way that a computer can tell beforehand which states will be important. We describe an efficient variational method for finding an optimal

truncation of a given size which minimizes the error in the ground state energy. Ground state energies and spin-spin correlations are obtained for clusters with up to thirty-two sites without the need to restrict the symmetry of the structures. The results are compared to full-space calculations and to unfrustrated structures based on the honeycomb lattice.

1.3 ACRES: Adaptive Coordinate, Real-space Electronic Structure

Within the Born-Oppenheimer approximation, the electronic state of an atomic, molecular, or condensed matter system is described by the ionic positions $\vec{R}_1, \ldots, \vec{R}_n$ and the many body electronic wavefunction $\Psi(\vec{r}_1, \ldots, \vec{r}_N)$. The time evolution of $\Psi(\vec{r}_1, \ldots, \vec{r}_N)$ is determined by the Hamiltonian

$$H = \sum_i -\nabla_i^2 + \sum_{i \neq j} \frac{1}{|\vec{r}_i - \vec{r}_j|} + \sum_i V_{ion}(\vec{r}_i) \tag{1.1}$$

where ∇_i refers to differentiation with respect to $\vec{r}_i$, and V_{ion} describes the effect of the ions on an electron. In an all-electron calculation, V_{ion} consists of a Coulomb potential centered on each ion, and N is the total number of electrons in the system. In a pseudopotential calculation, V_{ion} includes the effect of some core electrons, which are taken to be frozen, and N includes only the remaining electrons. For a particular arrangement of the ions, a very important case is the electronic ground state of the system, which is obtained when $\Psi(\vec{r}_1, \ldots, \vec{r}_N)$ is taken to be the eigenvector of H with the smallest eigenvalue. Since electrons usually relax several order of magnitude faster than ions, many properties of physical systems at typical temperatures can be predicted by studying their electronic ground states. Of central interest is the smallest eigenvalue of the Hamiltonian E_0, which gives the energy of a given configuration of ions when the electrons are fully relaxed.

Because Ψ is a function of a $3N$ dimensional space, a direct calculation of the

ground state is very difficult for $N > 2$. The density functional approach to electronic structure replaces this intractable problem with the much simpler problem of finding N single-body fictitious wavefunctions, which produce the same density as the real electrons. The density functional approach is based on the variational formulation of the eigenvalue problem. This formulation requires that

$$E_0 = \min_{\Psi} \langle \Psi | H | \Psi \rangle \tag{1.2}$$

where the minimization is over all states Ψ that are antisymmetric with respect to exchange of the electrons. If we define the electronic density

$$\rho(\vec{r}) = \int |\Psi(\vec{r}, \vec{r}_1, \ldots, \vec{r}_{N-1})|^2 d^3\vec{r}_1 \ldots d^3\vec{r}_{N-1}, \tag{1.3}$$

we can rewrite Eq. (1.2) as

$$E_0 = \min_{\rho} \left\{ \min_{\Psi \to \rho} \langle \Psi | H | \Psi \rangle \right\}, \tag{1.4}$$

where the outer minimization is over densities ρ that are produced by any Ψ, and the inner minimization is over wavefunctions Ψ that produce the density ρ. Then, defining a universal functional of the density

$$F[\rho] = \min_{\Psi \to \rho} \langle \Psi | \sum_i -\nabla_i^2 + \sum_{i \neq j} \frac{1}{|\vec{r}_i - \vec{r}_j|} | \Psi \rangle, \tag{1.5}$$

we obtain

$$E_0 = \min_{\rho} \left\{ F[\rho] + \int V_{ion}(\vec{r}) \rho(\vec{r}) d^3\vec{r} \right\}. \tag{1.6}$$

This procedure was developed and shown to be well defined in 1964 by Hohenberg and Kohn [1]. In 1965, Kohn and Sham [2] suggested defining a new functional $E_{XC}[\rho]$ and expanding $F[\rho]$ as

$$F[\rho] = \tilde{T}[\rho] + \frac{1}{2} \int \frac{\rho(\vec{r}) \rho(\vec{r}')}{|\vec{r} - \vec{r}'|} d^3\vec{r} d^3\vec{r}' + E_{XC}[\rho] \tag{1.7}$$

where $\tilde{T}$ is the kinetic energy of a set of N noninteracting 'electrons' located in a potential $\tilde{V}$ chosen such that the total density is ρ. The motivation for this expansion

is that if the kinetic energy of the real interacting electrons is similar to $\tilde{T}$, then the exchange-correlation energy E_{XC} should be small, and approximating E_{XC} should have a small effect on E_0. Kohn and Sham [2] also showed that if

$$\tilde{V}(\vec{r}) = \int \frac{\rho(\vec{r}')d^3\vec{r}'}{|\vec{r} - \vec{r}'|} + \frac{\delta E_{XC}[\rho]}{\delta\rho(\vec{r}} + V_{ion}(\vec{r}), \tag{1.8}$$

and if the lowest N eigenstates $\psi_1, \ldots, \psi_N$ of the Kohn-Sham Hamiltonian $-\nabla^2 + \tilde{V}$ produce a total density ρ, then ρ is the desired minimum in Eq. (1.6). In practice, this self consistency is achieved iteratively by choosing an initial ρ, calculating $\psi_1, \ldots, \psi_N$, computing the resulting density ρ', and updating ρ toward ρ'. The procedure is repeated until convergence is achieved.

Up to this point, the density functional approach has been an exact rewriting of the original problem and has involved no new approximations. However, all of the complicated exchange and correlation effects contained in Ψ have been swept into the unknown functional E_{XC}. Practical implementations of the method require approximating E_{XC} by a known functional. The most frequently used type of approximation for E_{XC} are the various local density approximations (LDA). In the LDA, at each point $\vec{r}$, $E_{XC}(\vec{r})$ is taken to be E_{XC} for a uniform electron gas with density $\rho(\vec{r})$. For the uniform electron gas, E_{XC} can be obtained from quantum Monte-Carlo simulations or perturbation expansions. Another type of approximation for E_{XC} are the generalized gradient approximations (GGA) in which corrections depending on the gradient of ρ at position $\vec{r}$ are added to the LDA. For both LDA and GGA, there are several different versions of the approximation that correspond to different calculations of E_{XC} for a uniform electron gas, different parameterizations of the results, and slightly different gradient corrections. The various versions of LDA and GGA can give slightly different answers, but in general electronic structure calculations based on density functional theory have been remarkably successful at predicting properties of real molecules and solids.

We describe our ACRES approach to density functional calculations, which has three desirable properties: sparsity, parallelizability, and adaptability. A sparse Hamil-

tonian enables effective use of iterative numerical methods, which allow large savings in computer time and memory. A natural mapping onto a parallel computer that assigns equivalent tasks to every processor and produces a structured communications pattern that is local makes it easy to obtain excellent parallel efficiency. The ability to adapt the resolution in different regions of space in such a way that only important details are represented allows efficient treatment of inhomogeneous problems such as all-electron computations, pseudopotential computations for atoms with $1s$, $2p$, $3d$, or $4f$ valence electrons, or systems with large regions of vacuum such as atoms, molecules, clusters, or solid surfaces. The ACRES method achieves these properties by calculating on a *regular* mesh in *curvilinear* space, which is mapped by a change of coordinates to an *adaptive* mesh in *real* space. The underlying regular mesh provides sparsity and parallelizability. The coordinate transformation preserves these properties while providing adaptability. There are several choices involved in the implementation of the method. These include the form and optimization of the coordinate transformation, the expression for the discretized Laplacian, the regularization of the ionic potential for all-electron calculations, the method of calculating the forces, and the algorithms used. Band structure calculations are implemented by adding a phase shift at periodic boundary conditions. We report all-electron calculations for atoms and molecules with $1s$ and $2p$ valence electrons, and pseudopotential calculations for molecules and solids.

1.4 Inverse Iteration

Since it is more efficient, an adaptive approach allows a higher maximum resolution than an approach that requires an equal level of resolution everywhere. A high resolution gives rise to high energy states in the spectrum of a typical operator, and therefore standard eigensolver algorithms may have unusually poor performance. In particular, this is a problem for the Lanczos algorithm, which is the standard iterative algorithm used to avoid an $O(n_e^2 N_b)$ orthogonalization cost when finding n_e eigenvectors in a space

with N_b basis states. We present an efficient iterative eigensolver based on inverse iteration that is suitable for finding a few eigenvalues and eigenvectors of a large sparse matrix. This eigensolver avoids global orthogonalization of the eigenfunctions and has a convergence rate that does not depend on the width of the spectrum. The core of the method is an algorithm that converges to eigenstates located within a given energy range, while strongly suppressing effects from states outside the energy range. Since eigenstates with different energies are automatically orthogonal, orthogonalization between different energy ranges is avoided. We discuss an implementation in the context of our ACRES method. The algorithm successfully resolves several important computational bottlenecks in these calculations. Aspects of our approach that are of particular interest include: (a) a scanning technique used to find the eigenvalues from little *a priori* knowledge about the spectrum; (b) systematic avoidance of strongly singular equations; (c) diagonalization within small subspaces to resolve nearly degenerate states; (d) use of a multigrid preconditioned conjugate gradient solver to handle long wavelength modes efficiently; and (e) use of the exact inverse Hamiltonian within small subspaces to precondition nearly singular systems generated by the inverse iteration. Although the algorithm was designed to avoid a problem that arose during adaptive calculations, it could itself be described as adaptive in the sense that it treats different time scales and length scales with different methods. The short time scales related to highly excited states in the spectrum are effectively handled with the inverse iteration approach, while the long time scale associated with the small differences between almost degenerate eigenvalues are quickly treated by standard noniterative diagonalization. Likewise, short length scales are rapidly converged using the conjugate gradient algorithm, while long length scales are efficiently handled by the multigrid preconditioning.

1.5 Applications

The processing of the surface of silicon is of great importance to the electronics industry, and therefore a better understanding of the phenomena that take place during this processing would be useful. Increasingly, investigations of such phenomena focus on microscopic properties and first principles electronic structure calculations are uniquely suited to the investigation of such properties. Because it can efficiently handle both large regions of vacuum and oxygen atoms, the ACRES method is especially well suited to the study the absorption of the oxygen at the silicon surface, which is the first step of the extremely important process of converting semiconducting Si to insulating SiO_2. Investigation of the absorption process is complicated by reconstruction of the Si surface and by the large number of possible paths that could be involved in the process. We present the results of ACRES calculations for tilted and untilted dimer reconstructions on the clean Si surface as well as an untilted dimer with an oxygen atom incorporated into the bond.

Chapter 2

Variational Hilbert Space Truncation Approach to Quantum Heisenberg Antiferromagnets on Frustrated Clusters

2.1 Introduction

The spin-$\frac{1}{2}$ Heisenberg antiferromagnet (HAFM) has long been studied as a simple example of a strongly interacting quantum many-body system [3]. Recently, it has attracted considerable attention in the context of the copper oxide high-temperature superconductors [4, 5]. The Hamiltonian of the HAFM is given by

$$H = J \sum_{\langle i,j \rangle} \vec{S}_i \cdot \vec{S}_j \equiv J \sum_{\langle i,j \rangle} S_i^z S_j^z + \frac{1}{2} \left(S_i^+ S_j^- + S_i^- S_j^+ \right) \tag{2.1}$$

where J takes positive values, $\langle i,j \rangle$ refers to nearest neighbor pairs, $\vec{S}_i$ is the spin operator for a spin-$\frac{1}{2}$ located at site i, and S_i^+ and S_i^- are the corresponding raising and lowering operators. The operator $S_i^+ S_j^- + S_i^- S_j^+$ exchanges antiparallel spins, but vanishes when applied to a pair of parallel spins. The terms of this type produce off-diagonal matrix elements equal to $\frac{J}{2}$ between basis states (i.e. spin configurations) that are related by a single exchange of nearest neighbor spins. The terms of the form $S_i^z S_j^z$ combine to give a diagonal matrix element for each state equal to $\frac{J}{4}$ times the difference between the number of parallel nearest neighbor spins and the number of anti-parallel nearest neighbor spins in that configuration. Despite the simplicity of the model, no analytic solutions have been found for nontrivial structures except in one dimension [3].

Since the Hamiltonian is invariant under uniform rotations of the spins, one can choose its eigenstates to be simultaneous eigenstates of the operators $\vec{S}_{TOT}^2$ and S_{TOT}^z, where $\vec{S}_{TOT}$ is the total spin. For a system containing an even number of spins n, whatever the ground state value of $\vec{S}_{TOT}^2$, there is always a ground state with $S_{TOT}^z = 0$. Therefore, a ground state can always be found in the subspace spanned by the

$$N_{total} = \frac{n!}{(n/2)!(n/2)!} \tag{2.2}$$

basis states with an equal number of up and down spins. The generalization to an odd number of spins is straightforward. Since the Hamiltonian is real, the ground state eigenvector can be chosen to be real.

In this chapter, we solve this model for a series of structures that embody the basic structural features of the fullerenes, which are spherical shells of threefold coordinated carbon atoms arranged in pentagonal and hexagonal rings. It can be shown that every such structure must have twelve pentagonal faces [6]. The total number of sites can be varied by changing the number of hexagons. The smallest such structure contains no hexagons and has 20 sites. Figure 2.1 shows several fullerene related structures that we discuss in this chapter. We shall refer to the structures in Fig. 2.1 (a)–(e) as F-20, F-24, F-26, F-28, and F-32, respectively. For simplicity, we shall treat all of the bonds in these

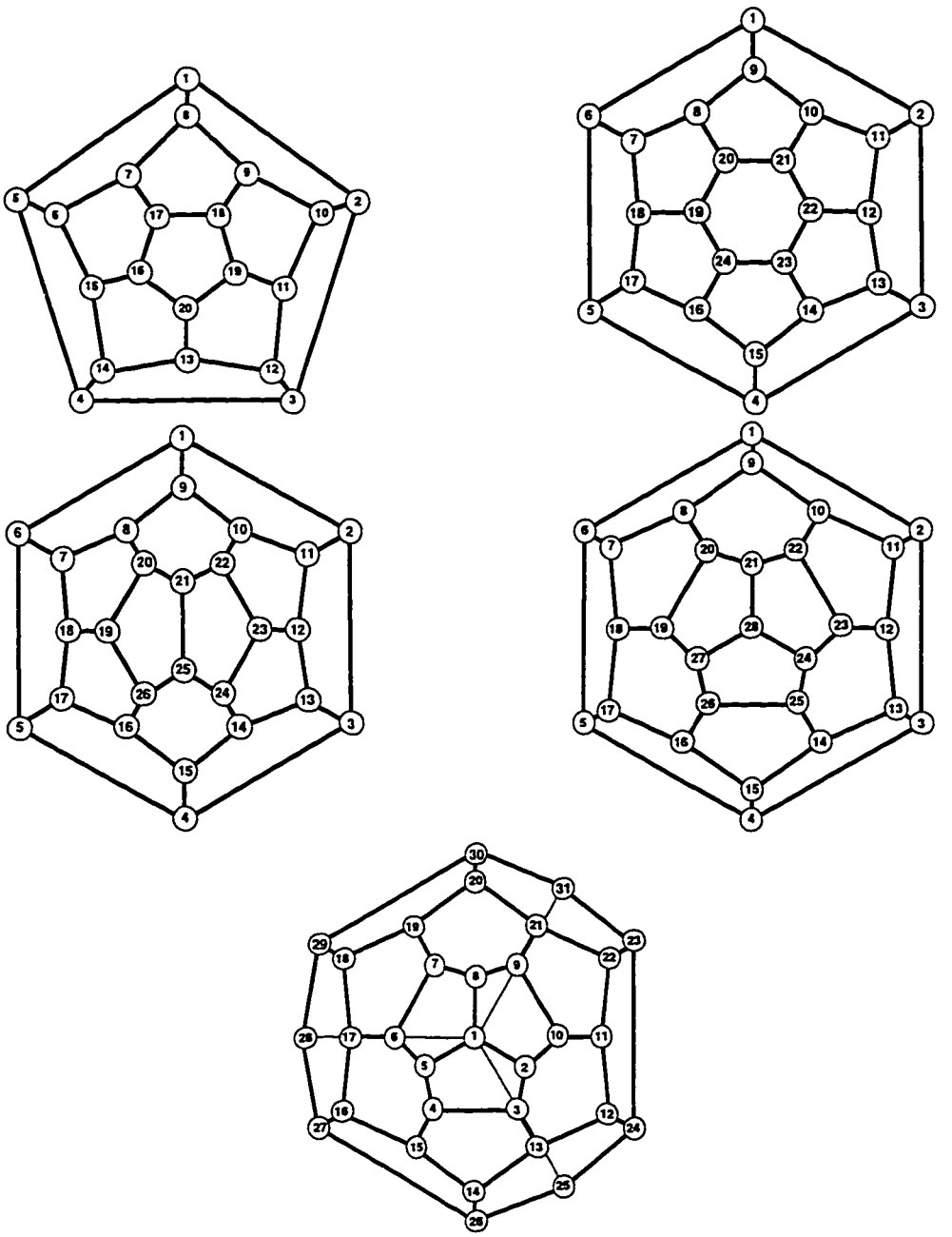

Figure 2.1: (a) 20-site, (b) 24-site, (c) 26-site, (d) 28-site, and (e) 32-site frustrated structures with spherical topology. The 3-dimensional structures are shown projected on a plane, which introduces a distortion of relative distances. Therefore, the figures only indicate the connectivity of the the structures, and the apparent lengths of the bonds are not to be interpreted literally.

Table 2.1: Some properties of the structures considered in this chapter. Symmetry indicates the point group symmetry of the structure.

Structural Properties				
Structure	**Symmetry**	**Pentagons**	**Hexagons**	**Dimension of Space**
F-20	I_h	12	0	184,756
F-24	D_{6d}	12	2	2,704,156
F-26	C_{3v}	12	3	10,400,600
F-28	T_d	12	4	40,116,600
F-30	C_{2v}	12	5	155,117,520
F-32	D_3	12	6	601,080,390
F-60	I_h	12	20	1.2×10^{17}
H-18	C_{3v}	0	9	48,620
H-24	C_{3v}	0	12	2,704,156
H-26	C_{3v}	0	13	10,400,600

structures as equivalent even though in actual carbon clusters they may differ. On a pentagonal ring, it is impossible to arrange all spins in an antiferromagnetic pattern. This introduces frustration in the classical ground state where nearest neighbor spins would prefer to be antiparallel. For comparison, we also study several unfrustrated structures that are derived from the honeycomb lattice by applying periodic boundary conditions. These structures are shown in Fig. 2.2 (a)–(c). We refer to these structures as H-18, H-24, and H-26, respectively. These structures have toroidal topology rather than the spherical topology of the frustrated structures. Table 2.1 summarizes the geometrical features of the structures that we investigate.

A group of powerful techniques used to investigate quantum many-body systems such as the HAFM are based on quantum Monte-Carlo methods. In systems with frustration, these methods either require the summation of a very large number of

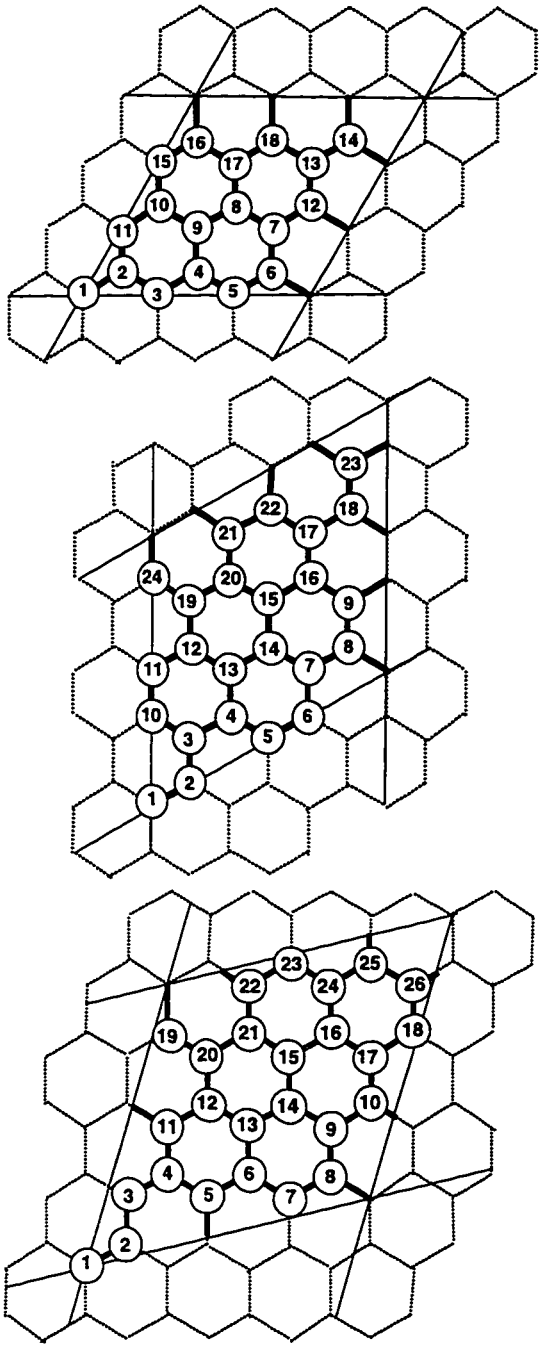

Figure 2.2: (a) 18-site, (b) 24-site, and (c) 26-site unfrustrated structures derived from the honeycomb lattice by applying periodic boundary conditions along the thinner solid lines.

terms with alternating signs (known as the sign problem) or depend on a "guiding" wavefunction which must be properly guessed. Here we use a different approach based on exact diagonalization of the Hamiltonian matrix. This approach has the advantage of not being affected by the sign problem, but is limited to rather small system sizes because the number of states in the Hilbert space grows exponentially with the size of the system. For example, in Table 2.1 we list the number of states in the $S^z_{TOT} = 0$ subspace for each cluster that we investigate. Thus, it takes a major increase in either computer power or efficiency of the algorithm to get a modest increase in the size of system that can be investigated.

Using exact diagonalization techniques, modern computers can handle systems with $\leq$ 36 spins. A 36 spin system has about 9 billion basis states in the subspace with $S^z_{TOT} = 0$. The Hamiltonian matrix is sparse and has only about 300 billion nonzero entries for this size system. Memory constraints make it difficult to store this matrix. The symmetries of the structure must be used to reduce the size of the basis space in order to make calculations tractable. The usefulness of symmetrization depends on how many mutually commuting symmetry operations can be found. Symmetry is most useful for lattices where all translations commute, such as the square lattice. Even noncommuting symmetries could be easily exploited if the ground state was known to transform according to the identity representation of the symmetry group. This can not be assumed to be the case for the frustrated HAFM. To our knowledge, the largest structure that has been solved using exact diagonalization and taking advantage of all of its symmetries is the 36-site square lattice [7]. It would be very difficult to find the ground state of a structure with the same size and a lower number of commuting symmetries without approximation.

One way to manage larger systems is to restrict the wavefunction to the space spanned by a subset of the basis states. In this approach, the problem is transformed into finding a subspace that accurately approximates the full-space result, but that is

small enough to be handled computationally. In this chapter, we variationally optimize the truncation of the Hilbert space, and exactly diagonalize within the truncated space. The rest of the chapter is organized as follows: section 2.2 contains a justification for our choice of optimal wavefunction and truncated space, section 2.4 discusses the ground state properties that we obtain with this approach for a series of frustrated and honeycomb clusters, and section 2.5 summarizes our conclusions.

2.2 Choice of Truncation

Consider a truncation of the space to the basis states $\{|\alpha_1\rangle, |\alpha_2\rangle, \ldots, |\alpha_{N_{trunc}}\rangle\}$ where $N_{trunc} < N_{total}$. Define a truncated Hamiltonian that consists of those elements of the original Hamiltonian that connect states retained in the truncated space. Let $E(\{\alpha_1, \alpha_2, \ldots, \alpha_{N_{trunc}}\})$ denote the smallest eigenvalue of the truncated Hamiltonian. We define the optimal truncation as the one that minimizes E with respect to all sets with N_{trunc} basis states. By the variational principle, the ground state wavefunction of the corresponding truncated Hamiltonian is the wavefunction that, subject to the constraint of vanishing for all but N_{trunc} states, minimizes the expectation of the full Hamiltonian. Therefore, E for the optimal truncation is the smallest possible variational upper bound on the true ground state energy that can be obtained using trial wavefunctions that have no more than N_{trunc} nonzero components.

The minimization over sets of basis states is accomplished using a stochastic search: An initial truncation is chosen and the ground state energy of the corresponding truncated Hamiltonian is found using the Lanczos method [8]. Moves in the stochastic search consist of adding states to the space and eliminating others while keeping the overall number of states fixed. The Lanczos method is used at each step to find the ground state energy for the new truncation, and the move is accepted or rejected according to a Metropolis algorithm [9, 10]. This procedure is repeated until all new moves are rejected, in which case a minimum of $E(\{\alpha_1, \alpha_2, \ldots, \alpha_{N_{trunc}}\})$ has been found. If this

is the global minimum, the resulting truncation is the ideal truncation. We have found no evidence that the procedure gets trapped in local minima.

For systems that are small enough that the full problem can be solved, we have also applied an alternative truncation procedure for purposes of comparison with our variational scheme. This consists of keeping only the basis states that have the largest weights in the full-space ground state solution and varying the cutoff weight below which states are excluded from the basis. The energy obtained from this alternative procedure must be greater than or equal to the variational result, but the wavefunction from this alternative procedure is expected to be closer to the true ground state. Therefore, this alternative procedure might be expected to yield better results for correlation functions. A comparison of results obtained using these two independent methods helps to assure that the variational procedure is converging properly and shows that the procedure produces reasonable correlation functions.

In order to optimize the variational search, it is necessary to bias the selection of the states to be added to, or eliminated from, the truncated basis during each step. The procedure proposed here is analogous to force-bias Monte Carlo. In our case, the equivalent of the force in a particular direction is the difference between the energy when a particular state $|\beta\rangle$ is included in a truncation and the energy when the state is not included in the truncation:

$$\nabla_\beta E(\{\alpha_1, \alpha_2, \ldots, \alpha_{N_{trunc}}\}) \equiv E(\{\alpha_1, \alpha_2, \ldots, \alpha_{N_{trunc}}, \beta\}) - E(\{\alpha_1, \alpha_2, \ldots, \alpha_{N_{trunc}}\}). \tag{2.3}$$

In force-bias Monte Carlo, the force is a function of the configuration of the system, and correspondingly $\nabla_\beta E$ is a function of the set of states included in the truncation. In our case, since each state is either included or not included, we must take

$$\nabla_\beta E(\{\alpha_1, \alpha_2, \ldots, \alpha_{N_{trunc}}, \beta\}) \equiv \nabla_\beta E(\{\alpha_1, \alpha_2, \ldots, \alpha_{N_{trunc}}\}). \tag{2.4}$$

$\nabla_\beta E$ can be estimated easily for each β using the solution from the previous truncation: We denote the states that are included in the previous truncation as internal states, and

the remaining states of the full Hilbert space as external states. The internal states are the states that could be eliminated from the previous truncation in the process of forming the new truncation, while the external states are the states that could be added. For each internal state, we wish to calculate the change in the variational energy caused by eliminating it from the previous truncation. Let the ground state wavefunction for the previous truncation be $|\Psi_0\rangle$ and let $\psi_\beta = \langle\beta|\Psi_0\rangle$. We approximate the ground state of the truncation with the state $|\beta\rangle$ eliminated by assuming that the rest of the wavefunction remains unchanged except for an overall normalization factor,

$$|\Psi_{0-\beta}\rangle = \frac{|\Psi_0\rangle - \psi_\beta|\beta\rangle}{\sqrt{1-|\psi_\beta|^2}}. \tag{2.5}$$

To first order in $|\psi_\beta|^2$ this approximation gives,

$$\nabla_\beta E = E_0 - \langle\Psi_{0-\beta}|H|\Psi_{0-\beta}\rangle = |\psi_\beta|^2 (E_0 - H_{\beta\beta}) \tag{2.6}$$

where $H|\Psi_0\rangle = E_0|\Psi_0\rangle$ and $H_{\beta\beta} = \langle\beta|H|\beta\rangle$. Similarly, the effect of adding an external state is approximated using second order perturbation theory as:

$$\nabla_\beta E = \frac{|\langle\beta|H|\Psi_0\rangle|^2}{E_0 - H_{\beta\beta}}. \tag{2.7}$$

Note that $\nabla_\beta E$ will be zero if $|\beta\rangle$ is neither an internal state nor an external state that is connected by the Hamiltonian to an internal state. Depending on the stage of the variational procedure, a set of trial states is chosen which either contains all of the states for which $\nabla_\beta E$ is nonzero, or a randomly chosen subset of such states. $\nabla_\beta E$ is calculated for this set, and the new truncation is formed by taking the states with the largest values. Choosing a random subset of trial states introduces a stochastic element into the computation and effectively reduces the variational step size. During a minimization procedure where the full set of trial states is used at every step, a move will eventually be rejected in the Monte-Carlo evaluation. Further iterations beyond this point will simply generate the same move. This is similar to a gradient minimization with a fixed step size where the step overshoots the minimum. Here, since each state is

either included or not included, it is impossible to reduce the step size in the usual sense. Instead, the step size can be effectively reduced by using a randomly chosen subset of the components of the gradient. The fastest minimization is achieved by using all of the trial states until the first move is rejected, and then considering a random subset which is gradually reduced in size. For the HAFM model considered in this chapter, we found that our move selection algorithm was so effective that additional moves after the first rejected move produced minimal improvements in the energy. Accordingly, we stop the variational procedure when the first move is rejected.

The idea of iterative improvement of a Hilbert space truncation using perturbative estimates of the importance of new states has a long history in the quantum chemistry literature [11, 12, 13, 14, 15]. In addition, for this class of problems, the final truncated results are typically corrected with a perturbative treatment of the remaining states [16, 17, 18, 19]. Extrapolation methods are also frequently used [20]. Such methods would likely be a useful addition to our method, but since the emphasis of this chapter is on a variational approach, we have avoided such corrections. Iterative improvement of a Hilbert space truncation has also been studied in the context of quantum lattice models. De Raedt and von der Linden estimated the importance of a new basis state by means of the energy lowering obtained from a Jacobi rotation involving the state [21]. Riera and Dagotto added basis states that are connected by the Hamiltonian to states with a large weight in the current truncated solution [22]. In this previous work, the basis is expanded by adding selected new states until either the desired quantities converge or computational limits are reached. In contrast, our emphasis is on finding the optimal basis of a given size. Working with a constant size basis has two advantages:

(1) It allows us to define the optimal basis in an unambiguous manner and to express the problem of finding this optimal basis as a minimization problem. This makes it possible to harness the full power of the Metropolis algorithm and the simulated annealing approach.

(2) It allows us to tackle problems with no clear hierarchy of importance among the basis states. In quantum chemistry, there is a hierarchy of states in which higher excitations are progressively less important. In contrast, the frustrated HAFM lacks any clear *a priori* hierarchy among the basis states. As a result, truncation can induce level crossings and change the character (e.g. the symmetry) of the ground state. If a basis selection process were to start with an incorrect ground state, augmentation of the truncation runs the risk of not selecting the basis states that are important for the true ground state. This makes it likely that the true ground state would never be found. By working with a basis of a constant size, which is variationally optimized, we avoid this problem.

The effectiveness of the variational Hilbert space truncation procedure can be demonstrated by comparing its results to those obtained from the full-space solution. Define the fractional error in the energy for a given truncation by

$$\delta\epsilon(\{\alpha_1, \alpha_2, \ldots, \alpha_{N_{trunc}}\}) = \frac{E(\{\alpha_1, \alpha_2, \ldots, \alpha_{N_{trunc}}\}) - E_{N_{total}}}{E_{N_{total}}}, \tag{2.8}$$

where $E_{N_{total}}$ is the full-space ground state energy. Figure 2.3 shows $\delta\epsilon$ for the truncation resulting from the variational truncation procedure and the truncation resulting from keeping the states with the largest weights in the full-space solution. The energies found using the variational procedure are just slightly below those found by truncating based on the full-space solution. The fact that the variational energies are the lowest energies indicates that the variational minimization is converging properly. The closeness of the two results indicates that our definition of a best truncation is successful in capturing the most important parts of the full-space wavefunction. The difference between the two results grows as the retained fraction of the space diminishes and as the physical system gets smaller, but it stays relatively insignificant except for the smallest truncation size of the smallest structure. For example, retaining only $\frac{1}{6}$ of the basis states of the F-20 structure results in only about 1 percent error in the energy. Note that in order to get the same fractional error, a smaller fraction of the basis vectors is required for the larger

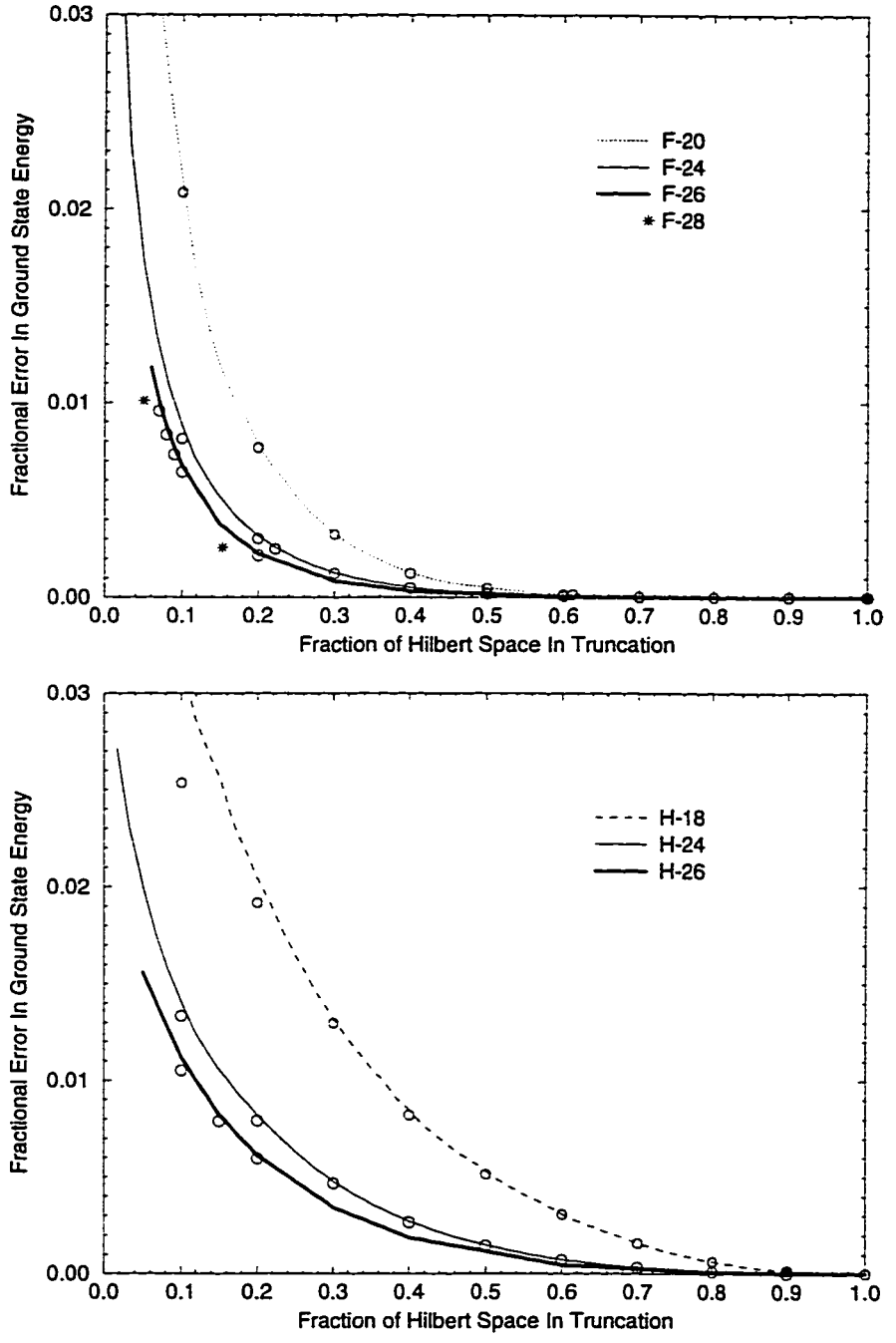

Figure 2.3: Error in the ground state energy resulting from truncation of the Hilbert space for (a) some frustrated structures and (b) some unfrustrated structures. Lines indicate the results of truncating the Hilbert space based on the weights of states in the full space solution. Individual points indicate results from the variational method and correspond to the same structure as the line immediately above them. The stars indicate results for the 28 site system, where results from truncating based on the full-space solution are unavailable.

systems. As a result, the number of states that must be retained in the truncated space grows more slowly than the number of states in the full-space. Therefore, larger systems make truncation increasingly useful. The curves resulting from the frustrated structures have a different shape than the curves resulting from the unfrustrated structures. The error falls more slowly for the unfrustrated structures than for the frustrated structures as the retained fraction of space increases. This suggests that the method is more useful for frustrated structures.

Figure 2.4 shows the correlations for the honeycomb lattice structures as a function of the fraction of space retained in the truncation. Since for these structures the nearest neighbor correlation function is proportional to the energy, it is not included. The multiple lines are due to the fact that the 24 and 26 site structures each have two inequivalent 3rd neighbor correlations, and the 24 site structure has two inequivalent 4th neighbor correlations. Again, both the results of the variational truncation method and the results of truncating the Hilbert space based on the weights of states in the full-space solution are shown. The truncation based on the full-space solution is expected to give a better approximation to correlation functions than the variational method, but for the correlations considered here, the results of two methods are almost indistinguishable. Furthermore, truncation down to a few percent of the space by either of these methods introduces only a few percent error in the correlations. Since the HAFM on the honeycomb lattice has long range order, all of the correlations are fairly large in magnitude. This causes our truncation methods to give particularly good results for these correlations.

In contrast, correlations between sites that are far apart on the frustrated structures are a worst case situation. Correlations on the frustrated structures usually become very small at long distances. As a result, the fractional error in these correlations is quite large. Figure 2.5 shows the fractional error in the correlation that is smallest in magnitude for the 20, 24, and 26 site frustrated structures. The full-space values of these

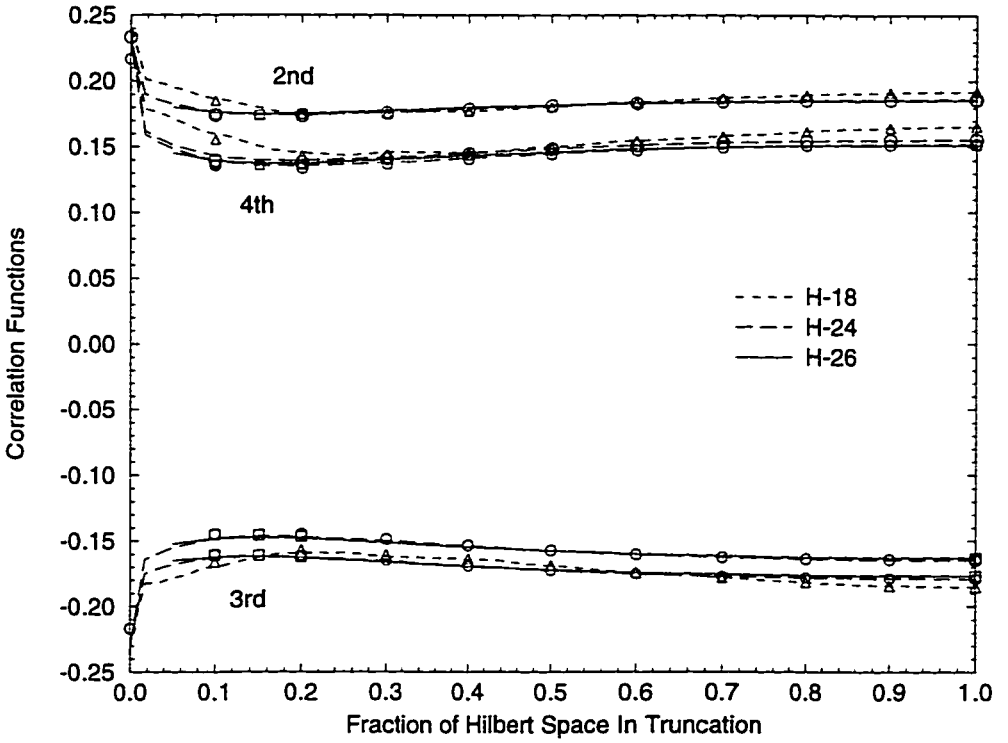

Figure 2.4: Spin-spin correlations for the unfrustrated structures based on the honeycomb lattice as a function of the fraction of states retained in the truncated Hilbert space. Lines indicate results from truncating the Hilbert space based on the weights of states in the full-space solution. Individual points indicate results from the variational method. The groups of lines are labeled by nearest neighbor distances. The points near zero abscissa correspond to small, but finite, truncations.

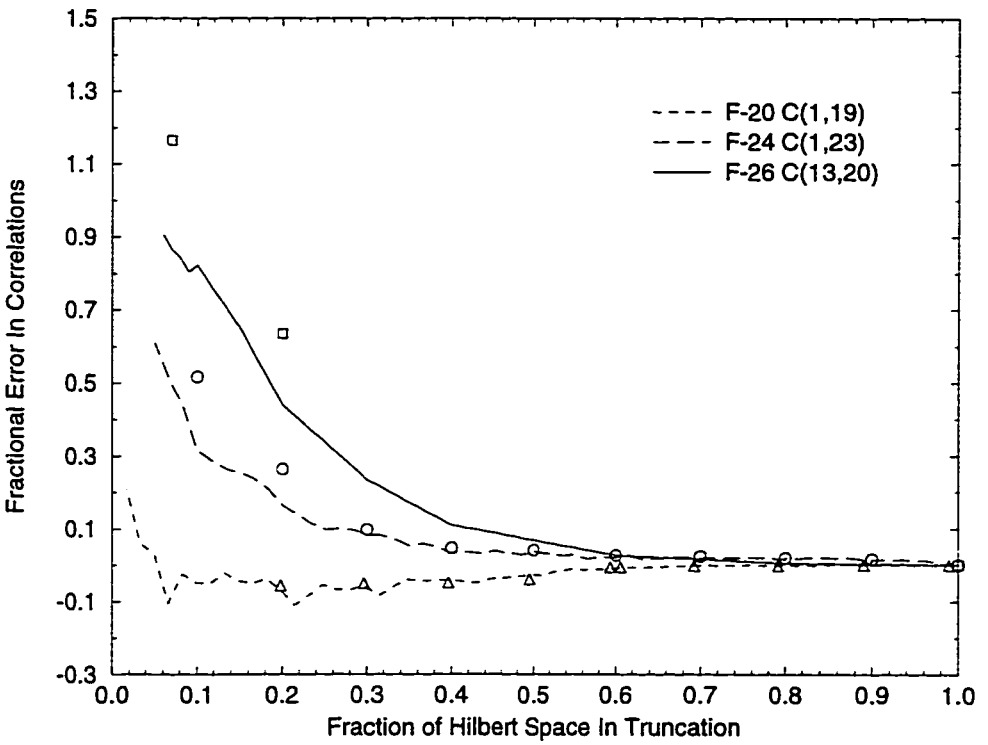

Figure 2.5: Fractional error in the smallest spin-spin correlations on the frustrated structures as a function of the fraction of states retained in truncated Hilbert space. Lines indicate results from truncating the Hilbert space based on the weights of states in the full-space solution. Individual points indicate results from the variational method.

correlations are 3.31×10^{-2}, -3.43×10^{-3}, and 2.02×10^{-3}, respectively. With less than half of the space retained, the fractional error introduced in these correlations becomes substantial. The error resulting from the variational truncation method is rather similar to the error introduced by truncating the Hilbert space based on the weights of states in the full-space solution. The fractional error in a correlation seems to grow with the inverse of the magnitude of the correlation.

2.3 Implementation of the Method in a Massively Parallel Architecture

Since we are interested in the most accurate approximation to the full-space properties of the system, it is desirable to make the size of the truncation as large as possible. As mentioned above, memory is the primary constraint on the size of the system that can be handled using exact diagonalization techniques. Thus, effective implementation of this algorithm requires careful treatment of memory usage. The requirement of maximizing speed while minimizing memory usage provides a particular programming challenge to implementing the variational Hilbert space truncation method. We have implemented the method on the Naval Research Laboratory's 256 node Thinking Machines Corporation CM-5E supercomputer.

The largest size truncation that we solved consisted of 20 million states, which is 3.33% of the full-space of the F-32 structure. The diagonalization of such matrices is a time consuming process. Our implementation on the CM5 massively parallel architecture provided a vivid demonstration of the conflict between efficient use of memory and efficient use of CPU time. The Hamiltonian matrix can either be stored in core memory or generated during each matrix-vector multiply required by the Lanczos method. Storing the Hamiltonian reduces the time by about a factor of ten at the expense of a factor of four increase in the memory. A third possibility would be to store the Hamiltonian

on an external device with fast access, such as the Scalable Disk Array (SDA). Because the SDA's total capacity is only about three times that of the core memory, we have not implemented this option.

Multiplication of the wavefunction by the unstructured, sparse Hamiltonian matrix requires general communication between sections of memory distributed to different processors, and therefore it is not expected to parallelize efficiently. Such multiplications form the core of the Lanczos algorithm. Careful implementation of these multiplications as well as the generation of the new truncations and Hamiltonians is essential to good parallel performance. We separate the techniques used to obtain reasonable efficiency, while avoiding excessive memory use, into three categories: the use of previous results during the generation of new results, the balanced division of work over both processors and time (load balancing), and the usage of sorting instead of searching. These are discussed in order:

2.3.1 Use of previous results

There are three tasks that must be accomplished during each iteration of the variational Hilbert space truncation method: generation of the truncated space that will be investigated during the iteration, generation of the corresponding Hamiltonian, and diagonalization of the Hamiltonian. Since each truncation is a variation of the previous truncation, it is possible to use results from the previous iteration to speed up the calculation considerably. The most important gain in efficiency is obtained by initializing the Lanczos routine with a guess wavefunction derived from the results for the previous iteration by using first order perturbation theory. This requires very little extra work since all of the expensive steps of the perturbation theory are already carried out as part of the generation of each new truncation. This procedure can reduce the time required to find the ground state by a factor of 100.

2.3.2 Load balancing

In order to get a reasonable rate of performance out of a parallel computer, it is necessary to group sets of similar operations together. On the other hand, avoiding the use of large amounts of memory requires divying up similar operations over time so that the memory needed to perform each group of operations can be reused. Therefore, getting good utilization of both processors and memory requires groups of operations that are neither too big, nor too small. In general, the best performance is achieved by identifying the largest unavoidable use of memory, and then using groups of operations that are somewhat smaller. One example is the generation of the Hamiltonian, where the best compromise is to consider all of the elements resulting from exchanging one pair of nearest neighbor spins at the same time. Another example is provided by the selection of each new truncation. It is necessary to compute $\nabla_\beta E$ for each trial state $|\beta\rangle$ [see Eqs. (2.6) and (2.7)]. For most cases of interest, there are many more trial states than states in the truncation. Thus, if the Hamiltonian is not stored, it is efficient in terms of memory usage to divide the trial states into smaller sets and compute for one set at a time. This is possible since only the N_{trunc} states with the largest values of $\nabla_\beta E$ need to be retained at each step. This technique allows sets of trial states of arbitrary size to be considered when generating each variational step.

2.3.3 Sorts instead of searches

A standard problem encountered during numerical calculations involving spin models is that information (in this case, the components of the wavefunction) about each of the spin configurations must be packed into memory in some manner that allows its quick retrieval. It is trivial to associate each spin configuration with a unique number, but the resulting set of numbers is not usually dense. Considerable research effort has been expended in developing efficient hashing routines for locating the memory addresses associated with a given spin configuration [24, 25]. Since the set of basis states changes

stochastically during each iteration of our variational truncation process, the algorithm requires a flexible hashing procedure without substantial overhead for setup. For our implementation, we also needed a procedure that parallelizes efficiently. The radix sort algorithm, which consists of hashing on successive blocks of bits, sorts a list of N_{keys} in a time proportional to N_{keys}. This algorithm parallelizes ideally (it uses a time proportional to N_{keys}/N_{proc} on a machine with N_{proc} processors) and is stable (if two entries are equal, the entry with the smaller initial subscript will be sorted to the location with the smaller final subscript). This points to combining many searching operations together and using sorts to do searching efficiently on a parallel machine. We implemented a procedure based on inter-sorting a list of states with unknown memory addresses with a list of all the states in the truncation. This procedure worked so well that we were able to generate the Hamiltonian during each matrix-vector multiply rather than storing it, thereby saving on memory usage and extending the size of the system that could be handled.

2.4 Results

Table 2.2 summarizes some of the ground state properties of the HAFM on the structures we considered. The expectation of $\vec{S}^2_{TOT}$ can be calculated by summing the correlation functions between all pairs of sites. Since each structure considered has an even number of spins, the possible exact eigenvalues are $s(s+1)$ where s is an integer. Deviation from these values can be expected for truncated solutions because the truncation procedure breaks the invariance of the model under global spin rotation. For each of our full-space solutions (which includes all structures studied except F-32), the expectation of $\vec{S}^2_{TOT}$ is 0 to the accuracy of the solution. Thus, for every system except F-32, the calculated ground state is a spin singlet. For the truncated solution of the F-32 system, this expectation is ≈ 0.5. This value is between the values expected for a spin singlet ($s(s+1) = 0$) and a spin triplet ($s(s+1) = 2$). It is much closer to the value of the

Table 2.2: Ground state energy and nearest neighbor correlation functions of each structure. Results for the F-32 structure are from a truncation retaining 20 million of the 601 million states. All other results are from full-space solutions.

Ground State Properties					
Structure	**E_0/Site**	$H-H$	$H-H'$	$H-P$	$P-P$
F-20	−1.722219				$C_{1,2} = -0.324$
F-24	−1.726614	$C_{1,2} = -0.409$		$C_{1,9} = -0.203$	$C_{8,9} = -0.371$
F-26	−1.719921	$C_{1,2} = -0.424$ $C_{2,3} = -0.339$	$C_{1,9} = -0.103$	$C_{2,11} = -0.265$	$C_{11,12} = -0.332$
F-28A	−1.719633	$C_{1,2} = -0.275$ $C_{1,6} = -0.362$ $C_{2,3} = -0.425$	$C_{1,9} = -0.327$ $C_{3,13} = -0.063$	$C_{2,11} = -0.321$ $C_{6,7} = -0.269$	
F-28B	−1.719633	$C_{1,2} = -0.433$ $C_{1,6} = -0.346$ $C_{2,3} = -0.283$	$C_{1,9} = -0.151$ $C_{3,13} = -0.415$	$C_{2,11} = -0.286$ $C_{6,7} = -0.338$	
F-32	−1.736	$C_{2,3} = -0.420$ $C_{2,10} = -0.352$ $C_{3,13} = -0.407$ $C_{4,15} = -0.509$ $C_{11,12} = -0.335$	$C_{3,4} = -0.101$ $C_{11,22} = -0.123$	$C_{1,2} = -0.279$	$C_{12,13} = -0.333$
H-18	−1.871907	$C_{1,2} = -0.374$			
H-24	−1.860839	$C_{1,2} = -0.370$			
H-26	−1.858385	$C_{1,2} = -0.369$			

spin singlet than to the triplet. Moreover, we have found that the variational procedure tends to decrease this value, indicating that the ground state of F-32 is also a spin singlet. Table 2.2 contains two entries for the F-28 structure because its ground state is a rotational doublet. The rest of the states are rotational singlets. The two F-28 states are distinguished by considering their transformation properties under improper rotation about the symmetry axis through the center of the bond between site 19 and site 20 (see Fig. 2.1 (d)). Under this transformation, the F-28A state has eigenvalue 1, while the F-28B state has eigenvalue -1.

The first column of Table 2.2 contains the ground state energy per site. As expected, frustration raises the ground state energy. The energies per site of the structures based on the honeycomb lattice reveal the expected finite size effects for the HAFM on a lattice: the energy per site increases as the size of the system increases. Finite size effects are not as clearly evident in the frustrated structures, but the trend from F-24 to F-26 to F-28 is rather similar to what could be expected from finite size effects. The trend is reversed in F-32. These clusters are not especially similar to each other except for overall topology, so it is reasonable that finite size effects are obscured by effects due to details of the structure. Furthermore, as the size of the frustrated structures increases, the hexagonal rings become more plentiful and closer together. Thus, these systems should behave more like the unfrustrated structures at larger sizes. Eventually, the energy must decrease toward the unfrustrated value. It is likely that the drop in energy between F-28 and F-32 indicates the beginning of this trend. Note that this drop in energy can not be a result of using a truncated solution for the F-32 system since the energy resulting from the truncation must be greater than the full-space energy.

The rest of the columns in Table 2.2 show the nearest neighbor spin-spin correlations. The correlation between site i and site j is defined by

$$C_{i,j} = \langle \Psi_0 | \vec{S}_i \cdot \vec{S}_j | \Psi_0 \rangle \tag{2.9}$$

where Ψ_0 is the ground state wavefunction. The sum of all of the nearest neighbor

correlations for a particular structure gives the ground state energy. Even though the ground state energies vary relatively little, the nearest neighbor correlation functions vary dramatically (see Table 2.2). The nearest neighbor correlations are divided into four columns. The column labeled $H - H$ contains correlations between sites that are both located on the same hexagonal ring. The column labeled $H - H'$ contains correlations between sites that are located on two different hexagonal rings. The column labeled $H - P$ contains correlations between a site located on a hexagonal ring and a site that is not located on any hexagonal ring. The column labeled $P - P$ contains correlations between two sites neither of which is on a hexagonal ring. Fig. 2.1 and Fig. 2.2 serve as keys to the labeling of the sites.

All of the nearest neighbor correlation functions are negative, which is not surprising since the ground state wavefunction is chosen to minimize the sum over these correlations. In order to provide physical insight into the results, we consider the following argument: it is possible to solve the HAFM analytically on a structure consisting of a central site and its three neighbors. The sum of the three correlations for this system is $-5/4$. The variational principle can then be used to show that for a general structure, the sum of the three correlations between a given site and its neighbors can not be less than $-5/4$. This sum is reduced in magnitude by frustration and by quantum fluctuations when additional sites are included in the structure. However, the existence of the strict bound discussed above suggests that a strong correlation between a site and one of its neighbors will reduce the correlations to the rest of its neighbors. This behavior is exemplified by the correlations in Table 2.2. The strongest correlations, those in the $H - H$ column, are for the bonds between two sites that are on the same hexagonal ring. Furthermore, the strongest of these correlations are found on the frustrated structures where the bonds that form the hexagonal ring do not have to compete with two other identical bonds. The drop in energy between F-28 and F-32 can be attributed to an increase in the number of bonds of this type. The weakest nearest neighbor cor-

relations are found between sites that are located on different hexagonal rings. These bonds are frustrated and also suffer from strong competition from the bonds on each of the hexagonal rings. To illustrate these arguments in a specific example, consider the F-26 structure. The $C_{1,9}$ and $C_{11,12}$ correlations are both frustrated since each of these bonds is included in two pentagonal rings. The $C_{1,9}$ correlation is much weaker (-0.103) than the $C_{11,12}$ correlation (-0.332) because the $C_{1,9}$ correlation has competition from four strong (-0.424) correlations of the $C_{1,2}$ type (correlations between sites that are on the same hexagon but not on any other hexagons). For similar reasons, the $H-P$ correlations are weaker than the $P-P$ correlations.

The correlation functions for the 28 site frustrated structure are constrained by the symmetries of the wavefunction, and this results in several anomalously small correlations, especially $C_{3,13}$ for the A wavefunction. Although the original structure is tetrahedral, the process of resolving the two degenerate states breaks this symmetry by singling out the symmetry axis through the bond between sites 19 and 20. There is an approximate equivalence of correlations between the results for the two wavefunctions. The role of $C_{1,2}$ is switched with $C_{2,3}$, the role of $C_{1,9}$ is switched with $C_{3,13}$, and the role of $C_{2,11}$ is switched with $C_{6,7}$. Roughly speaking, the correlations that are closest to the axis through the bond between sites 19 and 20 switch places with the correlations that are furthest away form this axis. The F-28 structure has unusually strong long range correlations between the sites labeled as 7, 11, 15, and 28 in Fig. 2.1 (d). These sites form the corners of a tetrahedron. For the F-28A state, the correlations of this type perpendicular to the symmetry breaking axis ($C_{7,28}$ and $C_{11,15}$) are 0.141 and the other correlations of this type ($C_{7,11}$, $C_{7,15}$, $C_{11,28}$, and $C_{15,28}$) are 0.136. For the F-28B state, these correlations are 0.134 and 0.139 respectively. This result is interesting because it suggests strong ferromagnetic correlations between the spins on the four apex sites that form the corners of a tetrahedron in F-28. This is consistent with quantum mechanical calculations of the electronic structure of the C_{28} molecule, which is believed to have

the same structure as the F-28 cluster: in those calculations, the molecule is found to have an $s = 2$ ground state, with the spins in the four apex sites aligned [23].

2.5 Conclusion

The variational Hilbert space truncation approach provides an effective way to extend the range of structures for which exact diagonalization of the HAFM is feasible. Substantial reductions in memory can be obtained with less than a 1% error in the ground state energy. A few percent error is introduced in most correlations. The exception is very weak correlations for which the method will give a rough idea at best. For system sizes that are at the current leading edge of computational capabilities, a reduction of the Hilbert space by a factor of thirty can be achieved. For the HAFM, a factor of thirty reduction in memory use allows structures with about 5 additional sites to be handled. Our method is compatible with symmetrization techniques, which, depending on the structure under consideration, can achieve a similar reduction in memory requirements. Finally, our method should be useful for models other than the HAFM. In fact, much larger reductions in the size of the Hilbert space can be expected for systems where the ground state is dominated by a few of the basis states used in the expansion of the wavefunction. For such systems, the method should be capable of identifying the important basis states, and thus the important physics of the ground state.

Using this variational approach, we have successfully determined the ground state properties of the HAFM on a series of frustrated and unfrustrated structures. An interesting and unexpected result is the doublet nature of the ground state of the 28 site frustrated structure. The 32 site frustrated structure seems to be a rotational singlet, but it would be interesting to know whether other larger structures of this type also break structural symmetries.

Acknowledgement

This work was supported by ONR Contract #N00014-93-1-0190. The computations were performed on the NRL 256-node CM-5 supercomputer. We acknowledge helpful input during the initial stages of this project from Prof. L. Johnsson.

Chapter 3

ACRES: Adaptive Coordinate, Real-Space Electronic Structure Calculations for Atoms, Molecules, and Solids

3.1 Introduction

Ab initio electronic structure calculations based on the Hohenberg-Kohn-Sham density functional theory [1] have the remarkable capability of accurately predicting physical properties of real systems. However, the size of system that can be treated with these methods is limited because solving the Kohn-Sham equations [2] is computationally demanding. Therefore, the development of new methods that improve the efficiency of such calculations is an important priority.

3.1.1 Desirable properties

An efficient electronic structure method should satisfy certain requirements. Three desirable properties for any electronic structure algorithm are sparsity, parallelizability, and adaptability.

Sparsity

Let N_b be the size of the basis used to represent the wavefunctions in an electronic structure calculation. If the Hamiltonian is a dense matrix, storing the Hamiltonian requires N_b^2 memory locations. As N_b increases, the memory required to store a dense Hamiltonian will eventually become prohibitive. Furthermore, traditional matrix diagonalization techniques designed to find all eigenvalues and eigenvectors of dense matrices require $O(N_b^3)$ operations. The prefactor in these methods is small, but as the system size increases they eventually lose to methods with a more efficient scaling. In contrast, storage of a sparse matrix with a constant sparsity requires only $O(N_b)$ memory locations and multiplying a vector by such a matrix requires only $O(N_b)$ operations. A variety of iterative eigensolver algorithms that require only an implicit representation of the matrix through its action on an arbitrary vector have been developed. As a result, a sparse representation of the Hamiltonian offers the potential of solving for each occupied wavefunction in only $O(N_b)$ operations. This would give an overall $O(n_e N_b)$ scaling of an electronic structure computation with n_e occupied states. Therefore, a sparse Hamiltonian coupled with iterative algorithms could vastly reduce both memory and time requirements. In addition, since $O(N)$ methods require either the wavefunctions or the density matrix to be localized to some subset of the basis functions, a sparse Hamiltonian is also a prerequisite for any $O(N)$ treatment of electronic structure. Due to these considerations, attempts to develop more efficient electronic structure methods usually start with a sparse Hamiltonian.

Parallelizability

The ever growing gap between the power of parallel computers and the best performance that can be achieved from a single processor demands that a state-of-the-art electronic structure method must be able to harness efficiently the computational power of massively parallel architectures. This need for good parallelizability imposes additional constraints on the method. The main criteria for the efficient use of a parallel machine are good load balance and efficient communications.

A mapping onto a parallel machine that assigns identical tasks to each processor ensures good load balance. In order to avoid some processors being idle while waiting for others to finish their work, computational complexity must be evenly divided among processors. Likewise, if memory requirements are not evenly divided, some processors will run out of memory while others still have space available. If the processors all execute the same set of operations, but on different sets of data, both computational effort and memory requirements are automatically ensured to be balanced.

A mapping onto a parallel machine that produces a local, structured interprocessor communication pattern ensures efficient communications. Efficient communications is critical for parallel performance because communications operations typically take tens or hundreds of times longer than operations that involve only one processor. As much as possible, both the information that is needed for a given computation and the results of the computation should be located in the processor in which the calculation will occur. When this is not possible, it is advantageous to assign such data to processors that have physical connections to the processor that will actually do the computation. This avoids wasteful forwarding of messages between processors. A communications pattern in which data is only sent between nearby processors is called 'local'. Since efficient ways to simulate common patterns such as 2 or 3 dimensional rectangular grids or binary trees have been worked out for the physical arrangements of processors used in modern parallel computers, it is not actually necessary to arrange the data explicitly so that it

is local with respect to processors on a given parallel machine. Instead, it is sufficient to find an arrangement of the data that is local with respect to some standard pattern. In addition to locality, structure is required in order to ensure efficient communications. In a typical structured communications pattern, all processors send data in the same direction at the same time. This maintains load balance during the communications and prevents conflicts in which one processor is forced to handle more than one message at the same time.

When a computation lacks any *a priori* structure that can be exploited to find an efficient parallel decomposition, the dual requirements of load balance and efficient communications lead to a computational problem known as 'partitioning an unstructured mesh'. A large amount of work on this problem has resulted in considerable progress (see, for example [26, 27, 28]). However, even the best partitioning methods typically recover only a fraction of the performance achievable with a highly structured communications pattern. Furthermore, the partitioning often involves considerable overhead. Therefore, a natural mapping onto a parallel machine that assigns identical tasks to each processor while producing a local, structured interprocessor communication pattern is a highly desirable attribute of an electronic structure method.

Adaptability

Adapting the resolution of the calculation in different regions of space to the demands of the physical system is required for efficient treatment of inhomogeneous systems. Without adaptation, the most demanding part of the problem determines the required precision, and the convergence of physical quantities with increasing basis size is very slow for highly inhomogeneous systems. Examples of inhomogeneous systems include all-electron calculations, pseudopotential calculations with $1s$, $2p$, $3d$, or $4f$ valence electrons, and systems with regions of vacuum.

In all-electron computations, it is necessary to represent accurately the singularities

in the Coulomb potential at the nuclei. Furthermore, the core electron wavefunctions are highly localized and have cusps at the nuclear positions, while the valence electron wavefunctions oscillate rapidly in the core due to orthogonality requirements and the large amount of kinetic energy gained by an electron when it comes close to a nucleus. Accurately representing these rapidly changing and singular features requires an extremely high resolution in the core regions.

Even when pseudopotentials are used to eliminate the Coulomb singularities, the core electrons, and the oscillations in the valence wavefunctions, atoms with $1s$, $2p$, $3d$, or $4f$ valence electrons result in an inhomogeneous problem. These valence electrons are not required to be orthogonal to core electrons with the same symmetry, and therefore they can approach close enough to the nuclei to gain a large amount of kinetic energy. Accurately representing the resulting rapid changes in the wavefunctions requires a much higher resolution than is needed between atoms. Various modified pseudopotentials [29, 30, 31] have been proposed with the goal of reducing this inhomogeneity, but in general, the transferability of these pseudopotentials (i.e. their ability to produce accurate results in a variety of environments) has been poorly tested in comparison to standard Bachelet-Hamann-Schlüter pseudopotentials.

In contrast to the above examples, which involve increased precision near the cores of atoms, a reduction of precision relative to the bonding regions of a system is allowed by large regions of vacuum. Such large regions of vacuum typically occur for systems containing isolated atoms, molecules, clusters, or solid surfaces. In fact, many systems both require an increased resolution in the atomic cores and allow a reduction in resolution in regions of vacuum. Since the size of the atomic cores is much different from the size of the vacuum regions, an efficient calculation for these systems requires adaptation on two different length scales.

3.1.2 Real space methods

The most popular basis for density functional computations has been plane waves. Plane waves do not have any of the desirable properties discussed above: The potentials used in density functional calculations are local in real space, and therefore they produce a dense Hamiltonian in reciprocal space. Since Fourier transforms (the underlying operations in a plane wave basis) require communication from every processor to every other processor, they do not parallelize very well. And a plane wave basis has the same resolution everywhere in space. One might wonder why plane waves have been so widely used. The answer is that they do fairly well in a limited set of circumstances: For uniform materials with very soft pseudopotentials, the plane wave basis represents the wavefunctions quite efficiently. As a result, the number of occupied eigenvectors n_e is a significant fraction of the total number of basis states N_b, which is exactly the situation where $O(N^3)$ diagonalization becomes efficient. Finally, parallel computing did not become a major issue in electronic structure computations until recently. Not surprisingly, the majority of large scale density functional computations have been for materials such as bulk silicon where the above conditions hold. However, it is clear that if harder problems are going to be routinely treated, it is necessary to move beyond plane waves. Many of the most promising methods share a real space approach. In the rest of this section, we will discuss various real space approaches and see how well they achieve the properties of sparsity, parallelizability, and adaptability.

Regular grids

A number of groups [32, 33, 34, 35, 36] have recently reported electronic structure calculations using a regular grid in real space. Such an approach achieves sparsity and parallelizability, but not adaptability. All terms in the Kohn-Sham Hamiltonian [2] are local in real space, except the Laplacian. When discretized using a real space grid, the Laplacian of a function at a point involves only the values of the function at nearby

points. Thus, the Hamiltonian is represented as a sparse matrix. A representation based on a regular grid in real space is also a very natural choice for a massively parallel computer architecture — assigning successive, equivalent blocks of space to processors configured in a 2 or 3 dimensional grid provides a natural and efficient mapping onto a parallel machine. Since the subgrids in each block of space require the same operations, load balance is automatically optimal. Since the nearby points that are needed in order to compute the Laplacian at a grid point in one processor are either located in the same processor or in neighboring processors, communication is local. Since all processors need the same information from their neighbors, communications is structured. Thus, good load balance and efficient communications are trivially achieved. However, a regular grid has uniform resolution in space and can not be adapted to inhomogeneous physical systems.

Local refinement

Adaptability can be added to a real space grid by increasing the number of basis elements in regions where more resolution is needed. This 'local refinement' approach is shown schematically in Fig. 3.1. The local refinement approach maintains the sparsity of a regular real space grid, but gives up trivial parallelizability in order to gain adaptability. Assigning each processor an equivalent region of space would causes some processors to get more grid points than others, and thus it would ruin the load balance. Assigning each processor an equal number of points would result in a complicated communications pattern that varies from processor to processor, and thus it would ruin the structure of the interprocessor communications. Recent work using the local refinement approach has been carried out in the context of wavelets [37, 38], finite-elements [39, 40], and multi-grid methods [41, 42].

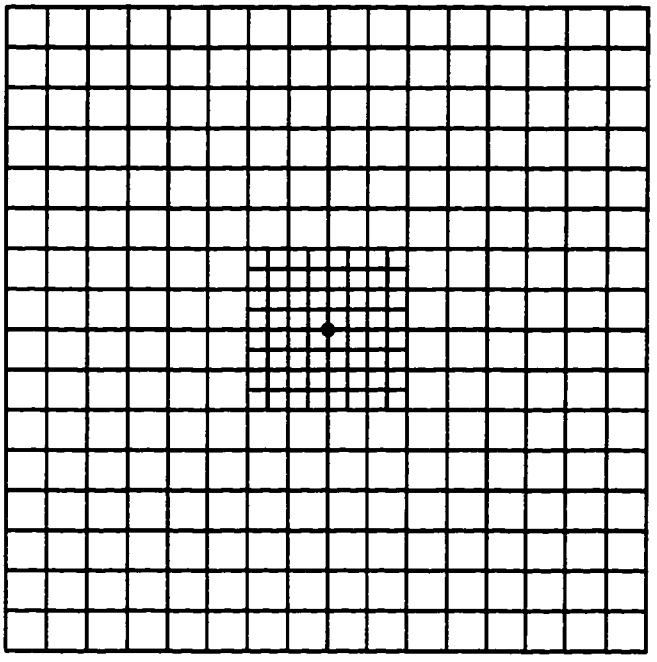

Figure 3.1: Schematic of a local refinement enhancement of the resolution in a difficult region near the center of the box. The intersections between the extra lines could indicate an added level of wavelets with a wavelet method, additional finite elements nodes in with finite element approach, or literally extra grid points with a multigrid method.

3.1.3 ACRES

In the following, we review our recently developed Adaptive Coordinate Real-Space Electronic Structure (ACRES) method, which achieves all three of the desirable properties discussed above: sparsity, parallelizability, and adaptability.

The central idea is shown in Fig. 3.2. We work on a regular grid in curvilinear space

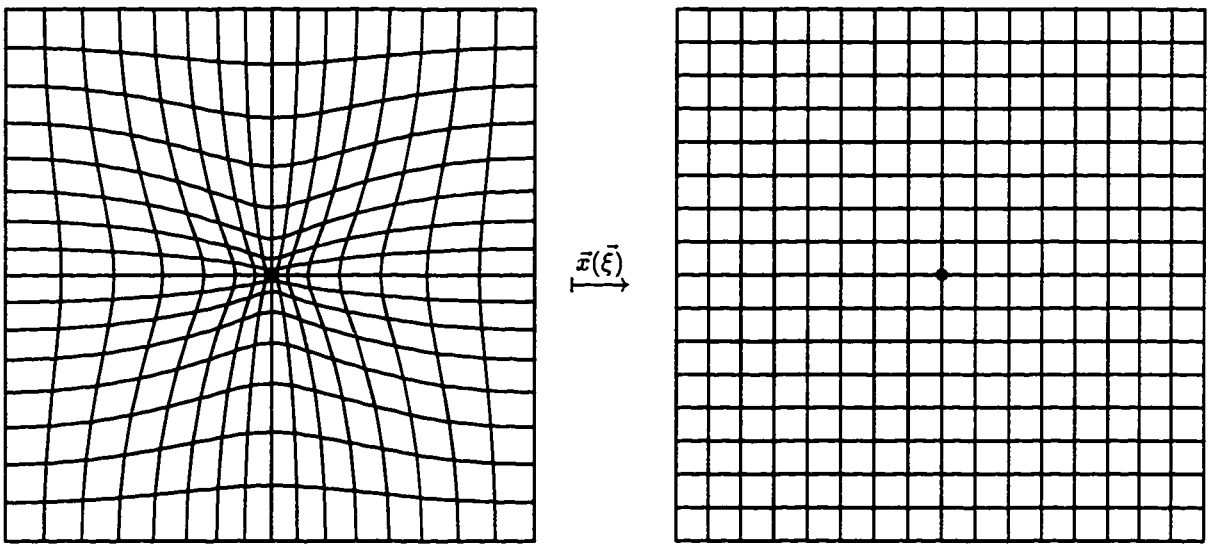

Figure 3.2: Schematic of the ACRES idea. The computations are done on a regular mesh in curvilinear coordinates, which is mapped by a change of variables to an adaptive mesh in real space.

$\vec{\xi}$, which is mapped by a change of coordinates $\vec{x}(\vec{\xi})$ to an adaptive mesh in real space $\vec{x}$ which is finer where high precision is needed. The coordinate transformation differs from "classical" coordinate systems such as spherical coordinates in two important

ways: (1) The grid is adapted to an arbitrary arrangement of atoms by taking a linear superposition of the adaptations associated with each atom. (2) The transformation is smooth and continuous everywhere and thus generates regular equations that are trivially parallelized. The combination of a real space approach and a coordinate transform is well established in such fields as fluid flow and heat transfer [43, 44, 45]. The type of coordinate transformation used in the ACRES method was pioneered in electronic structure calculations by F. Gygi [46] using a plane wave basis. This adaptive plane wave approach has been fruitfully pursued by several groups [46, 47, 48], but it does not produce a sparse Hamiltonian or parallelize particularly well. More recently, Gygi and Galli [49] have also studied a real-space approach using curvilinear coordinates and pseudopotentials.

3.2 Theory

We will now describe the method in more detail. First, a few notational remarks are necessary: We use the convention that Roman letters indicate real coordinate indices, and Greek letters indicate curvilinear coordinate indices. We also use the standard convention of upper indices for contravariant components, and lower indices for covariant components. We assume summation over repeated indices. The real space coordinates $x^i(\xi^\alpha; P^m)$ depend on the curvilinear coordinates ξ^α and on some set of parameters P^m that allow us to tune the change of coordinates to a particular problem. The Jacobian of the transformation is

$$J^i_\alpha(\xi; P) = \partial x^i / \partial \xi^\alpha \tag{3.1}$$

with $|J| = \det J$ its determinant. The trivial metric $g^{ij} = \delta^{ij}$ in real space becomes

$$g^{\alpha\beta} = J^{-1}{}^\alpha_i \delta^{ij} J^{-1}{}^\beta_j \tag{3.2}$$

in curvilinear coordinates. The Laplacian operator in curvilinear space is

$$\Delta = \frac{1}{|J|} \partial_\alpha \left(|J| g^{\alpha\beta} \partial_\beta \right), \tag{3.3}$$

and integrals are transformed according to $\int d^3x = \int d^3\xi \; |J|$.

The Coulomb energy between charge distributions ρ_1 and ρ_2 becomes

$$\int\int dx\, dx' \frac{\rho_1(x)\rho_2(x')}{|x-x'|} = \int d\xi \, |J| \, \rho_1(\xi) \, v_2(\xi) \tag{3.4}$$

with v_2 the solution of the Poisson equation

$$\Delta v_2(\xi) = -4\pi \, \rho_2(\xi). \tag{3.5}$$

Through these transformations, the problem has been entirely rewritten in terms of the curvilinear coordinates $\vec{\xi}$ — the physical space $\vec{x}$ has completely disappeared from the formulation of the problem. It is only when computing pseudopotentials or when plotting the density or wave functions that we need to consider the physical space $\vec{x}$.

Finally, the above equations are discretized in a box of linear size Λ_i, using a finite difference scheme on a regular grid in curvilinear space $\vec{\xi}$ with N_i points in each direction. Any boundary conditions can easily be implemented in this approach. In Section 3.3.6, we will discuss a set of boundary conditions that allow us to do multiple k-point calculations for periodic solids by generating the Bloch states that correspond to selected points in the Brillouin zone. For now, assume that periodic boundary conditions are used. From the mathematical point of view, we are just solving partial differential equations by discretizing on a regular mesh. Through our finite difference scheme, we have an approximation of the original equations, rather than a projection of the original problem onto a basis. Consequently, the variational aspect of a basis is lost, and our electronic eigenvalues are not necessarily upper bounds of the true electronic eigenvalues. Nevertheless, we will sometimes use the word 'basis' to describe the real space discretization.

3.3 Implementation

There are several choices involved in the implementation of the method. These include the form of the discretized Laplacian, the treatment of the Coulomb potential, the form

of the coordinate transform, the optimization of the coordinate transform, the method of calculating the forces, and the algorithms used.

3.3.1 The Laplacian

There are several ways of expressing the Laplacian in curvilinear coordinates. These expressions are equivalent in the continuum, but they are not necessarily equivalent after discretization. For example, one expression for the Laplacian is

$$\Delta = g^{\alpha\beta}\partial_\alpha\partial_\beta + A^\alpha\partial_\alpha \tag{3.6}$$

where $A^\alpha = 1/4\,\partial_\alpha \ln|g|$ is the connection associated with the metric g. Many standard numerical algorithms either work only for symmetric matrices or else require modifications that slow down their convergence for nonsymmetric matrices. Examples include the conjugate gradient and Lanczos algorithms. Therefore, it is desirable to have a self-adjoint (with respect to the measure $|J|$) discretization of the Laplacian. It is not possible to find a discrete representation of the derivative operator that makes Eq. (3.6) self-adjoint. For any representation, the difference between this Laplacian and its adjoint can be written in terms of the difference between the discrete and continuum derivatives, which clearly does not vanish. On the other hand, the Laplacian is also given by the expression

$$\Delta = \frac{1}{|J|}\partial_\alpha\left(|J|g^{\alpha\beta}\partial_\beta\right) \tag{3.7}$$

The discretization of this expression is self-adjoint if, for a fixed pair α,β, the discrete operators ∂_α and ∂_β are identical (and antisymmetric). For the $\alpha = \beta$ part of the operator, we take a representation of the derivative involving half-integer shifts, e.g., $(\partial f)_i = (f_{i+1/2} - f_{i-1/2})/h$ at first order. For the $\alpha \neq \beta$ part, we take a representation of the derivative based on integer shifts, e.g., $(\partial f)_i = (f_{i+1} - f_{i-1})/(2h)$ at first order. With these choices, Δf is expressed in terms of f evaluated only at the grid points, but the metric at half-integer points $g^{\alpha\beta}_{i+1/2}$ appears in the $\alpha = \beta$ part of the operator.

More generally, using the value of f at $2n$ points (integer or half-integer), we can define an order n (antisymmetric) representation $\partial^{(n)}$ of the derivative. A proper choice of the coefficients ensures that $\partial^{(n)} f = f^{(1)} + O\left(h^{2n} f^{(2n+1)}\right)$ where $f^{(p)}$ denotes the pth derivative of f. For example, at order $n = 2$, the half-integer representation is $(\partial^{(2)} f)_i = (-f_{i+3/2} + 27 f_{i+1/2} - 27 f_{i-1/2} + f_{i-3/2})/(24h)$, while the integer representation is $(\partial^{(2)} f)_i = (-f_{i+2} + 8 f_{i+1} - 8 f_{i-1} + f_{i-2})/(12h)$. The price for using a higher order representation is that more neighbors are needed in order to compute the Laplacian. Because of the off-diagonal term $\sim \partial_\alpha \partial_\beta$, the number of neighbors grows as $3((2n)^2 + 4n + 1)$. Since computing with a larger number of neighbors requires more communications between processors, using a high order representation of the derivative can be quite expensive on a massively parallel computer. Furthermore, storage of the discretized Laplacian is a major memory expense that grows with the number of neighbors. Clearly, there is a trade-off between the need to limit the time and memory required to compute the Laplacian, and the desire to achieve high accuracy with the minimum number of grid points.

Changing n changes both the expression for the Laplacian in the eigenvalue equation and the Coulomb potential (through the Poisson equation). This makes it quite difficult to analyze how errors in the final results depend on n. Thus, in order to determine a good value for n, we investigated a number of trial systems. The results of a set of all-electron computations for an H atom are shown in Fig. 3.3. We clearly see that $n = 1$ is not sufficient as the energy does not converge toward the large N limit as we increase the adaptation around the Coulomb singularity. Therefore, we need to take at least $n = 2$. In contrast, the results for $n = 2$ and $n = 3$ are essentially the same. Fig. 3.4 shows pseudopotential results for the same system. Again we find that $n = 1$ is not sufficient, while $n = 2$ and $n = 3$ give very similar results. In the pseudopotential case, we can clearly see that the error in the energy can have either sign since the finite difference approximation does not give a variational energy. Since we find little benefit

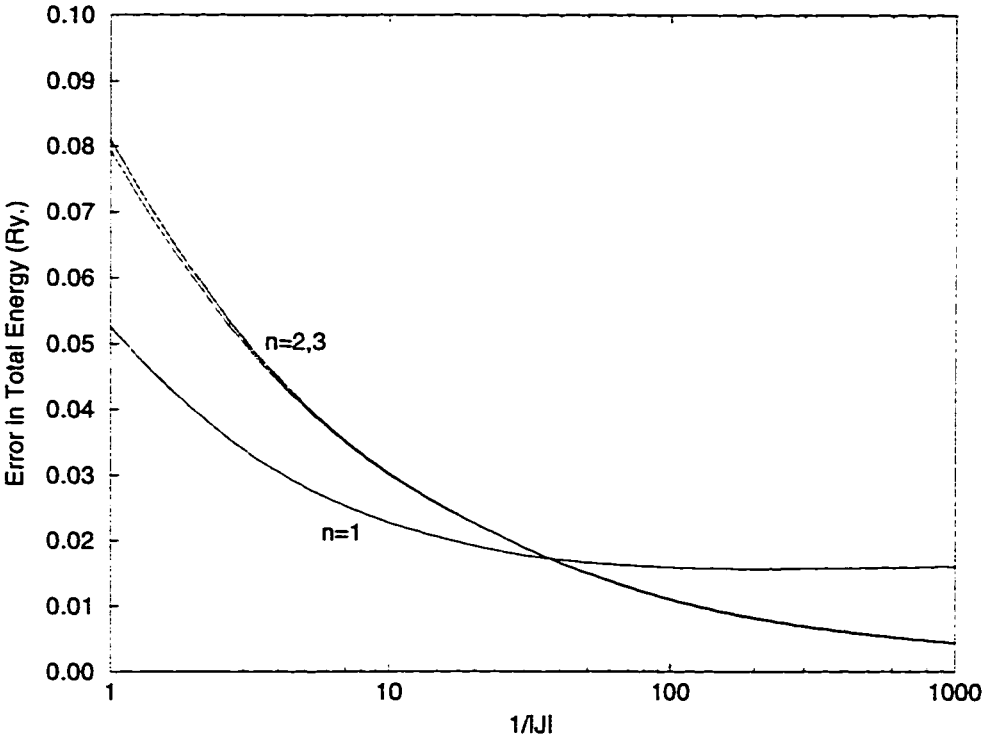

Figure 3.3: Convergence of the total energy as a function of the amount of adaptation for the first, second, and third order discretized derivatives. Results are from an all-electron calculation for a H atom using a grid with $32 \times 32 \times 32$ points.

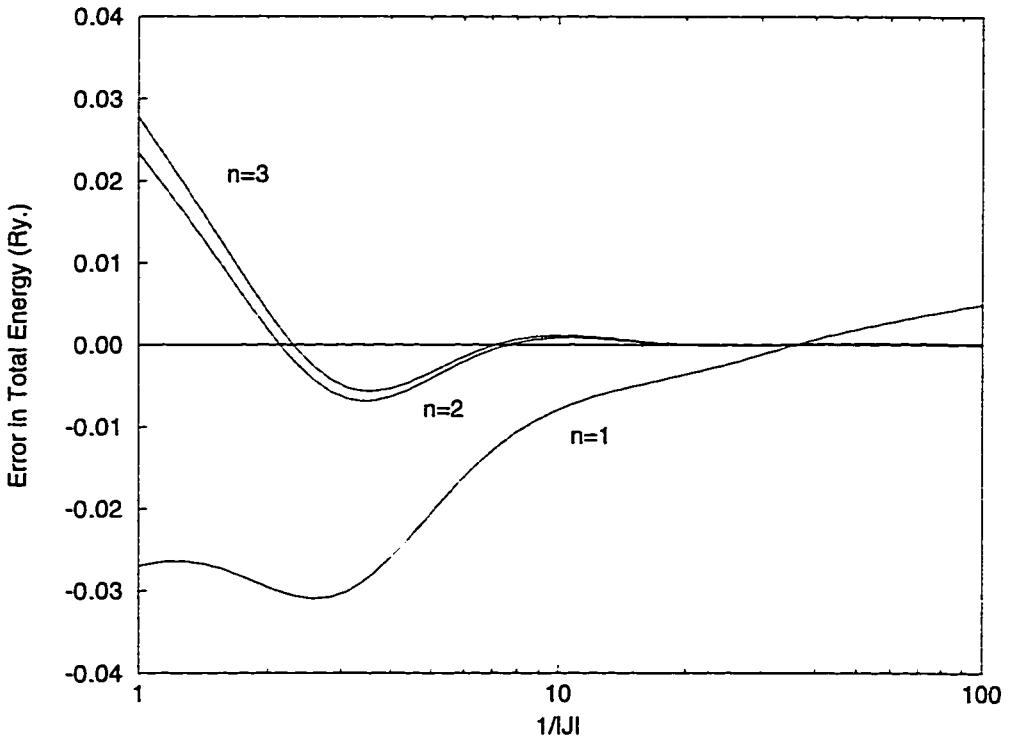

Figure 3.4: Convergence of the total energy as a function of the amount of adaptation for the first, second, and third order discretized derivatives. Results are from a pseudopotential calculation for a H atom using a grid with $32 \times 32 \times 32$ points.

in increasing to $n = 3$, there is no justification for the added expense of taking $n > 2$. Therefore, we use $n = 2$ for the rest of our computations.

3.3.2 The Coulomb potential

The grid breaks translational invariance, i.e., the total energy can vary as the entire system is moved around the box. This broken translational invariance also introduces an error in the dependence of the total energy on the position of each atom. Since computing accurate structural properties generally requires much more accurate relative energies than absolute energies, considerable care must be taken to minimize this effect. The errors introduced by broken translational invariance become negligible when the density of grid points is sufficiently high that the functions of interest (wavefunctions, density, potentials, etc.) are well represented. Adaptation increases the density of grid points in regions where such functions are rapidly varying, and therefore adaptatation reduces the effect of the grid. In the case of pseudopotential calculations, we find that moderate adaptation eliminates the translational invariance problem whenever the number of points in the grid is sufficient to give reasonably accurate absolute energies. The problem is more serious for all-electron computations because the ionic potential is divergent due to the Coulomb singularity. We call the distance between an atomic center and the nearest grid point the *offset*, and the total energy depends on the offset. If the ionic Coulomb potential is directly evaluated at the grid points (say by an Ewald sum), the energy diverges to minus infinity as the offset of any of the atoms goes to zero. The standard finite difference scheme interprets the value assigned to a given grid point as the actual value of the represented function at that point. With this interpretation, a singular function will never be well represented for any number of points. Clearly, it is necessary to regularize the Coulomb potential such that the value assigned to each grid point represents an average over the grid cell. A natural regularization is provided by solving numerically the Poisson equation [discretized in curvilinear coordinates by

means of the Laplacian, Eq. (3.3)] with the nuclear charge as the source. In addition to providing a natural regularization for the ionic potential, this approach also saves computational effort relative to an Ewald sum: It is already necessary to solve the Poisson equation numerically in order to find the Hartree potential, so we get the ionic potential for free simply by adding the ionic charge to the electronic charge density.

Computing the ionic Coulomb potential using the discretized Poisson equation requires a representation of the nuclear charge on the grid. For an atom with atomic number Z at position $\vec{R}$, the nuclear charge is $\rho(\vec{\xi}) = Z\,\delta(\vec{\xi};\vec{R})$, where $\delta(\vec{\xi};\vec{R})$ is a representation of a Dirac δ function at $\vec{R}$ on the regular grid in $\vec{\xi}$ space. There is some freedom in this representation as the distribution only needs to converge to the continuum δ function in the limit of the number on grid points going to infinity. Beside the normalization condition on the δ function,

$$\int d\xi\,|J|\,\delta(\vec{\xi};\vec{R}) = 1, \tag{3.8}$$

an important constraint on its representation on a finite grid is that the first moment of the distribution must correspond to the location of the δ function, i.e.,

$$\int d\xi\,|J|\,\delta(\vec{\xi};\vec{R})\,\vec{x}(\vec{\xi}) = \vec{R}. \tag{3.9}$$

This constraint is necessary in order to ensure that the nuclear charge appears to other atoms to be at the position $\vec{R}$, i.e., the tails of the resulting potential correspond to a $1/r$ divergence at $\vec{R}$. Without this constraint, there will be fluctuations in the apparent distances between atoms as the whole structure is translated around the box. Constraints on higher moments could also be imposed [50], but we did not find this to be necessary. The two constraints Eq. (3.8) and Eq. (3.9) still leave substantial freedom in the choice of $\delta(\vec{\xi};\vec{R})$. We experimented with a *linear interpolated* δ function defined as follows: For any function of the curvilinear coordinates $f(\vec{\xi})$, let $\tilde{f}(\vec{\xi})$ be the linear interpolation (in ξ space) of f between the points of the regular grid. Then, $\delta(\vec{\xi};\vec{R})$ is

chosen such that for an arbitrary f,

$$\int d\xi \, |J| \, \delta(\vec{\xi}; \vec{R}) \, f(\vec{\xi}) = \tilde{f}(\vec{\Xi}) \tag{3.10}$$

where $\tilde{x}_i(\vec{\Xi}) = R_i$ for i =1–3. We also experimented with a *Gaussian* representation of the delta function.

$$\delta(\vec{\xi}; \vec{R}) \propto \exp\left(-|\vec{\xi} - \vec{\xi}_0|^2 \,/\, 2\sigma^2 \Delta\xi^2\right) \tag{3.11}$$

with $\Delta\xi$ the regular grid spacing, σ an adjustable parameter, and $\vec{\xi}_0$ chosen to satisfy the constraint on the first moment of the distribution. Since its width is proportional to the grid spacing and thus goes to zero in the continuum limit, this small Gaussian is a good representation of the delta function on the discrete grid. Furthermore, by defining this representation in ξ space, we ensure that it goes to a continuum δ in the limit of strong adaptation. This δ is spread over a number of grid points that is set by σ independently of $\vec{x}(\vec{\xi})$, N, and Λ. Therefore, σ sets the rate at which δ approaches a continuum delta function as the number of grid points or the adaptation increases. For various values of σ, Fig. 3.5 shows the error in the total energy of a hydrogen atom as a function of the amount of adaptation. The curves correspond to an atom located either directly on a grid point, or as far as possible from a grid point. These cases were found to correspond to the minimum and maximum values of the total energy. Thus, for a given value of σ, the difference between the two lines indicates the maximum amount that the energy will vary as the atom is moved around the cell. As the value of σ is increased, this difference decreases rapidly, but there is also an upward shift in the energy. This upward shift is a result of replacing the nuclear charge distribution with a small Gaussian. Thus, there is a trade-off between reducing the broken translational invariance and avoiding a systematic error in the total energy. A value of $\sigma \simeq 0.6$ seems to be a good compromise. The linear delta function results in an even larger dependence of the energy on the atomic position than the $\sigma = 0.4$ Gaussian delta, and thus we found the Gaussian delta to be more useful. Note in Fig. 3.5 that adaptation further reduces

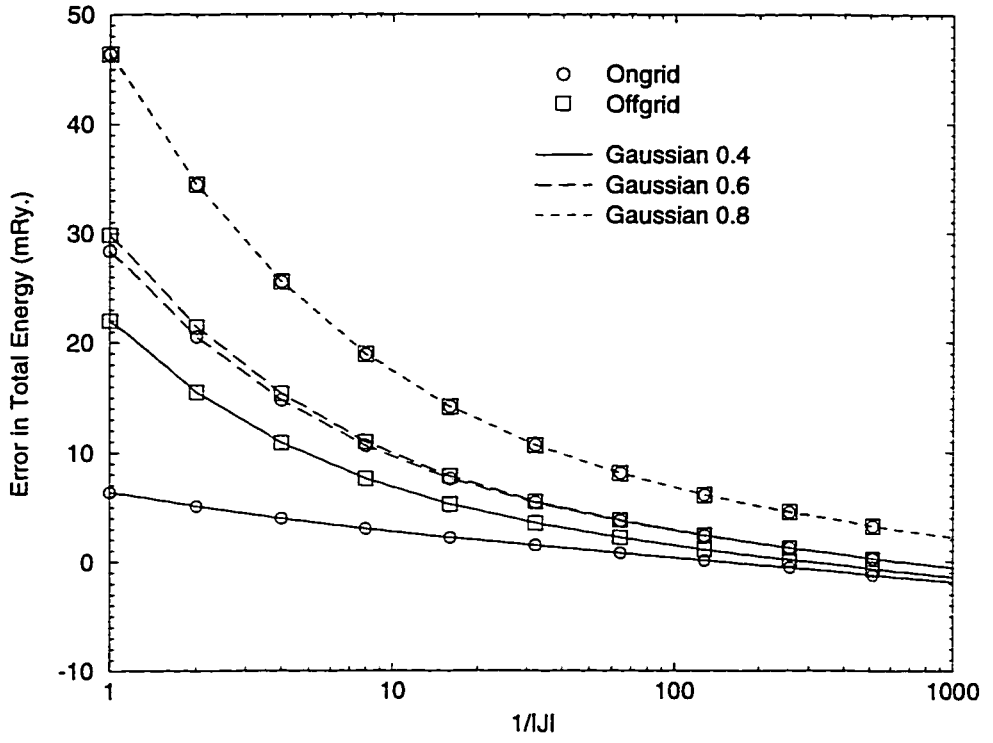

Figure 3.5: Error in the total energy of a hydrogen atom as a function of the amount of adaptation for all-electron calculations using a Gaussian delta function with various values of σ. The points marked by circles result from placing the atom directly on a grid point (zero offset), while the points marked by squares result from placing the atom as far as possible from a grid point (maximum offset). The supercell is 12 a.u. on each side, the grid has 64 points in each direction, no backdrop is used, and the adaptation radius is 0.5 a.u.

the dependence of the energy on the position of the atom. For grids of practical size (the grid used in Fig. 3.5 was chosen to be too coarse for accurate calculations in order to emphasize the effects discussed above), the combined use of strong adaptation, the Poisson regularization, and the Gaussian δ function eliminates the translation invariance problem.

3.3.3 The coordinate transformation

In the ACRES method, the burden of adapting the basis to the physics of a particular problem is swept into the change of coordinates $\vec{x}(\vec{\xi})$. An analytical mapping allows derived quantities such as J and $g^{\alpha\beta}$ to be calculated accurately. There are several conditions that must be satisfied by the coordinate transformation. The mapping between $\vec{x}$ and $\vec{\xi}$ must be single valued, i.e. the grid in x space must not be folded. Furthermore, since the Laplacian involves the derivative of the metric, and the metric is computed from the Jacobian, the mapping must be twice continuously differentiable on a 3D domain with periodic boundary conditions. It is also desirable that the mapping be spherically symmetric around an atom, at least to a good approximation (since we are working in a rectangular box with periodic boundary conditions, this cannot be an exact property). Even with all these requirements, there are still a vast number of possible choices of curvilinear coordinates. When necessary, we use a two level change of coordinates: a *global backdrop* and further *local adaptation* around each atom.

The global backdrop allows efficient long wavelength adaptation and is used only for systems with global inhomogeneities, such as large regions of vacuum. Typical systems where the backdrop is used are atoms, molecules, clusters, and surfaces. Since the backdrop consists of an independent change of coordinates in each direction $x^i = x^i(\xi^i)$, the grid cells remain rectangular. The functions $x^i(\xi^i)$ are chosen to create a central flat region of size $\bar{x}_i$ with a density of points increased by $1/\bar{J}_i$, surrounded by a rapidly decreasing density of points [51]. Thus, the parameters P associated with the backdrop

are $\bar{x}_i$ and $\bar{J}_i$ for i = 1-3. For rods or slabs of atoms, $\bar{x}_i = \Lambda_i$ in one or more directions. For a given direction, we take $x = \bar{J}\xi$ where $\bar{x} = \bar{J}\bar{\xi}$ for $0 < \xi < \bar{\xi}$,

$$x = \bar{J}\xi + \frac{\Lambda}{4}(1-\bar{J})\left(\frac{\xi-\bar{\xi}}{\Lambda/2-\bar{\xi}}\right)^q \left\{q+1-(q-1)\left(\frac{\xi-\bar{\xi}}{\Lambda/2-\bar{\xi}}\right)\right\} \qquad (3.12)$$

with $q = 3$ for $\bar{\xi} < \xi < \Lambda/2$, and $x(\xi) = -x(-\xi)$ for $-\Lambda/2 < \xi < 0$. In order to ensure continuity of the second derivative, $q > 2$ is required. Fig. 3.6 shows the backdrop used for a pseudopotential calculation for an O_2 molecule in a 12 × 12 × 24 a.u. box. Note

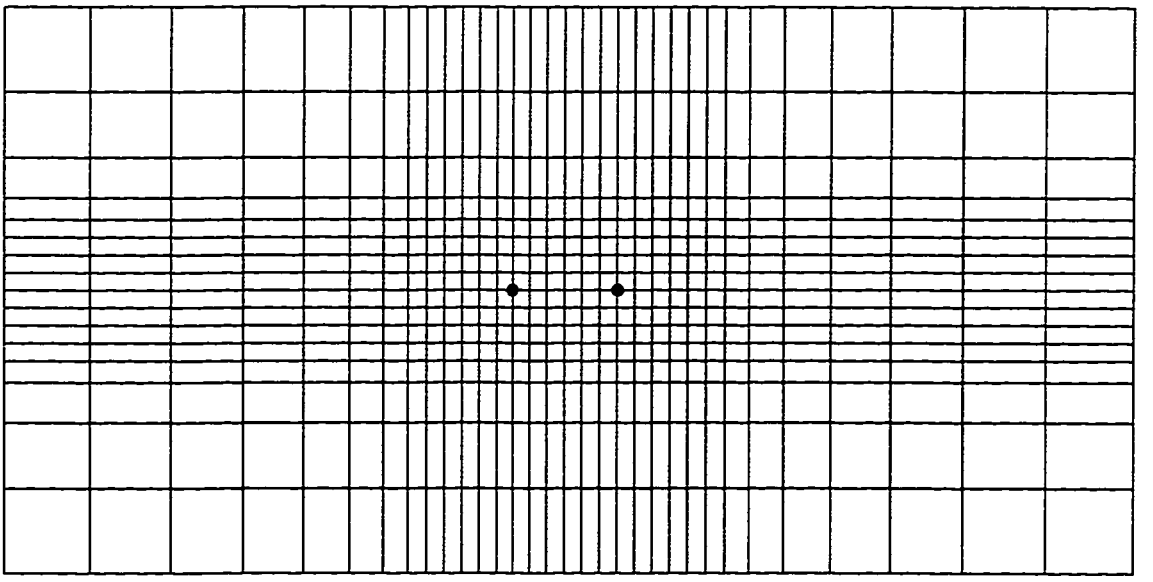

Figure 3.6: Backdrop used for a pseudopotential calculation for an O_2 molecule in a 12 × 12 × 24 a.u. box., in a horizontal cross-section through the atoms (every fourth line shown). The small dots indicate the locations of the atoms. The backdrop parameters are $\bar{x}_1 = 6$ a.u., $\bar{x}_2 = \bar{x}_3 = 3$ a.u., and $1/\bar{J}_i = 2$ for i = 1 - 3.

that the grid cells remain rectangular.

On top of the backdrop, the local adaptation creates a spherical deformation of the grid around each atomic center $\vec{R}_\nu$. For each atom ν, the parameters P associated with the local adaptation are the amount of adaptation $|J|_\nu$ at the position of the atom, and the radius of the adapted region κ_ν. In order to produce the local adaptation, we use a coordinate transformation with the form

$$\vec{x}(\vec{\xi}; P) = \vec{\xi} - \sum_\nu \sum_T f\left(|\vec{\xi} - \vec{\Xi}_\nu - \vec{T}|/\tau(|J|_\nu, \kappa_\nu)\right) \bar{\bar{Q}}_\nu \cdot (\vec{\xi} - \vec{\Xi}_\nu - \vec{T}) \qquad (3.13)$$

where the expressions $\vec{\Xi}_\nu$, $\bar{\bar{Q}}_\nu$, f, τ, and $\vec{T}$ are explained in following paragraphs.

The vectors $\vec{\Xi}_\nu$ are chosen so that $\vec{x}(\vec{\Xi}_\nu) = \vec{R}_\nu$, and the rank 2 tensors $\bar{\bar{Q}}_\nu$ are adjusted to obtain $J^i_\alpha(\vec{\Xi}_\nu) = |J|_\nu^{1/3}\, \delta^i_\alpha$. The values of $\vec{\Xi}_\nu$ and $\bar{\bar{Q}}_\nu$ must be found self-consistently. For each atom, $\vec{\Xi}_\nu$ contains 3 unknown scalars, while $\bar{\bar{Q}}_\nu$ contains 9 more. Correspondingly, the condition on the attraction center provides 3 constraints, while the condition on the Jacobian provides 9 more. Therefore, the number of unknowns is equal to the number of constraints. Since the system is nonlinear, this does not guarantee that a unique solution exists, but it is encouraging. In practice, we have had no trouble finding solutions for reasonable values of the parameters P. We use Jacobi relaxation to solve the system of equations, and we find that converging the solution takes only a very small proportion of the total time needed for an ACRES calculation. The above conditions ensure that the coordinate transformation near each atom is independent of the positions and parameters associated with all other atoms. This can be seen by considering a Taylor expansion of $\vec{x}(\vec{\xi})$ about the position of an atom. The conditions above determine the derivatives, and therefore they determine the function to first order in the displacement from the atomic position (the constant term is not determined, but it corresponds to a trivial shift in the underlying grid). This asymptotic independence of $\vec{x}(\vec{\xi})$ and the environment becomes very important when the adaptation regions overlap. If the coordinate transformation in the cores of

the atoms is allowed to depend on the arrangement of the atoms, any errors in the energy due to underconvergence of the calculation in the atomic cores will also depend on the arrangement of the atoms. This will introduce an extra dependence of the total energy on the atomic positions. Since errors in relative energies are generally much more serious than systematic errors in absolute energies, calculated structural properties may become inaccurate. Furthermore, without enforcement of the condition on the Jacobian described above, the grid cells between two atoms quickly become very elongated along the axis between the atoms as the overlap of the adaptation regions is increased. Soon the grid overlaps, and the grid generation procedure fails. When all values of κ_ν are much smaller than the distances between atoms, simplifications of the above procedure can be used. One possibility is to take $\overline{\overline{Q}}_\nu$ proportional to the identity matrix and only require $|J(\vec{R}_\nu)| = |J|_\nu$. A further simplification is provided by approximating $\vec{\Xi}_\nu = \vec{R}_\nu$. Using these simplifications can save a small amount of time when the adaptation regions do not overlap. Higher order generalizations of Eq. (3.13) can be imagined also: $\overline{\overline{Q}}_\nu \cdot (\vec{\xi} - \vec{\Xi}_\nu - \vec{T})$ could be replaced with a higher order polynomial in $\vec{\xi}$, and the coefficients could be determined by matching higher order derivatives. Taylor's Theorem guarantees that the number of coefficients will be the same as the number of derivatives at any order. Since such higher order generalizations would require solving more complicated equations for more unknown coefficients, they would be more expensive. The form of adaptation given in Eq. (3.13) has worked well for all the systems that we have studied, so we have not tried to implement the generalizations.

The function f can be any rapidly decaying function with zero derivative at the origin. All the computations presented below have been done with the simple Gaussian form $f(x) = \exp(-x^2/2)$. We also experimented with an exponential form $f(x) = 1/\cosh(x)$. The results are quite similar, but slightly favor the Gaussian form. The function $\tau(|J|_\nu, \kappa_\nu)$ has been chosen such that for one atom sitting at the origin, κ

corresponds to the radius of the adapted region independently of $|J|_\nu$, i.e.,

$$1 - \det J(\kappa) = (1 - \det J(0))/2. \tag{3.14}$$

In Eq. (3.13), $\vec{T}$ is summed over enough lattice vectors to make $\vec{r}(\vec{\xi})$ effectively periodic. Since the adaptation range κ is generally much smaller than the box size A, we have found that two lattice vectors in each direction is sufficient.

Fig. 3.7 shows the combined effect of the backdrop and the local adaptation for a pseudopotential calculation for an O_2 molecule in a $12 \times 12 \times 24$ a.u. box. Note that despite the significant overlap of the adaptation regions, the grid cells remain nearly square close to the atomic positions.

3.3.4 Optimization of the change of coordinates

A question of central importance is how to choose the parameters that determine the change of coordinates. Consider the case of one atom. Fig. 3.8 shows the error in the total energy of an O atom as a function of the amount of adaptation for three different values of the adaptation radius. These results can be viewed in terms of a competition between a large negative error in the energy associated with the region very close to the nucleus and a smaller positive error in the energy associated with the region further away from the nucleus. The first error decreases rapidly with adaptation as more points are moved into the core region. This results in a sharp initial increase in the energy. The second error may either increase or decrease with adaptation depending on the value of the adaptation radius. For a small value of the adaptation radius, the density of points in the region responsible for this error decreases. Therefore, the second type of error increases, and the total energy does not converge to the correct energy for large amounts of adaptation. On the other hand, for a large value of the adaptation radius, the density of points increases throughout the region where functions need to be accurately represented in order to get accurate total energies, and both types of error decrease with adaptation. These situations are demonstrated by Fig. 3.8.

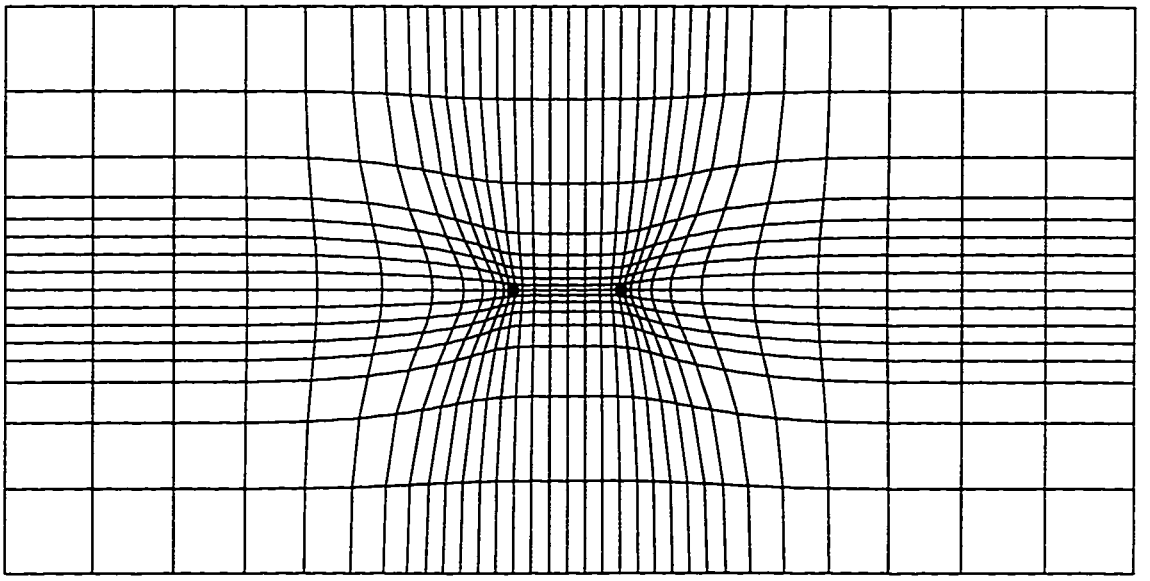

Figure 3.7: Grid used for a pseudopotential calculation for an O_2 molecule in a 12 × 12 × 24 a.u. box., in a horizontal cross-section through the atoms (every fourth line shown). Notice the effect of the global backdrop (crosslike region with many grid points) and the local adaptation around each atom. The backdrop parameters are $\bar{x}_1 = 6$ a.u., $\bar{x}_2 = \bar{x}_3 = 3$ a.u., and $1/\bar{J}_i = 2$ for $i = 1 - 3$. For each atom, the local adaptation parameters are $1/|J|_\nu = 32$ and $\kappa_\nu = 1.4$ a.u. The spacing between the atoms is 2.28 a.u.

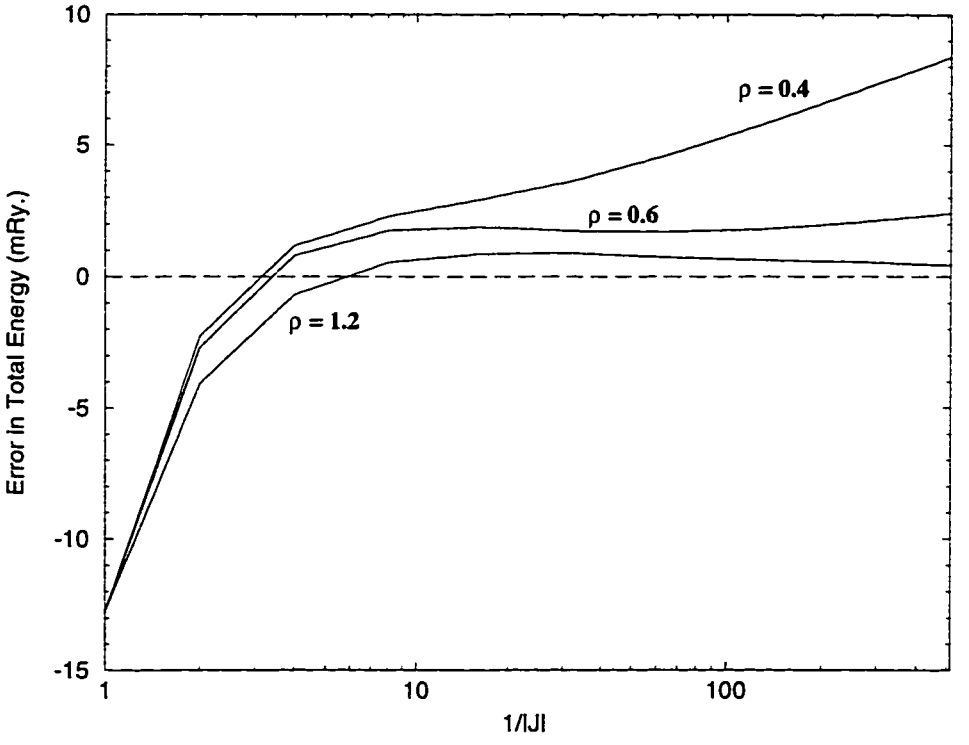

Figure 3.8: The error in the total energy of an O atom as a function of the amount of adaptation for three different adaptation radii κ. The results are from a pseudopotential calculation using a $48 \times 48 \times 48$ grid and a box 6 a.u. on each side. No backdrop was used. The exact large N limit of the energy was approximated using a $96 \times 96 \times 96$ grid.

The $\kappa = 0.4$ a.u. curve in Fig. 3.8 shows the effects of an overly small adaptation radius. For moderate to large amounts of adaptation, the error in the energy is dominated by regions further than 0.4 a.u. from the radius. Since this error gets worse with increasing adaptation, the total energy continues to increase and diverges away from the correct energy for large adaptations. Since the two types of errors have an opposite sign, and the larger one decreases while the smaller one increases, the total error passes through zero. However, this is an accidental cancellation, and the important functions are not particularly well represented for this amount of adaptation. Therefore, other quantities, such as the forces, will not be especially accurate at the point where the error in the total energy cancels. Furthermore, since the total energy changes relatively quickly with adaptation near this point, this cancellation of errors is fragile. Perturbations in the coordinate transformation due to other atoms should ruin the cancellation. Since such perturbations depend on the locations of the other atoms, this would introduce extra errors in relative energies and derived structural properties. Therefore, it is not a good idea to try to choose the amount of adaptation in such a way that the errors in the total energy cancel exactly.

In Fig. 3.8, the $\kappa = 1.2$ a.u. curve shows typical results for a large adaptation radius. In this case, the atom essentially lies within the adaptation region. Therefore, both types of error decrease with increasing adaptation. Since the large error due to the core region decreases much more rapidly than the error from further out, the total error still passes through zero at some point. Again, this cancellation is accidental, and there is nothing particularly special about this point. Once the error due to the core region becomes negligible, the error in the total energy decreases slowly with increasing amounts of adaptation, and the total energy converges toward the large N value. Since the density of points near the atom continues to increase with adaptation, all important functions become better and better represented, and all derived quantities should become accurate.

The number of points in the adaptation region grows as $\kappa^3/|J|$. Furthermore, in systems with more than one atom, the various adaptation regions must compete for points. Therefore, it usually is not practical to work with such a large adaptation radius that the atoms are completely within the adaptation regions. A typical compromise is shown by the $\kappa = 0.6$ curve in Fig. 3.8. Here, the total energy diverges away from the correct energy for very large adaptations, but there is a broad region of adaptation parameters where the error in the energy is small and nearly constant. Since all of the important functions are fairly well represented throughout this region, derived quantities, such as the forces, are also accurate.

The discussion above shows that we want to put more points around the nucleus in order to describe the potential and wave functions more accurately, but we do not want to deplete the tails too much. There is clearly a trade-off, and we want a quantitative criterion to help us choose a good compromise. In order to generate a nearly optimal mesh for a given physical problem, we define a *merit functional* $m(f; P)$ that measures how well a function f is represented by the grid, and choose the parameters P that minimize this quantity. Since our approach does not use a basis in a Hilbert space, the computed energy is not an upper bound to the ground state, and a minimal energy optimization as used by Gygi [46] in the adaptive plane wave approach is not possible. To motivate our choice of a merit functional, consider an estimate of the error in a periodic, one-dimensional integral $I(f) = \int dx\, f(x) = \int d\xi\, \tilde{f}(\xi; P)$ with $\tilde{f}(\xi; P) = |J|(\xi; P)\, f(x(\xi; P))$. The integral is computed numerically on a regular mesh in ξ coordinates

$$I_N(f; P) = \sum_i \Delta\xi\, \tilde{f}(\xi_i; P) \tag{3.15}$$

with mesh spacing $\Delta\xi$. Due to the definition of $\tilde{f}$, the numerical integral depends on the coordinate transformation both through the Jacobian factor and through the points at which f is evaluated. We warn the reader against a pitfall encountered when estimating the error in a numerical integral: the second Euler-Maclaurin summation

formula, as given in most textbooks, is only an asymptotic formula and can lead to obvious contradictions. For example, when applied to the discretized integral of an infinitely differentiable periodic function, as in our case, it gives identically zero for the error, independently of N ! Therefore, we have to develop another method. Without actually knowing the precise value of $I(f)$, we want to choose P so that $I_N(f;P)$ is the best possible approximant of $I(f)$. We could use I_{2N} as an approximant for I, but this would require computation on the finer grid. Our solution is to use I_N as an approximant for I and to compare its value to $I_{N/2}$. By optimizing the parameters P, an optimal grid of size $N/2$ is generated, and the same set of parameters should also generate a good grid at size N. Fig. 3.9 shows the contribution to an integral from an elementary volume of the coarser grid, and the contribution from the same region evaluated with the finer grid. With $N/2$ points, the rectangular element of integration is $\delta I_{N/2} = 2\Delta\xi\, \tilde{f}(\xi_i;P)$. With

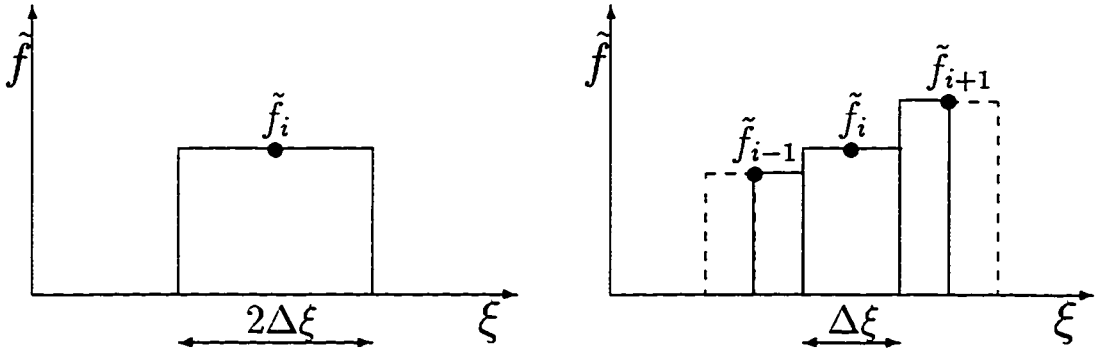

Figure 3.9: The contributions to an integral from an elementary region of space evaluated using a regular grid with $N/2$ points and a regular grid with N points. $\Delta\xi$ indicates the grid spacing.

N points, the same region contributes $\delta I_N = \Delta\xi\,[\tilde{f}(\xi_{i-1};P)/2 + \tilde{f}(\xi_i;P) + \tilde{f}(\xi_{i+1};P)/2]$ to the integral. An estimate of the error contributed by this region is given by

$$\delta e(f) = \delta I_N - \delta I_{N/2} = \Delta\xi^3\, \tilde{f}_i''/2. \tag{3.16}$$

This expression is similar to a rigorous upper bound due to Peano [52]: The error in a one-dimensional integral evaluated by using the simple trapezoidal rule is bounded by

$$\delta e(f) = \frac{\Delta\xi^3}{12} \max_{a<\xi<b} f''(\xi) \tag{3.17}$$

with $a = \xi_i - \Delta\xi/2$ and $b = \xi_i + \Delta\xi/2$.

To avoid cancellation of errors from different regions of space, we take our merit functional to be

$$m(f;P) = \left(\sum_i (\delta e)^2\right)^{1/2} = \frac{1}{2}\,\Delta\xi^{5/2}\left(\sum_i \Delta\xi\,(\tilde{f}_i'')^2\right)^{1/2}. \tag{3.18}$$

This provides a measure of how well the grid represents the function f and is suitable for minimization with respect to P. In Eq. (3.18), the expression is separated in such a way that the quantity inside the parentheses is the numerically evaluated integral of a function. Therefore, this part of the expression should approach a constant as the number of points is increased. An alternative view of $m(f;P)$ given by Eq. (3.18) is that it is proportional to the squared norm of the difference between f and the function obtained by interpolating f between the neighboring points. The above idea can be easily generalized to three-dimensional integrals (whereas the rigorous Peano bound is difficult to extend to higher dimensions). If we define

$$D^2 f = 7\sum^{8} f_{corner} - 2\sum^{12} f_{edge} - 4\sum^{6} f_{face} - 8 f_{center}, \tag{3.19}$$

where $D^2 f$ depends on the 27 values of f in a cube around the point (i,j,k), we obtain

$$m(f;P) = \frac{\Delta\xi_i\Delta\xi_j\Delta\xi_k}{8}\left(\sum_{i,j,k}(D^2\tilde{f}_{i,j,k})^2\right)^{1/2}. \tag{3.20}$$

In order to obtain the scaling dependence of Eq. (3.20) in the continuum limit, consider a grid such that $\Delta\xi_i = \Delta\xi_j = \Delta\xi_k = \Delta\xi = (V/N)^{1/3}$ where V is the volume of the system and N is the total number of points in the grid. Then,

$$D^2 f = (\Delta\xi)^2\,\nabla^2 f, \tag{3.21}$$

and substitution gives

$$m(f;P) = \frac{1}{8}\left(\frac{V}{N}\right)^{7/6}\left(\sum_{i,j,k}\Delta\xi^3(\nabla^2\tilde{f}_{i,j,k})^2\right)^{1/2} \tag{3.22}$$

Once more, the terms have been collected in such a way that the expression in parentheses is the integral of an ordinary function and should approach a constant as the number of grid points increases. Therefore, the merit functional should scale inversely with the 7/6 power of the density of points. If there was no cancellation of errors from different regions of space, the convergence of finite grid results toward their continuum limits would have the same scaling, but a substantial amount of cancellation of errors generally occurs for functions with periodic boundary conditions.

The last step of using our merit functional is picking a function f that produces a good grid when $m(f;P)$ is minimized. For several atoms where we knew the large N limit, we carried out calculations with a range of grid parameters P. We compared the resulting errors in the energy with the functions $m(f;P)$ obtained using various functions f. Fig. 3.10 and Fig. 3.11 show $m(f;P)$ from pseudopotential calculations for an O atom. The adaptation radius κ is fixed at the small value 0.4 a.u. in Fig. 3.10, while it was increased to the large value 1.2 a.u. in Fig. 3.11. Note that the larger radius generally gives much smaller values of $m(f)$, indicating better represented functions, at moderate to strong adaptation. This is in agreement with the errors in the total energies given in Fig. 3.8. As expected, different functions are represented best at somewhat different values of the parameters. However, since the minima are quite broad, there is a range of parameters where all of the important functions are reasonably well represented. This indicates that the results of a calculation should not be very sensitive to the exact values of the grid parameters as long as reasonable values are used.

It is natural to consider choosing f to be an energy density. For example, the band energy density can be defined as

$$e_{band}(\vec{x}) = \sum_i \Psi_i(\vec{x})[\hat{H}\Psi_i](\vec{x}) = \sum_i \epsilon_i|\Psi_i(\vec{x})|^2 \tag{3.23}$$

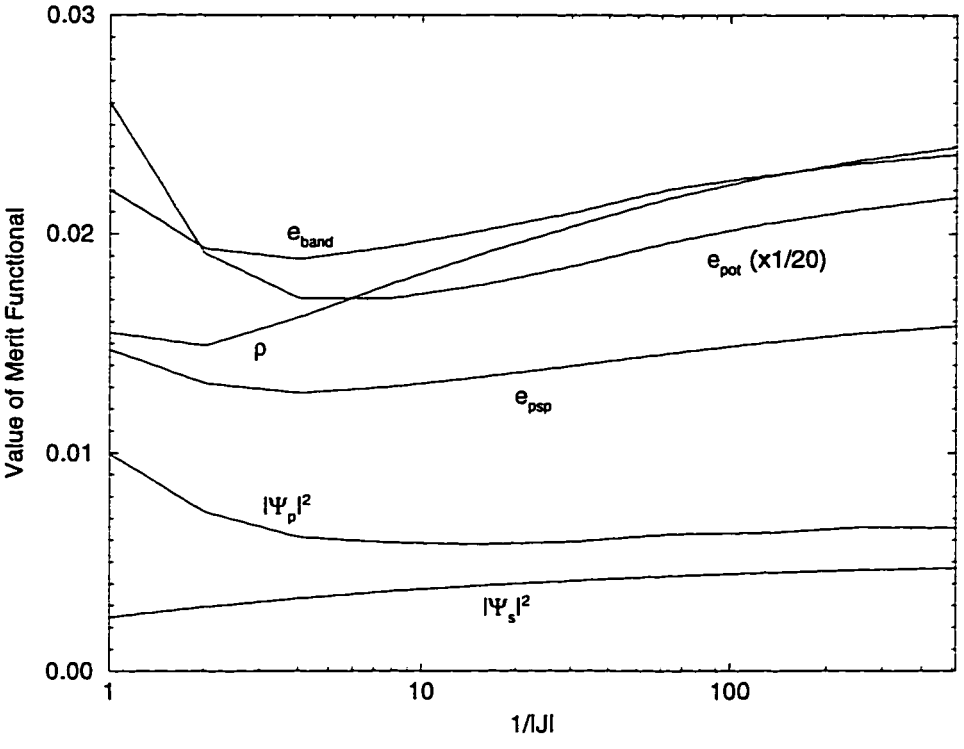

Figure 3.10: Merit functional $m(f;P)$ as a function of the amount of adaptation for various arguments f. The adaptation radius κ is fixed at 0.4 a.u. The results are from a pseudopotential calculation for an O atom in a $6\times6\times6$ a.u. box using a $48 \times 48 \times 48$ grid. The functions f are: e_{band}, the band energy density; e_{pot}, the potential energy density; e_{psp}, the pseudopotential energy density; ρ, the electronic charge density; $|\Psi_s|^2$, a (squared) s wavefunction; and $|\Psi_p|^2$, a (squared) p wavefunction. $m(e_{pot})$ has been divided by a factor of 20 in order to fit it on the scale of the other functionals.

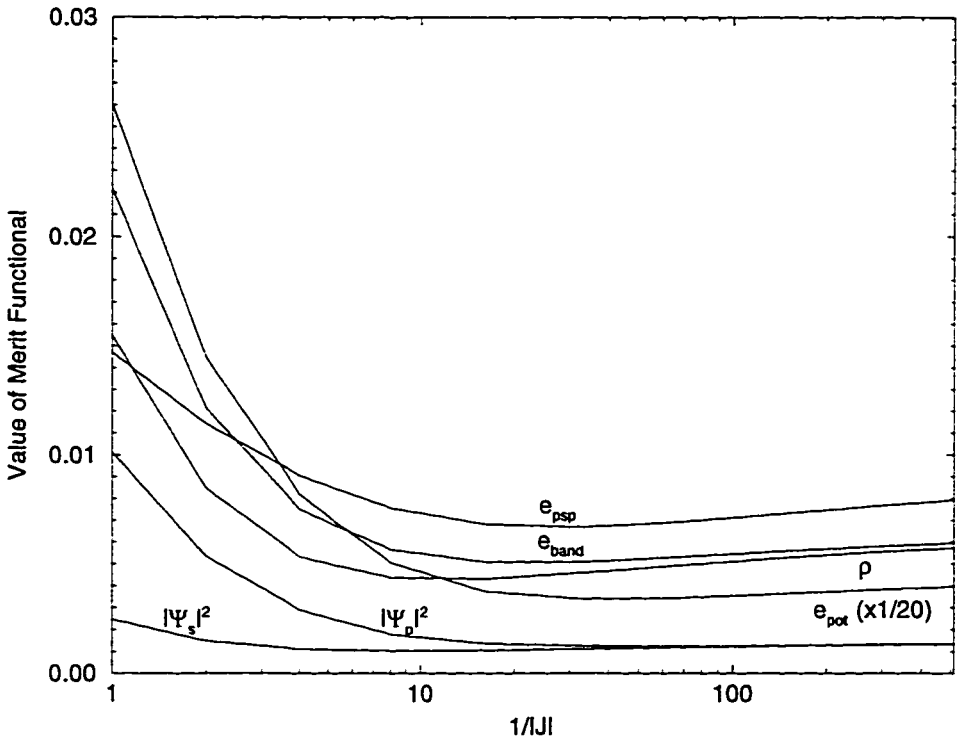

Figure 3.11: Merit functional $m(f;P)$ as a function of the amount of adaptation for various arguments f. The adaptation radius κ is fixed at 1.2 a.u. The results are from a pseudopotential calculation for an O atom in a $6\times6\times6$ a.u. box using a $48 \times 48 \times 48$ grid. The functions f are: e_{band}, the band energy density; e_{pot}, the potential energy density; e_{psp}, the pseudopotential energy density; ρ, the electronic charge density; $|\Psi_s|^2$, a (squared) s wavefunction; and $|\Psi_p|^2$, a (squared) p wavefunction. $m(e_{pot})$ has been divided by a factor of 20 in order to fit it on the scale of the other functionals.

where $\hat{H}$ is the Kohn-Sham Hamiltonian, and the sums are over occupied states. By substituting the appropriate operator for $\hat{H}$, we can similarly define a kinetic energy density e_{kin}, a potential energy density e_{pot}, and a pseudopotential energy density e_{psp}. For typical systems, both e_{kin} and e_{pot} are much larger in magnitude than e_{band}. For low energy states, this near cancellation of the kinetic and potential terms is forced by the Kohn-Sham equation. The relatively large magnitude of e_{pot} is reflected in its merit functional, and the values of $m(e_{pot})$ in Fig. 3.10 and Fig. 3.11 had to be reduced by a factor of 20 to fit on the same scale as the rest of the functions. Furthermore, since e_{band} is comparatively small, $e_{kin} \approx -e_{pot}$, and $m(e_{kin}) \approx m(e_{pot})$. For this reason, we have not included $m(e_{kin})$ separately from $m(e_{pot})$ in Fig. 3.10 and Fig. 3.11. We find e_{band} to be a useful indicator of grid quality since it captures the main sources of error that do not cancel between the kinetic and potential terms. The wavefunctions must be well represented in order to get accurate results, but the merit functional of an individual wavefunction (such as $m(|\Psi_s|^2)$ or $m(|\Psi_p|^2)$ in Fig. 3.10 and Fig. 3.11) does not provide a useful indicator of the overall quality of the grid. In order to combine the various wavefunctions into one useful result, we can consider the merit functional of the electronic density $m(\rho)$. However, Eq. (3.23) ensures that $m(\rho)$ behaves in a fashion similar to $m(e_{band})$. Finally, the pseudopotentials must also be well represented, and checking the merit functional for the pseudopotential energy density e_{psp} ensures that this is the case.

For the larger adaptation radius of Fig. 3.11, the various functions indicate somewhat different adaptation strengths, but the minima are very broad and the resulting values of the parameters are not significantly different. In general, an adaptation strength of $1/|J| = 8 - 64$ appears reasonable. Comparison to Fig. 3.8 shows that the error in the total energy is small and nearly constant throughout this region. Even though the error in the total energy decreases at very large adaptations, the merit functionals increase very slowly. This indicates that the representation of important functions is

getting worse, and it is not a good idea to use such very strong adaptations. Minimizing $m(f;P)$ indicates that substantially weaker adaptation should be used with the smaller adaptation radius of Fig. 3.10. The minima in $m(f;P)$ are not as broad as they were with the larger radius, and the minima do not overlap as well for various different choices of f. Still, the minima are roughly in the same region, and an adaptation strength of $1/|J| = 2 - 8$ appears reasonable. Reference to Fig. 3.8 shows that the indicated range of adaptation strengths gives the smallest error in the total energy and also the slowest variation of the error with respect to the adaptation parameters. Thus, the best possible compromise of parameters can be found using the merit functional, even when no optimal choice of parameters is available.

Since the expression for the merit functional Eq. (3.20) is nonlinear, it is a difficult problem to minimize $m(f;P)$ using results computed for only a small number of values of P. Since calculations for realistic systems are expensive, only a small number of results are generally available. Therefore, we do not attempt to minimize $m(f;P)$ for large systems. Instead, by studying $m(f;P)$ for single atoms, we find optimal parameters for each atomic species. Then, further computations are done using this fixed set of parameters.

Finally, for pseudopotential computations, the grid generated using the present approach can be compared with the one obtained with the minimal energy scheme in adaptive plane waves used by Gygi [46]. We systematically obtain grids that are more adapted. For example, using a similar box and grid size for oxygen, the linear grid size Δx at the atom location is reduced by a factor ~ 2 using Gygi's method, whereas we use a factor ~ 6 (2 for the backdrop and 3 for the local atomic adaptation). This difference likely originates in our emphasis on making sure that all important functions are well represented rather than just focusing on the total energy. The elastic constants μ used by Gygi in order to prevent the folding of the grid should also lead to less adapted grids.

3.3.5 The forces

The total force $\vec{F}_\nu^{TOT}$ on the ion ν is the sum of the ion-ion force $\vec{F}_\nu^{II}$ and the ion-electron force $\vec{F}_\nu$. The ion-ion contribution is evaluated using an Ewald sum. In order to evaluate the electronic contribution to the force, define

$$\tilde{H} = \hat{H} - \frac{1}{2}V_H(\rho) + \epsilon_{XC}(\rho) - V_{XC}(\rho), \tag{3.24}$$

where ρ is the electronic density, $\hat{H}$ is the Kohn-Sham Hamiltonian, $V_H(\rho)$ is the Hartree potential, $\epsilon_{XC}(\rho)$ is the exchange-correlation energy density per electron, and $V_{XC}(\rho)$ is the exchange-correlation potential. Then, the electronic contribution to the force is given by

$$\vec{F}_\nu = -\frac{d}{d\vec{R}_\nu}\sum_i \langle\Psi_i|\tilde{H}|\Psi_i\rangle. \tag{3.25}$$

In principle, since the grid changes as the atoms are moved, it is necessary to include Pulay corrections in the calculation of $\vec{F}_\nu$. If the basis is viewed as fixed, while the changes in the grid are swept into the Hamiltonian, an analog of the Hellmann-Feynman Theorem can be proven using that each eigenvector Ψ_i is an extremum of $\langle\Psi_i|\hat{H}|\Psi_i\rangle$, and ρ is an extremum of the total energy. In particular,

$$\vec{F}_\nu = -\sum_i \langle\Psi_i|\frac{d\tilde{H}}{d\vec{R}_\nu}|\Psi_i\rangle \tag{3.26}$$

where the derivative with respect to $\vec{R}_\nu$ is taken with the other atomic positions and ρ fixed, but with the coordinate transform $\vec{x}(\vec{\xi})$ allowed to change with $\vec{R}_\nu$. Expanding out the derivative gives

$$\vec{F}_\nu = -\sum_i \langle\Psi_i|\frac{\partial\tilde{H}}{\partial\vec{R}_\nu} + \frac{\partial\vec{x}(\vec{\xi})}{\partial\vec{R}_\nu}\frac{\partial\tilde{H}}{\partial\vec{x}(\vec{\xi})}|\Psi_i\rangle. \tag{3.27}$$

Only the local atomic potential V_ν and the pseudopotentials depend explicitly on $\vec{R}_\nu$, and the first term gives the standard Hellmann-Feynman force. The long-range part of the Hellmann-Feynman force can be expressed as

$$\vec{F}_\nu^{HF} = \sum_i \langle\Psi_i|\frac{\partial V_\nu}{\partial\vec{R}_\nu}|\Psi_i\rangle = \int d\xi |J|\, V_H\frac{\partial\rho_\nu}{\partial\vec{R}_\nu} \tag{3.28}$$

where V_H is the Hartree potential and ρ_ν is the ionic charge distribution for the atom ν. V_H is found by solving the discretized Poisson equation $\Delta V_H = -4\pi e^2 \rho$, and the derivatives $\frac{\partial \rho_\nu}{\partial \vec{R}_\nu}$ are computed analytically. This form has the advantage of requiring the solution of only one Poisson equation for any number of atoms in the system. Even though the ion-electron energy is found by integrating the discretized ionic potential times the electronic charge, while Eq. (3.28) involves the discretized electronic potential times the ionic charge, Eq. (3.28) gives the exact derivative of the ion-electron energy. This is because the *discretized* Laplacian is a symmetric matrix (with respect to the measure ρ), and therefore its inverse is also symmetric. A nonsymmetric Laplacian would require modification of Eq. (3.28). For pseudopotential calculations, there are additional non-local short range parts in the Hellmann-Feynman force that are straightforwardly evaluated by differentiating the pseudopotential. The pseudopotential is interpolated from values tabulated on a logarithmic grid, and it is important to differentiate the interpolation of the pseudopotential in order to get forces that are exact derivatives of the total energy. Our Hellmann-Feynman force is accurately evaluated, and we have verified that it gives the exact derivative of the energy when the grid does not change. The second term in Eq. (3.27) gives the Pulay corrections to the forces. Virtually every term in $\tilde{H}$ depends on $\vec{x}(\vec{\xi})$ either through the Jacobian weight $|J|$, through the discretized Laplacian, or directly through the points at which a function is evaluated (i.e. the pseudopotentials). We have implemented the evaluation of all of these terms. Fig. 3.12 shows the Hellmann-Feynman and Pulay corrected forces from an all-electron calculation for an H_2 molecule along with the numerically evaluated derivative of the energy. With the Pulay corrections, we obtain forces that reproduce the derivative of the energy with excellent accuracy. For this computation, we used a $64 \times 32 \times 32$ grid in a $24 \times 12 \times 12$ a.u. box, and the Hellmann-Feynman force differs from the derivative of the energy by about 0.01 Ry. per a.u. Considering that this grid contains only 1/64 the number of points that we would use in a high accuracy computation for this system,

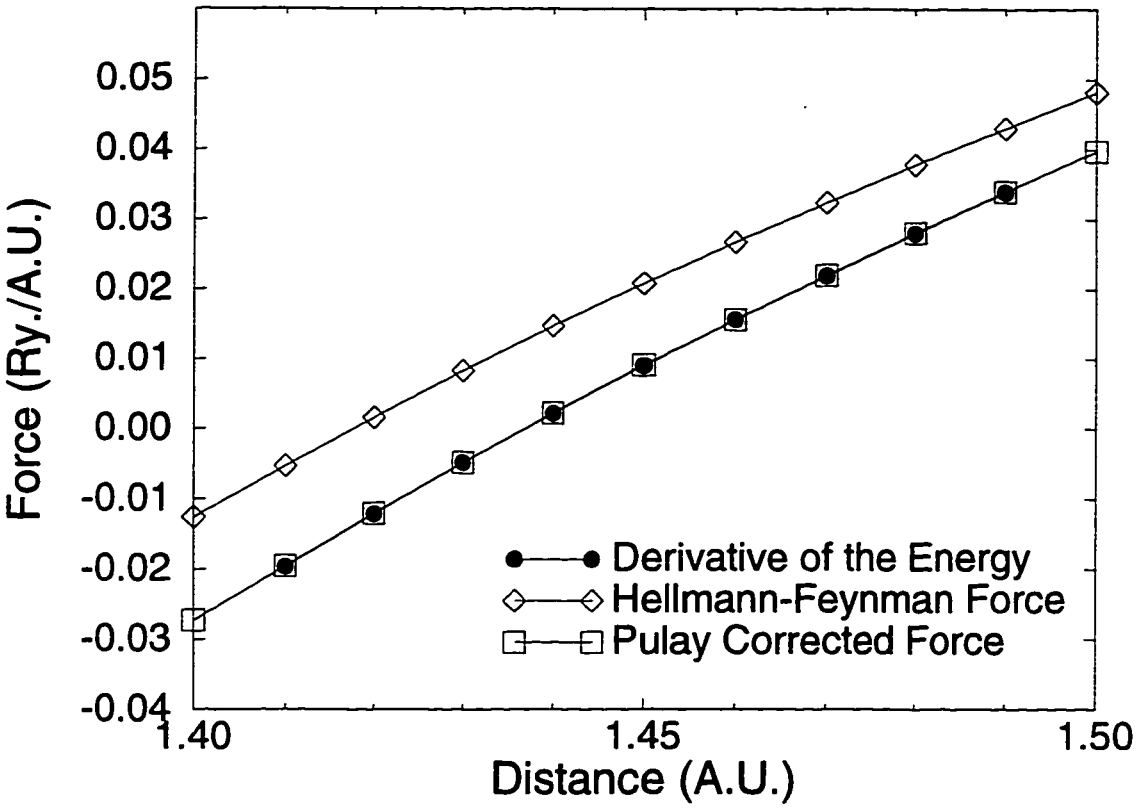

Figure 3.12: The forces from an all-electron computation for H_2 with and without Pulay corrections. The derivative of the energy is included for comparison. The box size was 24 × 12 × 12 a.u. and a 64 × 32 × 32 grid was used. The backdrop parameters were $\bar{x}_1 = 4$ a.u., $\bar{x}_2 = \bar{x}_3 = 2$ a.u., and $1/\bar{J}_i = 4$ for $i = 1 - 3$. For each atom, the local adaptation parameters are $1/|J|_\nu = 64$ and $\kappa_\nu = 0.5$ a.u. A Gaussian delta with $\sigma = 0.6$ was used.

this correction is remarkably small. Practical computations must be well converged with respect to the number of grid points, and we find that the Hellmann-Feynman forces are typically in such good agreement with the derivatives of the total energy that the extra expense of evaluating the Pulay corrections can not be justified during a relaxation of the atomic coordinates. On the other hand, it is useful to evaluate the Pulay corrections once the energy minimum has been found in order to make sure that the Pulay corrections would not change the answer. Furthermore, maintaining energy conservation during an extended molecular dynamics run would require including the Pulay corrections.

3.3.6 Band structure calculations

Bloch's Theorem requires the eigenstates of a periodic potential to have the form $\psi(\vec{x}) = u(\vec{x}) \exp(i\vec{k} \cdot \vec{x})$, where $u(\vec{x})$ has the periodicity of the potential. Most standard electronic structure algorithms solve for $u(\vec{x})$, which is an eigenfunction of a modified Hamiltonian with extra $\vec{k}$ dependent terms in the kinetic energy. An alternative approach is to solve directly for $\psi(\vec{x})$, which is an eigenfunction of the standard Hamiltonian with phase shifts at the boundaries. The phase shifts correspond to multiplication by a complex number whenever information is transported across the boundary in one direction, and multiplication by the complex conjugate when information crosses the boundary in the opposite direction. Picking the proper phase shifts allows a computation to be done for an arbitrary point in the Brillouin zone. This makes it possible to sample the Brillouin zone properly during calculations for solids. It also allows band structure calculations for solids. Complex phase shifts require complex wavefunctions. The Hamiltonian remains Hermitian, but some complexity is inevitably added to the algorithm. With careful coding to make sure that operations are not wasted, the required time is doubled. An extra complication arises from the nonlocal pseudopotentials. Since the pseudopotentials are short ranged, it is only necessary to consider one image of each

atom in the fundamental unit cell. However, extra phases must be included when multiplying by the nonlocal projectors in order to make sure that the phase corresponds to multiplication by the wavefunction in the unit cell that is closest to the atom. Fig. 3.13 shows that our real-space calculations give very good agreement with a plane wave calculation for the band structure of bulk Si.

3.3.7 The algorithms

The sparsity of the equations allows effective use of iterative algorithms to solve the Poisson and Schrödinger equations. In typical iterative algorithms, the operator only appears through its action on a vector. There are three main advantages of iterative algorithms:

(a) The sparse Hamiltonian can be stored in the most compact possible form. A typical grid with 128 points in each direction produces a Laplacian with 4 trillion elements, only 150 million of which are non-zero. It is not feasible to store such a matrix in dense form, so without the use of iterative algorithms, we would be limited to very small grids.

(b) The use of iterative algorithms makes it possible to compute only those eigenvectors that are actually needed. In a typical electronic structure calculation, it is only necessary to find the small fraction of the eigenstates that are actually occupied. Standard eigenvalue algorithms that find all of the eigenvalues and eigenvectors of a matrix are extremely wasteful when applied to such problems. Using an iterative algorithm that finds only selected eigenstates saves time in addition to memory.

(c) Standard iterative algorithms progressively improve an initial guess solution. Since self-consistency is achieved iteratively in Kohn-Sham electronic structure calculations, good initial guess solutions are usually available. Furthermore, it is quite common to do a series of calculations for structures that vary only slightly from each other. In this case, the solution for the previous system typically provides a good initial guess for

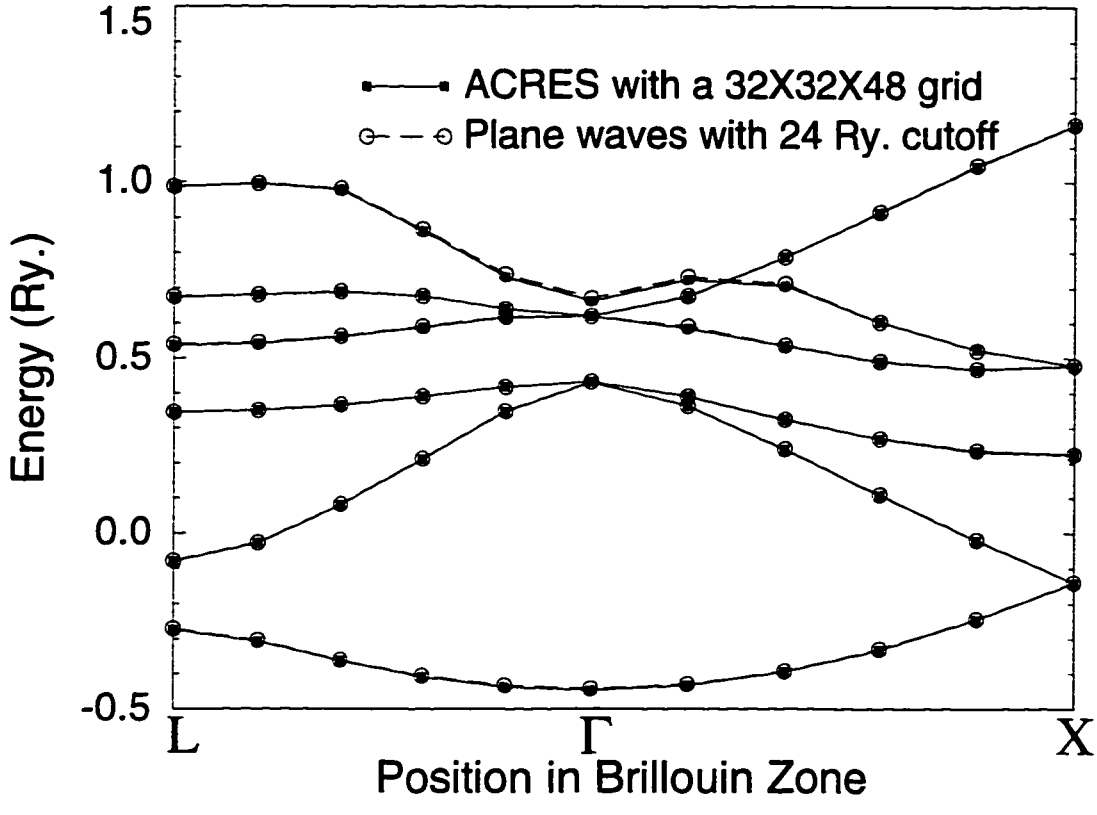

Figure 3.13: The bands structure of diamond structure Si unfolded from the results for a 4 atom unit cell. 30 inequivalent k-points were used to sample the Brillouin zone. Once self-consistency was achieved, calculations with a fixed density were done for the Brillouin zone locations indicated in the figure. A 32 × 32 × 48 grid was used without a backdrop. For each atom, the local adaptation parameters were $1/|J|_\nu = 8$ and $\kappa_\nu = 0.9$ a.u.

each new system. Iterative algorithms save a great deal of time by taking advantage of this knowledge.

Now, we will discuss the specific iterative algorithms that we have used: Lanczos, inverse iteration, and conjugate gradient.

For computing the eigenstates, we experimented extensively with a Lanczos algorithm. We concluded that we could not get acceptable performance from it. For Lanczos algorithms, the fastest states to converge are at both ends of the spectrum. The unwanted states need to be filtered out. Thus, the convergence time for Lanczos algorithms is proportional to the total width of the spectrum [53]. The width of the spectrum is dominated by the largest eigenvalue of the Hamiltonian, and is proportional to the inverse square of the minimum distance between points in the real space grid. This distance decreases with increasing adaptation or grid size. Therefore, Lanczos algorithms become increasingly slow as the adaptation or the number of grid points is increased. Even worse, with increasing adaptation or grid size, the upper states become increasingly wild and the algorithm can loose stability.

As a result of our experience with Lanczos, we developed a modified inverse iteration eigenvalue solver. The basic idea of inverse iteration is to apply $(H-\lambda)^{-1}$ to an initial guess vector repeatedly, where λ is an approximation to an eigenvalue of the Hamiltonian H. The result of applying $(H-\lambda)^{-1}$ to a vector $|\Psi_i\rangle$ is calculated by solving the linear system

$$(H-\lambda)|\Psi_{i+1}\rangle = |\Psi_i\rangle \tag{3.29}$$

for the unknown result $|\Psi_{i+1}\rangle$. We solve this system of equations iteratively by means of a conjugate gradient algorithm. In a standard inverse iteration algorithm, λ is driven to the eigenvalue by applying the shifts $\langle\Psi_i|\Psi_{i+1}\rangle^{-1}$. This approach drives the system Eq. (3.29) to singular behavior and leads to problems involved with solving ill-conditioned equations. Instead, we choose λ so that it is close to the desired eigenvalue, but not so close that the linear system becomes ill-conditioned. Noniterative diagonal-

ization within small subspaces consisting of states with energies near λ allows nearly degenerate states to be efficiently resolved. Further modifications allow us to find the spectrum reliably with very little *a priori* knowledge: a lower bound on the spectrum and a typical spacing for the eigenvalues. Alternatively, we can improve efficiently upon a previous solution. The inverse iteration algorithm only requires orthogonalization within degenerate (or nearly degenerate) groups of eigenstates, and thus the computational time scales as $N \times n_e$ with N the total number of points in the 3-dimensional grid and n_e the number of electrons in the system. Furthermore, since the inverse iteration procedure strongly suppresses states with energies far from λ, the rate of convergence does not depend on the total width of the spectrum, and the slowing down that kills Lanczos is avoided.

A conjugate gradient linear solver is used to solve the Poisson equation and forms the kernel of our inverse iteration algorithm. We use a multigrid based preconditioner in order to avoid a slowing of the algorithm with increasing grid size due to the time required to converge long wavelength modes. Further preconditioning based on subspace ideas improves the performance of the algorithm for nearly singular systems. Details of the algorithms are given in Chapter 4.

These iterative methods make it possible to employ rather large grids, and consequently allow us to investigate complex or difficult systems.

3.4 Results

Using the approach described in this chapter, we have implemented DFT/LDA [54] and DFT/GGA [55] electronic structure calculations on the Naval Research Laboratory CM-5 massively parallel supercomputer. Within this approach, all-electron computations involving atoms with $1s$, $2s$, and $2p$ valence electrons are feasible. We have also implemented the pseudopotential approach, using the norm-conserving nonlocal pseudopotentials of Bachelet et al. [56], and the Kleinman-Bylander procedure to render the

nonlocal components separable [57].

For the all-electron calculations, the adaptation of the grid is determined by the requirement that the density of grid points near the atomic cores is sufficient to represent accurately the $1/r$ divergence of the Coulomb potential. Fig. 3.14 shows the occupied wave functions of the O_2 molecule along a line through the centers of the two atoms. Notice the very large difference between the spacing of grid points in the vacuum region and near the atomic nuclei. The large number of grid points close to the nuclei allows accurate representation of the cusps and nodes of the wavefunctions.

For a more quantitative comparison to other theoretical results and to experiment, Table 3.1 and Table 3.2 show our calculated results for a number of test systems.

Our results are in good agreement both with experimental values and with other theoretical work using similar methods.

We find a relatively slow deviation of physical properties from converged values as the number of grid points is reduced. For example, Fig. 3.15 shows the fractional errors in the bond length, vibrational frequency, and total energy obtained from an all-electron calculation for a H_2 molecule using a number of different grid sizes. Even with the smallest grid, which has only 1/64 of the points in the largest grid, the errors in these physical quantities are only about 1%.

In order to compare the efficiency of ACRES to a more traditional method, we computed the total energy of a H atom using several sizes of unadapted, regular grids. The resulting fractional errors in the total energy and the computational times are compared to corresponding ACRES results in Table 3.3. The relevant comparison is between results of comparable accuracy: the ACRES computation with a $32 \times 32 \times 32$ point grid achieved slightly better accuracy (at a cost of 18 seconds) than the regular grid computation with $128 \times 128 \times 128$ points, which cost 300 seconds. Therefore, for an equivalent level of accuracy, ACRES greatly reduces computational time and memory requirements relative to a regular grid method. In fact, it would be extremely difficult

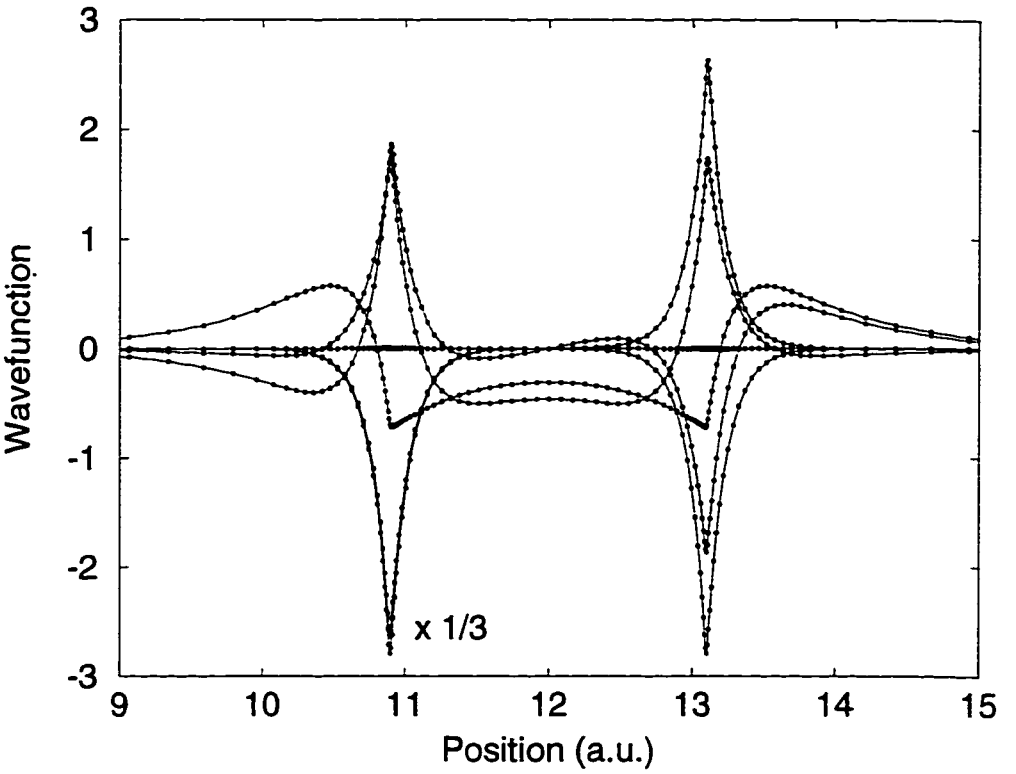

Figure 3.14: Occupied wave functions of the O_2 molecule, along a line through the centers of the atoms. The π bonding and anti-bonding wave functions collapse onto the horizontal axis (they have nodes through the atomic centers). The $1s$ bonding and anti-bonding states were scaled by a factor of 1/3 so they could be displayed on the same scale. Points on the curves indicate values at actual grid points used in the calculation.

Table 3.1: Comparison of our all-electron results with other theory and experiment. References are [a] Ref. 58, S-VWN; [b] Ref. 59; [c] Ref. 60, Perdew-Wang; [d] Ref. 58, B-LYP; [e] Ref. 60, PW GGA-II; [f] Ref. 61, Perdew-Zunger; [g] Ref. 62, $2p^4$ ^{1}D state; [h] Ref. 61, PW91; a_0 is binding distance, ω is vibrational frequency, E_0 is minimum energy, E_{at} is total energy.

	ACRES	**Other Theory**	**Experiment**
H_2 [LDA] a_0 (a.u.)	1.448	1.446[a]	1.401[b]
ω (cm^{-1})	4192	4207[a]	4401[b]
E_0 (Ry)	-2.276	−2.27[c]	−2.349[b]
H_2 [GGA] a_0 (a.u.)	1.416	1.413[d]	1.401[b]
ω (cm^{-1})	4381	4373[e]	4401[b]
E_0 (Ry)	-2.340	−2.34[e]	−2.349[b]
O [LDA] E_{at} (Ry)	-148.870	−148.938[f]	−150.027[g]
O [GGA] E_{at} (Ry)	-149.912	−149.994[h]	−150.027[g]
O_2 [LDA] a_0 (a.u.)	2.32	2.30[a]	2.28[b]
ω (cm^{-1})	1661	1642[a]	1580[b]
O_2 [GGA] a_0 (a.u.)	2.34	2.34[d]	2.28[b]
ω (cm^{-1})	1557	1518[d]	1580[b]

Table 3.2: Comparison of our pseudopotential results with other theory and experiment. References are [a] Ref. 58, S-VWN; [b] Ref. 59; [c] Ref. 63, VWN; [d] Ref. 63, Harmonic fitting to experiment; [e] Ref. 61, Perdew-Zunger. a_0 is binding distance, ω is vibrational frequency, θ is the bond angle, and B is the bulk modulus. The Si calculations are for the diamond structure.

	ACRES	**Other Theory**	**Experiment**
H_2 [LDA] a_0 (a.u.)	1.441	1.446[a]	1.401[b]
ω (cm^{-1})	4212	4207[a]	4401[b]
N_2 [LDA] a_0 (a.u.)	2.067	2.07[c]	2.07[b]
ω (cm^{-1})	2375	2401[a]	2377[d]
O_2 [LDA] a_0 (a.u.)	2.281	2.30[a]	2.28[b]
ω (cm^{-1})	1588	1642[a]	1580[b]
H_2O [LDA] a_0 (a.u.)	1.840	1.844[a]	1.812[a]
θ (deg)	104.1	103.6[a]	103.9[a]
Si [LDA] a_0 (a.u.)	10.14	10.16[e]	10.26
B (GPa)	97.34	96.57[e]	98.80

Table 3.3: Convergence with respect to grid size of the total energy of a H atom in a 12 × 12 × 12 a.u. box. The errors are computed relative to the largest (most accurate) ACRES calculation.

	Regular Grid		**ACRES**	
grid size	error	time (s)	error	time (s)
32^3	9%	11	0.9%	18
64^3	3%	32	0.1%	59
128^3	1%	300	–	442

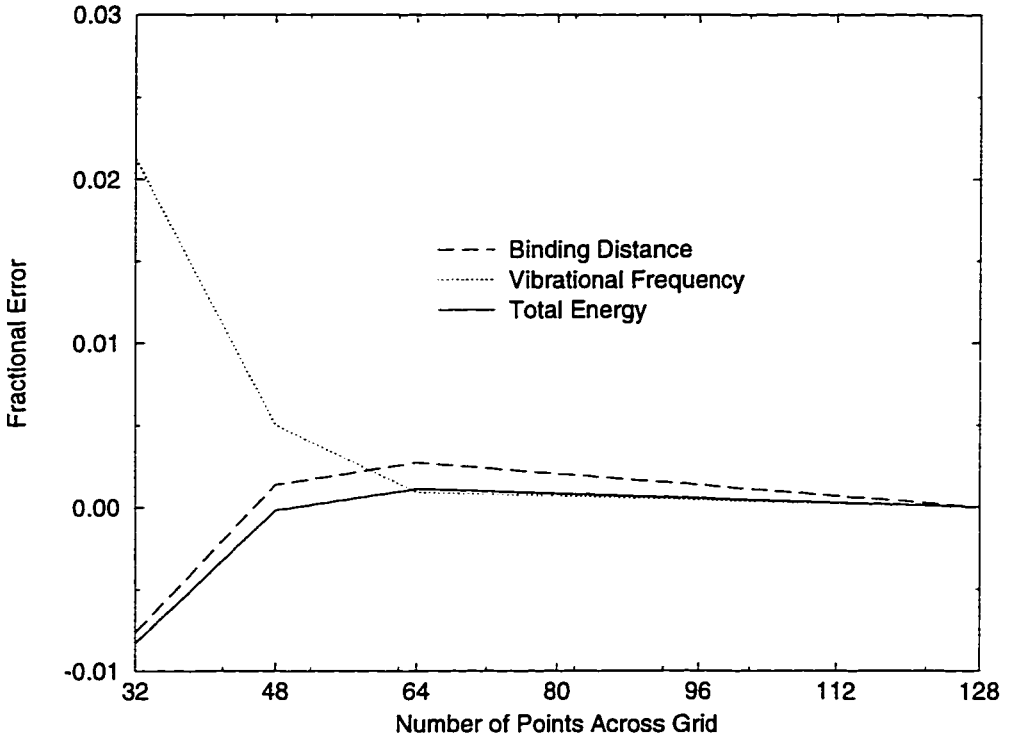

Figure 3.15: The convergence with increasing grid size of physical properties of a H_2 molecule. A 12 x 12 x 24 a.u. box was used. The grids have the indicated number of points in their short directions and twice as many in their long direction. The results from the 128 x 128 x 256 grid were taken to be exact.

to do accurate computations for the systems discussed in Table 3.1 without adaptation of the grid.

3.5 Conclusion

We have developed the ACRES approach and implemented it on a CM-5 massively parallel supercomputer. We have fully functional all-electron and pseudopotential codes that use LDA and GGA exchange correlation functionals. We can compute accurate forces, band structures, and structural properties for standard test systems such as H, O, N, H_2, O_2, N_2, H_2O, and bulk Si. We believe that ACRES provides a promising approach to large scale electronic structure calculations involving nonuniform length scales, especially on massively parallel computers.

3.6 Acknowledgments

This work was supported by the Office of Naval Research grant N00014-93-1-0190. We are grateful to F. Gygi for sharing his thoughts on adaptive methods. The authors would also like to thank Umesh Waghmare for improving the ACRES calculation of bulk properties of solids and for providing his improved results for silicon.

Chapter 4

Inverse iteration eigensolver with applications to efficient Kohn-Sham electronic structure computations

4.1 Introduction

Among large computational tasks, finding a few eigenvalues and eigenvectors of a very large matrix is perhaps one of the most common in many scientific applications. Quite often, the eigensolver used to perform this difficult numerical task is the most expensive part of a computation. Moreover, when the eigensolver lies within one or more outer loops of the code (which can be the case in many applications), the computational cost is augmented and the performance of the eigensolver becomes a critical issue. In physics, examples of problems where such a computational task arises are found in quantum mechanics (diagonalization of a Hamiltonian matrix), in statistical mechanics (diagonalization of a transfer matrix), etc. Outer loops arise in self-consistent approaches

(which are used to handle nonlinear terms), parameter updates (such as the relaxation of atomic coordinates in dynamical simulations), etc.

The difficulty of finding a few eigenvalues and eigenvectors of a large sparse matrix imposes important limitations on *ab-initio* calculations of the electronic and structural properties of molecules and solids, based on the Kohn-Sham density functional approach [2]. The term *ab-initio* refers to the fact that the only input in these calculations is the atomic numbers of the constituent atoms. This approach reduces the problem of finding the total-energy of a system of ions and interacting electrons to one of finding the wavefunctions of a set of fictitious particles whose density is identical to the density of real electrons. The wavefunctions are determined by solving a set of single-particle Schrödinger equations in a mean-field approximation, where the effects of exchange and correlation between electrons are taken into account by a functional of the local density (the so called local-density approximation). The use of a basis or a real-space representation of the wavefunctions, produces a matrix whose eigenvectors and eigenvalues are solutions of the single-particle Schrödinger equations. Since the density enters in the determination of the single-particle equations, and is given by the sum over absolute squares of occupied wavefunctions, the system of equations has to be solved self-consistently beginning with a guess for the wavefunctions.

Typically, the number of basis elements, n_b, needed to represent the wavefunctions, and the number of occupied wavefunctions, n_e, that must be calculated, grow linearly with the size of the system N. Since finding an eigenvector with length n_b requires at least $O(n_b)$ operations, the minimum possible scaling of a Kohn-Sham density functional computation is $O(n_e n_b)$, or $O(N^2)$. Electronic structure methods with an $O(N)$ scaling have been developed recently [64, 65, 66, 67, 68, 69, 70], but such methods require additional approximations which either restrict the wavefunctions to a subset of the basis elements or avoid explicit calculation of each occupied wavefunction. If one wishes to avoid additional approximations, the goal becomes to get as close as possible to

$O(n_e n_b)$ or $O(N^2)$ scaling.

Traditional matrix diagonalization techniques designed to find all eigenvalues and eigenvectors of dense matrices have an $O(n_b^3)$ scaling. The prefactor in these methods is small, but as the system size increases they eventually lose to methods with a more efficient scaling. In density functional computations, a real space representation of the wavefunctions produces a sparse Hamiltonian H. All terms in H are local in real space, except the Laplacian, which in discrete form involves only the values at nearby points, and thus it is represented as a sparse matrix. Since the number of nearby points that appear in the discretized representation of the Laplacian is independent of the system size, multiplying a vector v of length n_b by the Hamiltonian requires only $O(n_b)$ operations. Therefore, an eigensolver algorithm that needs only an implicit representation of the matrix H through its action on an arbitrary v, potentially offers the ability to solve for an eigenvector of H in only $O(n_b)$ operations. A variety of iterative algorithms that use this implicit representation have been developed. Therefore, combining a real-space representation of the Hamiltonian with such an iterative eigensolver avoids the $O(n_b^3)$ scaling of full diagonalization and offers a large savings in time and memory. However, most iterative eigensolvers either explicitly or implicitly require orthogonalization of the eigenvectors. As the size of systems grows, the cost of keeping the wavefunctions orthogonal, which scales like $O(n_e^2 n_b)$, becomes an increasingly important part of the computational cost.

Traditional Lanczos algorithms use a interesting approach to avoid orthogonalization of the eigenvectors, but their convergence rate is proportional to the eigenvalue spacing divided by the total width of the spectrum [53]. As the size of the matrix is increased, this ratio typically decreases. Therefore, standard Lanczos methods become increasingly slow as the size of the matrix is increased. In our work on an adaptive coordinate, real-space electronic structure (ACRES) code [71, 72], we encountered difficulties with Lanczos related to this fundamental limitation. The relatively large number of basis

functions used in a real-space representation leads to high energy states in the spectrum, a very wide spectrum, and consequently a very slow convergence of the algorithm. We found that this problem is particularly troublesome for adaptive real-space methods where the energy of the highest eigenstates can be huge. In fact, for strongly adapted systems, the highest eigenvalues become so large that the Lanczos algorithm can even loose stability and fail to converge.

To overcome the problems of the Lanczos method, we have developed an efficient algorithm based on inverse iteration to find a few eigenstates of an implicitly known (possibly large) matrix while avoiding global orthogonalization of the eigenvectors and dependence of the convergence rate on the width of the spectrum. In the rest of this chapter, we first introduce the standard inverse iteration algorithm (Section 4.2), we then describe our main modifications of the algorithm (Section 4.3), we discuss implementation details (Section 4.4), we elaborate on further modifications that improve performance (Section 4.5), and finally we give convergence results for a test system (Section 4.6).

4.2 Inverse Iteration

We begin with a brief review of the basic idea behind inverse iteration. Assume that we want the first few eigenstates of a symmetric matrix (extension to a Hermitian matrix is straightforward). Consider a matrix H with eigenvalues ϵ_i and eigenvectors φ_i

$$H\varphi_i = \epsilon_i\varphi_i \tag{4.1}$$

The algorithm turns a linear solver into an eigensolver by iteratively generating the sequences $v^{(n)}$ and $\lambda^{(n)}$ defined by the three step recursion

$$\left(H - \lambda^{(n)}\right)\tilde{v}^{(n+1)} = v^{(n)} \tag{4.2}$$

$$\lambda^{(n+1)} = \lambda^{(n)} + \frac{1}{\langle v^{(n)}|\tilde{v}^{(n+1)}\rangle} \tag{4.3}$$

$$v^{(n+1)} = \frac{\tilde{v}^{(n+1)}}{\sqrt{\langle \tilde{v}^{(n+1)} | \tilde{v}^{(n+1)} \rangle}}. \tag{4.4}$$

At each step, $\tilde{v}$ is determined by solving the linear system Eq. 4.2 using a linear solver algorithm. Then, the update of λ is calculated from Eq. 4.3, and finally v is obtained by normalizing $\tilde{v}$ with the standard normalization given by Eq. 4.4.

To understand how the algorithm works, expand $v^{(n)}$ in the basis composed of the eigenvectors of H

$$v^{(n)} = \sum \alpha_i^{(n)} \varphi_i. \tag{4.5}$$

After one iteration of the linear equation, we obtain

$$v^{(n+1)} = \sum \frac{\alpha_i^{(n)}}{\epsilon_i - \lambda^{(n)}} \varphi_i. \tag{4.6}$$

If $\lambda^{(n)}$ is very close to the eigenvalue ϵ_i, the corresponding eigenvector φ_i is very strongly amplified in $\tilde{v}^{(n+1)}$ because of the small denominator. Likewise,

$$\lambda^{(n+1)} = \lambda^{(n)} + \left(\sum \frac{|\alpha_i^{(n)}|^2}{\epsilon_i - \lambda^{(n)}} \right)^{-1}. \tag{4.7}$$

If $\lambda^{(n)}$ is close to the eigenvalue ϵ_i, the sum is dominated by the ith term giving

$$\lambda^{(n+1)} \approx \lambda^{(n)} + |\alpha_i^{(n)}|^{-2} (\epsilon_i - \lambda^{(n)}). \tag{4.8}$$

Then, if $\alpha_i^{(n)}$ is not so small that $\lambda^{(n+1)}$ overshoots ϵ_i, $\lambda^{(n+1)}$ is driven towards ϵ_i. Therefore, if $v^{(0)}$ is not far from the eigenvector corresponding to the eigenvalue nearest to $\lambda^{(0)}$, the sequences $\lambda^{(n)}$ and $v^{(n)}$ will converge to the nearest eigenstate in a few iterations.

The potential for avoiding global orthogonalization of the eigenvectors is apparent in the basic inverse iteration algorithm. Since inverse iteration converges to the eigenvalue closest to the initial guess $\lambda^{(0)}$, inverse iteration procedures started from widely different values of $\lambda^{(0)}$ should converge to different eigenvalues. Eigenstates with different eigenvalues are automatically orthogonal, thus orthogonalization between non-degenerate states is in principle unnecessary. In practice, unless the original guesses for

the eigenvalues are extremely accurate, it is necessary to orthogonalize a given state to states with nearby eigenvalues in order to avoid repeatedly converging to the same state. However, applying this local orthogonalization to find n_e orthogonal states requires only $O(n_e n_o n_b)$ operations where n_o is the size of the local orthogonalization block. Since n_o is typically of order 1, this offers a substantial savings over the $O(n_e^2 n_b)$ scaling of full orthogonalization.

The potential for avoiding the dependence of the convergence rate on the width of the spectrum is also apparent in the basic inverse iteration algorithm. Since the quantity $(\epsilon_i - \lambda^{(n)})^{-1}$ appears in Eq. 4.6 and Eq. 4.7, states that are outside the energy range near λ are very strongly suppressed. The very high energy states that cause problems for Lanczos methods are exactly the states that are most strongly suppressed.

4.3 Modified Inverse Iteration

There are two aspects of basic inverse iteration that have traditionally made it unsuitable for many applications: the spectrum must be known *a priori* and the linear system of Eq. 4.2 must be solved efficiently. We address both issues in the following.

4.3.1 Scanning for the spectrum

For inverse iteration to converge reliably to the nearest eigenvalue, the original guess $\lambda^{(0)}$ must be close to an eigenvalue. Accordingly, inverse iteration has usually been employed when either the spectrum was known exactly or very good estimates of the eigenvalues were available. When dealing with real physical problems that produce sparse matrices, sufficiently good estimates of the eigenvalues may or may not be available. In fact, both cases may occur while solving a given physical system. For example, in standard implementations of Kohn-Sham density functional theory, the eigensolver lies within a self-consistency loop that forces the electronic density to be consistent with the electronic wavefunctions. When solving a new structure, the spectrum is initially

completely unknown. Then, for the first few self-consistency iterations, the electronic density (and therefore the Hamiltonian) may change so much that there are many eigenvalue crossings, and the spectrum becomes completely scrambled at each step. Finally, as self-consistency is achieved, the density changes diminish until there are no longer any level crossings, and the spectrum from the previous step is a good estimate for the new spectrum. In order to handle efficiently all of these situations, our routine can be used in two modes: a mode with good guesses for the eigenvalues (referred to as "initialized mode") and a "scanning mode".

In the first mode, each eigenstate is found by using inverse iteration in the traditional way: $\lambda^{(0)}$ is initialized to each approximate eigenvalue, $v^{(0)}$ is initialized to each approximate eigenvector (if available), and the recursion described by Eq. 4.2, Eq. 4.3, and Eq. 4.4 is carried out for each eigenvalue until the eigenvector converges. Before the first iteration and during the normalization step of each iteration, v is orthogonalized to states with nearby eigenvalues in order to prevent accidental convergence to the same eigenvector twice.

When either the spectrum is unknown, or the uncertainties in the eigenvalues are comparable to the spacing between them, traditional inverse iteration will not find reliably all of the desired eigenstates. Standard routines for finding the eigenvalues involve transformations that ruin the sparsity of the matrix unless it has a very special form, which rarely arises in physical problems. For large matrices that are very sparse, such transformation techniques can be prohibitively expensive in terms of time and memory. Therefore, it is not practical to use a standard algorithm to find the eigenvalues, followed by inverse iteration to find the eigenvectors. In order to handle this case, we have developed the scanning mode — an algorithm that reliably obtains the eigenvalues and eigenvectors with little *a priori* knowledge of the spectrum. The information needed in the scanning mode is a lower bound on the spectrum λ_0 and an estimate of the typical spacing between non-degenerate eigenvalues. Assuming that this much is known,

λ is initialized to the lower bound on the spectrum. Then, the recursion described by Eq. 4.2, Eq. 4.3, and Eq. 4.4 is carried out. When v converges to an eigenvector, the recursion is started over with the same λ but with a new vector. The vectors used to initialize the procedure can be educated guesses (which improves convergence) or simply random. Reasonable educated guesses can be constructed in electronic structure calculations by superposition of atomic wavefunctions. In order to ensure that the algorithm always converges to a new eigenvector, v is orthogonalized to the previously found eigenvectors with eigenvalues close to λ before the first step of the recursion and during the normalization step of each iteration.

Stability of this approach is problematic. If the initial vector $v^{(0)}$ is not close enough to the next eigenvector, the algorithm may converge to a higher eigenvalue. This happens because the estimated change in λ given by Eq. 4.3 may be inaccurate when $v^{(n)}$ or $v^{(n+1)}$ is far from an eigenvector. As a result, λ can jump over an eigenvalue causing it to be missed. This problem can be avoided by limiting the maximum change in λ to a quantity g that we call the granularity. If the jump from $\lambda^{(n)}$ to $\lambda^{(n+1)}$ is larger than the granularity, we take $\lambda^{(n+1)} = \lambda^{(n)} + g$. This makes the algorithm safer, but somewhat slower when far from convergence. Taking g sufficiently small ensures that all eigenvalues will be found in succession. In practice, we find that the estimated shift in λ becomes quite accurate after the first iteration with λ closer to the new eigenvalue than the other unknown eigenvalues. Since λ is closer to the next highest eigenvalue than it is to any of the higher eigenvalues, it is sufficient to take g to be an approximation of the typical distance between non-degenerate eigenvalues (hence the term 'granularity' for g). We also apply this limiting of the λ shift in the first mode (when good guesses for the eigenvalues are available) in order to avoid poor results when the corresponding eigenvector guesses are poor.

For most types of matrices, the average spacing between eigenvalues depends on the location in the the spectrum. For example, in the case of an all-electron electronic

structure calculation, the energy level spacing decreases dramatically as the energy increases. Therefore, it may be beneficial to adopt an energy dependent granularity $g(\lambda)$. In our electronic structure calculations, we found that good results are obtained by taking

$$g(\lambda) = g_0(\lambda_\infty - \lambda)^{3/2} \tag{4.9}$$

with λ_∞ an upper bound on the desired eigenvalues. The exact form of $g(\lambda)$ must be determined for any new type of problem, but we have found that the stability and speed of the algorithm are not very dependent on it. Instead, good performance is obtained as long as $g(\lambda)$ captures the general trends in the eigenvalue spacing.

Furthermore, since the accuracy of the estimated shift in lambda given by Eq. 4.3 depends on how close $v^{(n)}$ and $v^{(n+1)}$ are to an eigenvector, the convergence can be improved by taking the granularity to be a function of the fluctuation σ

$$\sigma^{(n)} = \sqrt{\langle v^{(n)}|H^2|v^{(n)}\rangle - \langle v^{(n)}|H|v^{(n)}\rangle^2} \tag{4.10}$$

Since the fluctuation measures how far a vector is from a true eigenvector, the idea is to take the granularity magnitude inversely proportional to magnitude of the fluctuations. Thus, an increased granularity can prevent wasting many iterations stepping λ across a large gap when v already consists mostly of the next eigenvector. Likewise, a reduced granularity improves the stability when starting with a random initialization for the eigenvector. We found that good results are obtained with the expression

$$g(\lambda, \sigma^{(n)}, \sigma^{(n+1)}) = \frac{g(\lambda)}{\sqrt{\sigma^{(n)} + \sigma^{(n+1)}}}. \tag{4.11}$$

A difficult problem for any iterative eigensolver is to to obtain *all* the lowest eigenstates. In the case of the Lanczos algorithm, the accuracy must be set low enough to avoid spurious convergence, but no other control is possible. Since we want to obtain *all* eigenvalues, there is a trade off between speed and security. A very small value of the granularity ensures that no eigenvalues will be skipped, but it also slows down the

algorithm considerably. A larger value of the granularity is faster, but might result in a missed eigenvalue. An efficient compromise is to pick a value of the granularity that will probably, but not necessarily, find all of the eigenvalues and then check for missed eigenvalues using an additional scan. For this purpose, our inverse iteration routine features a third mode that we call "check mode". This mode works much like scan mode except that λ is always started from the lower bound on the spectrum. A random initial v is used, and it is orthogonalized to all of the vectors that have been previously found. Since the remaining spectrum is almost empty in the region of the scan, it is very difficult to miss an eigenvalue again. In case a missed eigenvalue is found, the check scan is repeated one more time to make sure that there are no additional missed eigenvalues. Since λ must scan all the way up from the lower bound, it is more efficient to find a vector during the initial scan than to pick it up in check mode. In practice, the best average performance has been found to result from picking a value of the granularity which allows the scan to miss an eigenvalue every few diagonalizations. The check mode is also useful in combination with the initialized mode, in order to check for a level crossing into the part of the spectrum under investigation, from the range outside. With reasonably accurate values of the granularity and a correct lower bound on the spectrum, check mode has never been observed to fail to find a missed eigenvalue.

4.3.2 Solving the linear system

Inverse iteration is usually employed with matrices with special forms, such as tridiagonal, that allow an analytic solution of Eq. 4.2. We have found that sparse systems can be handled by solving Eq. 4.2 with a conjugate gradient iterative solver [73]. With care to avoid a singular linear system and preconditioning of the conjugate gradient, an efficient eigensolver can be constructed.

The basic inverse iteration algorithm drives Eq. 4.2 to singular behavior and leads to slow convergence of the linear solver. This drawback is easily overcome: In order to

get good enhancement of the desired eigenvector, it is only necessary that $\lambda^{(n)}$ be close enough to ϵ_i so that φ_i is amplified much more than other states; it is not necessary to drive the linear system singular. We call the distance to the nearest eigenvalue the offset $\gamma^{(n)} = |\lambda^{(n)} - \epsilon_i|$. If we wish to resolve the eigenvectors efficiently using inverse iteration, λ must be chosen so that $\gamma << |\epsilon_{i+1} - \epsilon_i|$. The convergence time for conjugate gradient (and virtually all other iterative eigensolvers) grows sharply as the matrix $(H - \lambda)$ becomes singular. Therefore, controlling λ in such a way that γ does not become too small significantly reduces the convergence time of the algorithm. Since ϵ_i is not actually known, the quantity that is controlled is $\tilde{\gamma}^{(n)} = |\lambda^{(n)} - \langle v^{(n)}|H|v^{(n)}\rangle|$. If $\tilde{\gamma}^{(n)}$ is less than a quantity μ called the minimum offset, we take $\lambda^{(n)} = \langle v^{(n)}|H|v^{(n)}\rangle - \mu$. In initialized mode, let $\tilde{\epsilon}_i$ be the initial estimate for ϵ_i. Then, when solving for ϵ_i and the corresponding eigenvector the minimum offset μ is initialized to a constant fraction (0.05 works well for our systems) of $\tilde{\epsilon}_{i+1} - \tilde{\epsilon}_i$. In scanning mode, μ is initialized to a small constant (we use 0.05 again). Then, every time it is necessary to apply the minimum offset to λ, μ is divided by a constant (we use $\sqrt{10}$). This controlled reduction of μ ensures that the enhancement of the nearest eigenvector will eventually become large enough that the algorithm converges. The reduction factor is chosen in such a way that the algorithm converges before Eq. 4.2 becomes strongly singular.

The use of an offset to avoid a singular system is complicated somewhat by degenerate eigenvalues. In general, the spectrum is clustered into degenerate multiplets. If the initial estimates are accurate, applying the above procedure to a degenerate eigenvalue would result in $\mu = 0$ allowing the system to become very singular and the convergence to become very poor. This problem can be avoided by noting that in order to converge to the desired final accuracy δ, eigenvalues that differ by less than δ do not have to be resolved. Therefore, we can consider two estimated eigenvalues to be degenerate if $\tilde{\epsilon}_{i+1} - \tilde{\epsilon}_i < \delta$. Since we do not attempt to resolve degenerate eigenvalues, we can cluster together groups of degenerate eigenvalues. The criterion for efficient

separation of the eigenvalues in the cluster from the rest of the eigenvalues becomes $\gamma << \inf(\text{cluster}_{i+1}) - \sup(\text{cluster}_i)$. Therefore, we can initialize μ to a small fraction of $\inf(\text{cluster}_{i+1}) - \sup(\text{cluster}_i)$, which puts λ in the gap between consecutive clusters. The systematic reduction of the value of μ ensures that eigenvalues that are clustered but not truly degenerate will be quickly resolved well enough to converge.

4.4 Implementation Details

There are a number of issues that arose during the implementation of our inverse iteration algorithm. These include a choice of a convergence criterion for the eigenvectors, a choice of the convergence criterion for the linear solver, and a choice of a criterion for switching between the initialized mode and the scan mode.

A natural convergence criterion on a eigenvector is to compute the fluctuation defined earlier [Eq.(4.10)]. Let δ be the tolerance. The vector $v^{(n+1)}$ is considered to be sufficiently converged to an eigenvector when $\sigma^{(n+1)} < \delta$. Note that this criterion has the dimension of H and that other criteria are possible, for example $\sigma^{(n+1)}/\langle v^{(n+1)}|H|v^{(n+1)}\rangle$, which is dimensionless but clearly has problems for eigenvalues very close to 0. The best choice of a convergence criterion is problem dependent. Yet, $\sigma^{(n+1)} < \delta$ is not sufficient for a rather subtle reason. For a vector v decomposed as above using the basis of the eigenvectors of H, the fluctuations are

$$\sigma^2 = 1/2 \sum_{i,j} \alpha_i^2 \alpha_j^2 (\epsilon_i - \epsilon_j)^2, \tag{4.12}$$

the problem being that because of the $(\epsilon_i - \epsilon_j)^2$ factors, σ is not very sensitive to mixing between states that are close together but nondegenerate. In particular, this problem shows up in the failure of higher states to converge. Since the fluctuations are computed after orthogonalization to previously found eigenvectors, a previous eigenvector that has not been accurately resolved contributes a term to the fluctuation that can not be reduced by a more accurate solution for the new eigenvector. The cumulative effect

of a couple such terms can cause the new eigenvector to never reach the convergence criterion. In order to avoid such problems, we introduced a second convergence criterion based on

$$\left[\bar{\sigma}^{(n)}\right]^2 = \frac{\langle v^{(n)}|(H-\lambda)^{-2}|v^{(n)}\rangle - \langle v^{(n)}|(H-\lambda)^{-1}|v^{(n)}\rangle^2}{\langle v^{(n)}|(H-\lambda)^{-2}|v^{(n)}\rangle} = 1 - \frac{\langle v^{(n)}|\tilde{v}^{(n+1)}\rangle}{\langle \tilde{v}^{(n+1)}|\tilde{v}^{(n+1)}\rangle} \quad (4.13)$$

This criterion solves the problem because $\bar{\sigma}^{(n)}$ is much more sensitive to mixing between eigenvectors close to λ than $\sigma^{(n+1)}$. The computation of $\bar{\sigma}^{(n)}$ is almost free because the necessary quantities are already available, but it corresponds to the fluctuations of $(H-\lambda)^{-1}$ evaluated at the *previous* iteration. This does not seem to slow the convergence of the algorithm. Finally, $\bar{\sigma}^{(n)}$ is weighted with a function of λ in order to give a convergence criterion comparable to the one on $\sigma^{(n+1)}$.

It is pointless to solve the linear system very accurately if accurate eigenvectors are not needed. Therefore, in order to get the best possible performance, it is necessary to relate the convergence criterion for the linear solver to the tolerance for the eigenvectors. The convergence criterion for the solver is based on the norm of the residual of Eq. 4.2

$$|r| = \| \left(H - \lambda^{(n)}\right) \tilde{v}^{(n+1)} - v^{(n)} \|. \quad (4.14)$$

The solution $\tilde{v}^{(n+1)}$ must be found accurately enough that $|r| < \bar{\delta}$, where $\bar{\delta}$ is the convergence criterion for the solver. We have been unable to find a theoretical relationship between δ and $\bar{\delta}$. Therefore, we used an empirical relation of $\bar{\delta}$ as a function of δ, which ensures that a more accurate solution for $\tilde{v}^{(n+1)}$ would not affect the performance of the inverse iteration routine.

In order to take advantage of the initialized mode, a criterion is needed for deciding when the initial approximation to the spectrum is accurate enough. When the initialization data comes from a previous diagonalization of a similar matrix, the maximum difference between the eigenvalues of the old matrix and the expectation values of the new Hamiltonian in the old eigenvectors provides a useful measure of how well the old results approximate the desired new results. An even better determination is provided

by diagonalizing the new matrix within the subspace of the old eigenvectors, and comparing the subspace eigenvalues to the old eigenvalues. When the old results do provide an adequate initialization for the new diagonalization, the eigenvalues and eigenvectors of this subspace diagonalization also provide much better initialization guesses than the old eigenvalues and eigenvectors.

4.5 Optimized Inverse Iteration

4.5.1 Multigrid preconditioned conjugate gradient

When applied to sparse systems like those generated by discretizing a differential operator on a real space grid, iterative linear solvers generally suffer from a convergence slowdown due to long wavelength modes. An alternative way to view this problem is to consider it as arising from the communication time to transfer information from one side of the grid to the other in a consistent solution. Since multiplication by a sparse matrix only transmits information to nearby points, the information must travel across the grid one step at a time. By this argument, the number of matrix-vector multiplies needed to solve a system should at best scale proportionally to the number of points across the grid. Typical grids used in real space electronic structure calculations have of order 100 points in each spatial direction. Therefore, this effect can cause a significant slowdown.

Multigrid methods make it possible to avoid this slowdown [74]. The basic idea is to solve a problem on several different grids at the same time. Since typical iterative linear solvers converge modes with wavelengths comparable to the grid spacing very quickly, using grids with many different grid spacings allow all modes to be converged quickly. The multigrid technique has been very successfully applied to solving discretized partial differential equations. More recently, it has been successfully applied to electronic structure calculations with euclidean metrics [41, 33].

Typically, the multigrid approach has been used to accelerate relaxation methods such as Jacobi relaxation or steepest descent. A standard multigrid cycle starts with a few relaxation (or "smoothing") steps on the finest grid in order to converge the highest frequency modes. Once the high frequency modes are converged, the residual of the linear system has no high frequency components. This residual is then represented on the next coarser grid using a restriction operator R. Since the residual did not have any high frequency modes, this representation is accurate. The relaxation and restriction operations are repeated on successively coarser grids until, at some coarse level, the problem is easily solved exactly. The process is ran in reverse with a prolongation operator P, used to represent each coarse grid result on the next finer grid. At each grid level, the result from the next coarser grid is added to the previous solution, and a few relaxation iterations are applied in order to incorporate the coarse and fine results together. When the finest level is reached, the convergence criterion is evaluated and, if not satisfied, the whole process is repeated. The trick is that since the residual on each grid level is smooth after a few relaxation iterations, it can be accurately represented on a coarser level.

When used by itself, the conjugate gradient algorithm is usually much more efficient than relaxation methods, but suffers from a slowdown due to long wavelength modes. One advantage of the conjugate gradient algorithm is that it can be preconditioned. This means that if one wants to solve $Ax = b$, there is a straightforward modification of the algorithm that allows the solution of $\tilde{A}^{-1}Ax = \tilde{A}^{-1}b$ instead, where $\tilde{A}$ is some operator that is much easier than A to invert. If $\tilde{A}$ is an approximation of A, $\tilde{A}^{-1}A$ is close to the identity, and the linear system is much easier to solve. A natural idea is to take $\tilde{A}$ to be the representation of A on a coarser grid i.e. $\tilde{A} = PAR$. The coarser grid problem is much smaller and, therefore, much easier to invert. Furthermore, $\tilde{A}^{-1}A$ should appear be the identity to the long wavelength modes that are causing the slowdown problem. A look at the preconditioned conjugate gradient algorithm reveals

that $\tilde{A}^{-1}$ is only applied to the residual, which should be smooth after a few iterations. This is additional indication that this approach may work effectively.

We have found that this approach does indeed work, and that it successfully eliminates the grid size dependence due to long wavelength modes (see Section 4.6). One unexpected difficulty resulted from the fact that $\tilde{A}$ has a very large null space since it is a projection into a smaller space and back. This null space contains only high frequency modes, but due to the workings of the preconditioned conjugate gradient they will never be eliminated from the residual if PAR itself is used as the preconditioner. One way to avoid this problem is to sandwich PAR between two Jacobi relaxation steps. This keeps a symmetric preconditioner and provides a pathway through which the null space of PAR can be eliminated from the residual. The cost of this procedure is two extra matrix-vector multiplies in order to perform the two additional Jacobi relaxations per iteration. Despite this extra cost, this preconditioner provides the best performance that we have found. Another difficulty is that the attractive potential felt by the electrons near the nuclei quickly loses meaning when calculated on coarse grids. Therefore, it does not make sense to include these terms in the coarse grid representation of the Hamiltonian. A low order regular grid representation of the Laplacian and the constant shift λ are adequate on the coarse grid.

4.5.2 Subspace diagonalization

Despite its strengths, inverse iteration is not a very efficient method for resolving nearly degenerate states. In order to resolve such states in a few iterations, λ must be driven much closer to one of the eigenvalues than the other. In general, this results in a singular Eq. 4.2 and poor performance of the linear solver. In contrast, standard dense matrix routines are unacceptably slow when applied to very large sparse matrices, but they are very good at resolving nearly degenerate states. The advantages of the two approaches can be combined by replacing the local block orthogonalization steps in the inverse

iteration algorithm described above with local subspace diagonalizations. Subspace diagonalization works as follows: Given an arbitrary set of n_o vectors v_i, construct the subspace Hamiltonian $\tilde{H}_{ij} = \langle v_i|H|v_j\rangle$ and the subspace overlap matrix $\tilde{S}_{ij} = \langle v_i|v_j\rangle$. Then, solve the generalized eigenvalue equation

$$\tilde{H}\phi_i = \tilde{\epsilon}_i\tilde{S}\phi_i. \tag{4.15}$$

Since this is a n_o dimensional equation where n_o is a small number, the time used to solve it with an $O(N^3)$ algorithm is negligible compared to the rest of the computations. Mixing the vectors v_i according to the components of ϕ_j gives a new vector w_j. The set of w vectors has the useful properties $\langle w_i|w_j\rangle = \delta_{ij}$ and $\langle w_i|H|w_j\rangle = \tilde{\epsilon}_i\delta_{ij}$. The w vectors are as close as one can possibly get to eigenvectors using a combination of the v vectors. If the v vectors are the result of inverse iteration, all of the eigenvectors that correspond to far away eigenvalues have been strongly suppressed. Therefore, if there are as many independent v vectors in a local block as there are eigenstates in that region of the spectrum, the w vectors will be very close to eigenvectors of the full space. Note that it was not necessary to resolve the vectors within the subspace with the inverse iteration part. Therefore, λ can be chosen as a convenient value in the proper region of the spectrum, and singular equations are avoided. Each algorithm does what it is best at — inverse iteration eliminates all the eigenvectors that have much different energies, while the $O(N^3)$ diagonalization algorithm takes care of possible degeneracies. In practice, the local blocks that are used for local subspace diagonalization are somewhat different from the blocks used for local orthogonalization. With subspace diagonalization, it is useful to include nearby (i.e. $\tilde{\epsilon} \approx \lambda$) initial guess states that have not yet been refined, as well as the previously converged states. Furthermore, at each inverse iteration, instead of eliminating the old $v^{(n)}$ from the space and replacing it with $v^{(n+1)}$, it is useful to increase the dimension of the space and diagonalize with both states. After diagonalization, the highest energy state can then be eliminated. This ensures that all of the vectors in the block will steadily converge toward the desired eigenvectors.

4.5.3 Subspace preconditioning

Despite the best efforts to avoid singular systems in Eq. 4.2, they cannot be avoided completely. Therefore, it is useful to precondition the subspace of states with energies as close to λ as possible. Subspace diagonalization provides an effective means of doing this. Within the subspace spanned by the w vectors, the inverse of $H - \lambda$ is exactly

$$(H - \lambda)_w^{-1} = \sum_i |w_i\rangle \frac{1}{\tilde{\epsilon}_i - \lambda} \langle w_i|. \tag{4.16}$$

During the preconditioning step of the conjugate gradient algorithm, this part of the space can be projected out and preconditioned in this fashion, while the remainder of the space is preconditioned with the multigrid procedure described above. When the w vectors come close to spanning the subspace of states with energies close to λ (which is usually the case in the initialized mode), this is a very effective preconditioner that greatly reduces the dependence of the linear solver performance on the singularity of the system. In fact, for typical situations with this preconditioner, λ must differ from an eigenvalue by less than 10^{-4} Ry before the solver becomes noticeably slower.

4.6 Results

In order to evaluate the performance our inverse iteration algorithm, we carried out a number of all electron Kohn-Sham density functional calculations of the ground state energy of the hydrogen atom in a $12 \times 12 \times 12$ a.u. box. We used cubic grids with 32, 64, and 128 points on each side, and we used both a real space regular grid method and our adaptive coordinate real-space electronic structure (ACRES) method. The ACRES grid was adapted so that the grid spacing close to the hydrogen atom was reduced by a factor of 16 compared to the regular grid spacing. For each of these cases, we carried out a computation using three different eigensolver methods: Lanczos with implicit restart using a 40 vector Krylov space; Modified Inverse Iteration, the algorithm described in this chapter in Sections 4.2 – 4.4 (i.e. without multigrid preconditioning, subspace

Table 4.1: Average number of matrix-vector multiplies per diagonalization and average time per diagonalization as a function of grid size. Timings are from a density functional calculation of the ground state energy of the hydrogen atom using a real space regular grid method or our adaptive coordinate real-space electronic structure (ACRES) method. Times are cpu seconds per processor on a 256 node Thinking Machines CM-5E, for the three algorithms discussed here, Lanczos, Modified Inverse Iteration (I.I.) and Optimized I.I. (see text).

Grid type		**Lanczos**		**Modified I.I.**		**Optimized I.I.**	
type	size	multiplies	time	multiplies	time	multiplies	time
Regular	32^3	104	2.00	56	1.34	7	0.74
Regular	64^3	198	7.68	105	6.35	7	1.38
Regular	128^3	539	97.10	213	59.92	9	6.50
Adapted	32^3	4860	78.93	154	2.59	14	0.82
Adapted	64^3	22941	826.77	320	14.78	13	2.34
Adapted	128^3	—	—	709	147.25	15	13.34

diagonalization, or subspace preconditioning); and Optimized Inverse Iteration, the full algorithm described in this chapter including the optimizations in Section 4.5. For each computation, we recorded the number of matrix-vector multiplies and the cpu time per processor used to find the ground state eigenvector during each self-consistency step. Table 4.1 shows the results of averaging these values over the self-consistency iterations.

The number of multiplies used by the Lanczos calculation clearly shows the disastrous dependence of the convergence rate on the width of the spectrum. The results were not bad for the regular grid, in which case the number of multiplies approximately doubles when the number of points across the grid is increased by a factor of 2. However,

with the adapted grid, the Lanczos calculations were so inefficient that we were not able to obtain any results for the $128 \times 128 \times 128$ grid. The factor of 16 change in spacing between the adapted and regular grids, produces as much as a factor of 256 increase in the width of the full spectrum. This slows down the calculation by a factor of 47 and 115 for the 32^3 and 64^3 grids respectively. While the performance degradation is not as bad as expected if the the Lanczos algorithm is assumed to scale with spectrum width, it is clear that this algorithm does not provide a useful eigensolver for computations with strongly adapted real space grids.

Even without multigrid preconditioning, subspace diagonalization, and subspace preconditioning, our inverse iteration algorithm performs slightly better than Lanczos for the regular grid and dramatically better for the adapted grid. The convergence does not show the dependence on the spectrum width inherent in Lanczos. Instead, the adapted grid calculations consistently take about 3 times as long as the regular grid calculations. This is not unreasonable given that the nonuniform metric of the adapted grid makes the eigenvalue problem somewhat more difficult to solve. The required number of multiplies does show a dependence on the size of the grid: doubling the number of points across the grid doubles the number of matrix-vector multiplies. This is consistent with the argument that multiplying by a sparse Hamiltonian only communicates information one step at a time, since information must be communicated all the way across the grid in order to obtain a consistent solution.

Adding multigrid preconditioning, subspace diagonalization, and subspace preconditioning substantially reduces the number of matrix-vector multiplies required for a diagonalization. It also eliminates the dependence on the grid size as would be expected for a properly functioning multigrid algorithm. The adapted grid calculations are still somewhat more costly than the regular grid calculations with the same number of points, but now the average change is reduced to less than a factor of 2. Once again this probably reflects difficulties due to the position dependent metric of the adapted

grid.

The computational time should increase by a factor of 8 over the increase in the number of multiplies when the number of points across the grid is doubled. This is due to the $O(n_b)$ scaling of the matrix-vector multiply. This increase is not observed. The reason is that even a $64 \times 64 \times 64$ grid is not very big for a 256 node (1024 vector unit) parallel computer. The increase in parallel efficiency offsets a significant part of the increase in computational difficulty.

4.7 Conclusions

We believe that inverse iteration offers a promising iterative method for finding selected eigenvalues and eigenvectors of large sparse matrices. The traditional limitations on inverse iteration can be overcome by suitable modifications of the algorithm and by using carefully preconditioned conjugate gradient as the linear solver. Inverse iteration avoids a couple of pitfalls that plague other iterative eigensolvers, namely global orthogonalization of eigenvectors and dependence of the convergence rate on the width of the spectrum. Modifications of the algorithm allow it to resolve nearly degenerate eigenvalues and make it possible to avoid inefficiencies due to long wavelength mode propagation. The modified inverse iteration is particularly well suited to Kohn-Sham electronic structure calculations using a real space basis, especially if adaptive grids are used. Clearly in this algorithm, there is plenty of empirical numerical analysis and optimizations. This description of the algorithm should be regarded as a framework to be adapted to a particular class of problems and as a list of useful tricks that we found to perform well. It can not be considered as a black box routine. Yet, we suggest that adaptation of this method to other types of problems could offer a substantial payoff.

Acknowledgement

This work was supported by the Office of Naval Research, grant N00014-93-1-0190. We would like to thank Emil Briggs for sharing his knowledge of multigrid methods, and Norm Troullier for several enlightening conversations about subspace diagonalization approaches.

Chapter 5

Applications

5.1 Introduction

Virtually all modern electronic devices are made by processing silicon wafers. Better understanding of the phenomena that take place at the surfaces of silicon wafers is required in order to achieve better control over device processing, which can allow smaller, faster, and cheaper devices. As electronic devices have become smaller and the desired degree of control has increased, experimental and theoretical efforts to understand the complex, nonequilibrium phenomena that take place during the processing of silicon surfaces have increasingly focused on microscopic aspects. First principles electronic structure calculations are a valuable tool in these investigations due to their unique ability to predict the physical properties of systems at the atomic scale accurately. Our ACRES method is especially well suited to the study of surfaces in general, because it can handle efficiently the large regions of vacuum that are required in order to isolate surfaces. When difficult atoms such as those with $1s$, $2p$, or $3d$ valence electrons are present, the inefficiency of treating the entire vacuum region with the high resolution needed to represent the difficult states becomes overwhelming. In this case, the ACRES method offers an even larger advantage over standard methods. One of the

most important processing steps in the fabrication of silicon based electronic devices is the addition of oxygen in order to create regions of insulating SiO_2. A particularly important and poorly understood part of this process is the initial absorption of the oxygen at the silicon surface. Oxygen is one of the most difficult atoms to treat with standard pseudopotential electronic structure methods because of the extremely high resolution required to represent its $2p$ electronic states. Since the simulation of the oxygen absorption process involves both a large region of vacuum and a very difficult atom, it is in a sense an ideal application for the ACRES method. In the rest of this chapter, we will give a brief review of the reconstruction of the clean silicon surface in Section 5.2, we will consider some general points relevant to the absorption of oxygen in Section 5.3, we will give the results of some ACRES calculations in Section 5.4, and we will discuss our conclusions and future plans in Section 5.5.

5.2 Reconstruction of the Silicon Surface

The absorption of oxygen at the silicon surface is complicated by the the fact that the clean silicon surface is reconstructed. This reconstruction creates many inequivalent sites at which the oxygen could initially bond to the substrate.

In 1959, Schlier and Farnsworth studied the microscopic structure of the Si (100) surface using low energy electron diffraction and obtained evidence of a (2×1) reconstruction [75]. They proposed that this structure resulted from the formation of dimers on the surface. A reduction from 2 dangling bonds per surface atom to 1 dangling bond per surface atom is the primary driving force behind the dimerization. Let the z axis be oriented perpendicular to the (100) surface under discussion, and take the x axis to correspond to the [110] direction. The unreconstructed surface atoms are originally in their bulk positions, which form a square lattice with a lattice constant of 7.26 a.u., the second nearest neighbor distance. During dimerization, the atoms move toward each other along the x axis to form a bond. The dimers lie side by side in rows running

along the y direction, which is perpendicular to the dimer bonds. The dimerization also induces atomic displacements in the layers near to the surface.

Further measurements using low energy electron diffraction, photoemission, and He diffraction revealed the presence of preparation dependent $p(2 \times 2)$ and $c(4 \times 2)$ reconstructions of the silicon surface. An empirical tight-binding calculation by Chadi [76] indicated that a tilted dimer had a slightly lower energy than a symmetric dimer, and it was observed that tilted dimers could explain the larger reconstructions. If the reflection symmetry through the center of the dimer bond is spontaneously broken, the x and z displacements of the two atoms that form a dimer are different, and a tilted dimer is produced. The dimer is said to be 'buckled'. If alternating dimers in the same chain are tilted in opposite directions, but nearest neighbor dimers between chains are tilted in the same direction a $p(2 \times 2)$ reconstruction in produced. If neighboring dimers both along and perpendicular to the chains are tilted in opposite directions, a $c(4 \times 2)$ reconstruction results.

Tromp, Hamers, and Demuth directly observed the dimers with scanning tunneling microscopy and confirmed that dimer rows were the basic reconstruction on the Si (100) surface [77, 78]. At room temperature, their measurements revealed the presence of both tilted and untilted dimers. They also observed that the tilt directions of nearby dimers were arranged to produce regions of $p(2 \times 2)$ and $c(4 \times 2)$ reconstruction in addition to the basic (2×1) reconstruction.

Wolkow carried out scanning tunneling microscope measurements at low temperature and observed that the number of buckled dimers increased as the temperature was decreased [79]. He interpreted this as an indication that the buckled dimers had a lower energy than the symmetric dimers. A number of first-principles calculations have supported this picture [80, 81, 82]. The energies of symmetric and buckled dimers are rather similar, but it is now generally accepted that accurate first principles calculations give a slightly lower energy for the buckled dimer. At room temperature, the dimers

are predicted to fluctuate between the two bistable buckled states, but at very low temperatures the fluctuations become frozen in and alternating buckled rows are formed in which each dimer is buckled in the opposite direction of its four neighbors. This gives an overall $c(4 \times 2)$ reconstruction of the surface.

5.3 Oxygen Absorption on the Silicon Surface

Very little is known about the way in which oxygen is absorbed at the silicon surface. Therefore, a good deal of care is required in order to make sure that computational methods are appropriate and do not bias the results. In order to obtain an accurate quantitative understanding of the microscopic processes during the initial stages of oxygen uptake on the (100) surface, it is essential to apply a very accurate theoretical model to the study of the system. Empirical potentials are not accurate for surfaces and do not tilt the dimers. Tight binding simulations have only been extensively tested for systems with a single type of atom. On the other hand, without any empirical information, density functional electronic structure computations determine most properties of silicon within 1% agreement with experimental results, and consequently such calculations are well suited for quantitative study of silicon based structures. Furthermore, our ACRES approach is particularly well suited to this problem due to its ability to handle efficiently the large difference between the length scale needed to represent the vacuum region and the length scale needed to represent the $2p$ orbitals of the oxygen atom. Doing an accurate calculation for an oxygen atom in a $7 \times 15 \times 41$ a.u. box using a plane wave code would require of order 500000 plane waves and would produce a Hamiltonian with roughly 2.5×10^{11} nonzero entries. In contrast, the ACRES method uses only 124416 grid points and obtains a very sparse Hamiltonian, while maintaining a faithful representation of the system.

The large phase space of possible absorption paths for oxygen on the silicon surface makes the investigation of this system challenging. A molecular dynamics simulation

would be too expensive. Likewise, a full determination of the energy surface as a function of all of the atomic coordinates would be prohibitive. Instead, we aim to determine the energies of various special configurations, namely some of the local minima and saddle points the energy surface. The global minimum of the energy is particularly important because it gives the equilibrium configuration of the structure at low temperature. Other local minima are occupied at finite temperatures and may indicate metastable states in which the system can become stuck for extended periods of time. The saddle points of the energy surface correspond to the transition states of the system, and their energies give the energy barriers for transformation between the various minima. Thus, they determine the dynamic properties of the system. Collective properties of the system at finite temperature could be found by feeding the results of such a first-principles calculation into a simulation of larger scale processes using, for example, the Monte Carlo technique [83, 84].

In order to determine the energies of local minima and saddle points of the energy surface, we start with a given configuration and then relax the atomic coordinates according to the forces in a steepest descent approach. The procedure is continued until the forces are sufficiently small. Depending on the initial structure, either the global or a local minimum may be found. If various symmetries are maintained during the procedure, some of the saddle points of the energy surface can also be found.

5.4 ACRES Results

5.4.1 Symmetric Dimer

First, we investigated the surface reconstruction without any oxygen present. Initially, we relaxed the dimer but enforced reflection symmetry in a plane perpendicular to the dimer bond. This results in a symmetric dimer reconstruction. Fig. 5.1 shows the self-consistent electronic charge density that we obtained from this calculation. For this

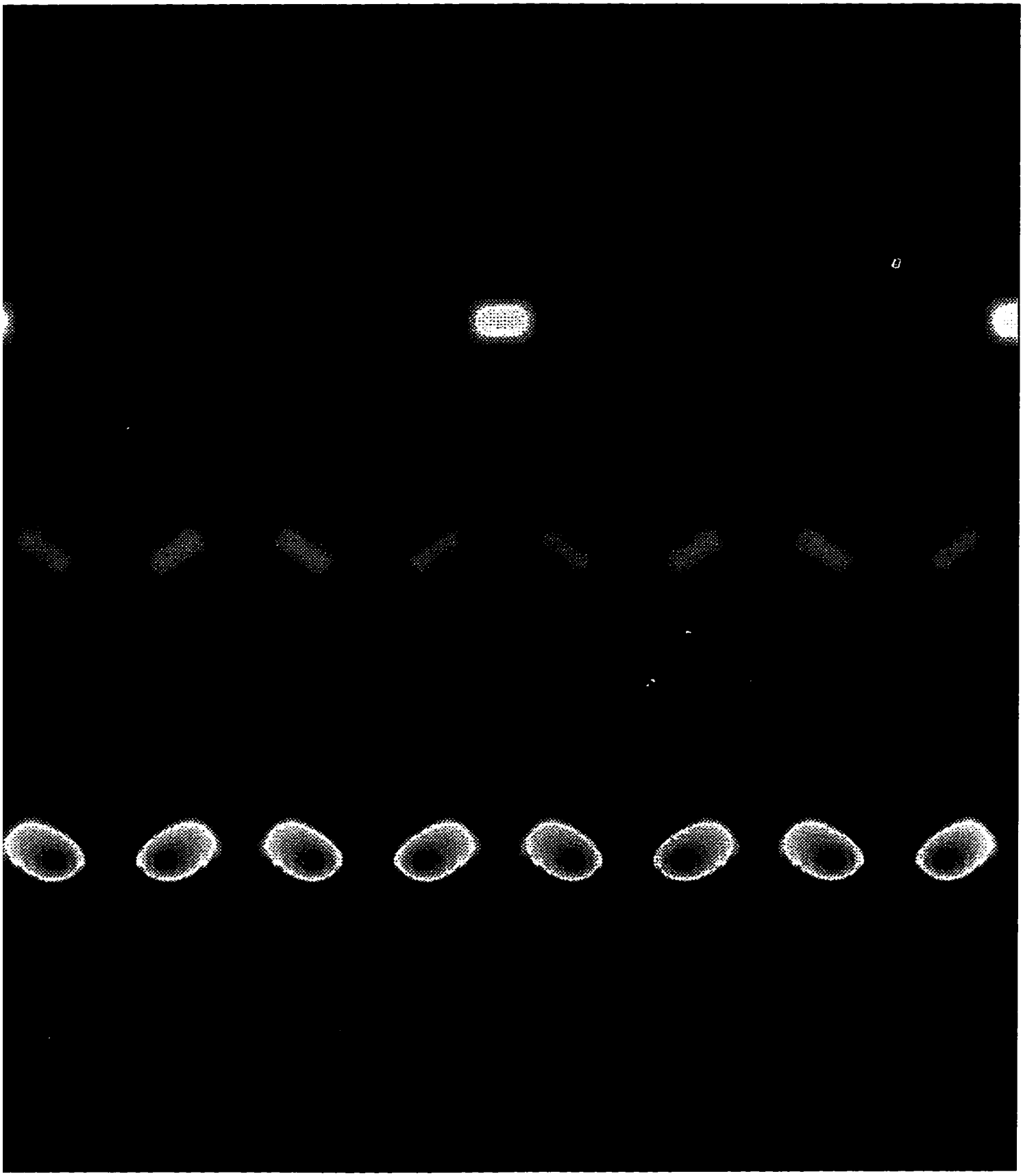

Figure 5.1: The charge density for a symmetric dimer reconstruction of the Si (100) surface shown in a cross-section through the center of the dimer bond. The size of the unit cell has been doubled in the direction parallel to the dimer in order to make it easier to see the dimer. 8 layers of Si atoms and 1 layer of surface terminating H atoms are included in the calculation, but only 4 layers of Si atoms and the H atoms can be seen in this cross-section. There are 2 additional rows of Si atoms located out of the plane of the picture between each of the sets of rows that are visible. The H atoms appear as bright spots because they have a much higher charge density than the Si atoms.

calculation, we held the bottom two layers of Si atoms fixed at their bulk locations, and we allowed the rest of the atoms to relax. In order to make the atoms far away from the surface act as much as possible like they were in the bulk solid without having to simulate a large region of bulk Si, we terminated the lower surface with hydrogen atoms. Since each hydrogen atom forms a single covalent bond, all of the Si dangling bonds are satisfied without creating any new dangling bonds.

With this reconstruction, the bands associated with the surface states cross the Fermi energy producing two partially filled bands. As is often the case in metallic systems, it is difficult to sample the Brillouin zone well enough to determine the Fermi surface accurately. For a reduced system with 5 layers of Si atom, we carried out a series of calculations with different sampling schemes with up to 15 inequivalent (32 total) k-points. For the purpose of relaxing the coordinates, we chose a sampling scheme with 6 inequivalent (8 total) k-points that was sufficient to converge the forces to better than 10 mRy/a.u. with respect to the number of k-points. Since the error in the relaxed energy is second order in the error in the atomic positions, this accuracy in the forces should be adequate for relaxation of the coordinates as long as only energy differences are needed. However, a much larger 15 inequivalent k-point calculation for the final atomic positions was needed in order to ensure that the energy itself was well converged with respect to the number of k-points.

5.4.2 Buckled Dimer

Next, we removed the constraint of reflection symmetry in a plane perpendicular to the dimer bond. The dimer tilts while relaxing the atomic positions without enforcing any symmetry between the two atoms of the dimer. This buckling of the dimer can be viewed as a Jahn-Teller distortion driven by the degeneracy at the open Fermi surface of the untilted dimer. By tilting the dimer, the Fermi surface is closed and the energy is lowered. Since we are not especially interested in low temperature properties, we

did not worry about the subtle interactions between the tilt directions of neighboring dimers. Instead, we assumed that every dimer tilts in the same direction. This gives us the smallest possible unit cell that can be used to investigate a dimerized surface. Fig. 5.2 shows the self-consistent electronic charge density that we obtained from this calculation. Since we determined from our calculations for the symmetric dimer that the relaxations were small after the first four layers from the surface, we have focused on the interesting region near the surface by including only 5 layers of Si atoms. We held the bottom layer of Si atoms fixed at their bulk locations, and we allowed the rest of the atoms to relax. We used 6 inequivalent k-points to sample the Brillouin zone, and we relaxed atoms until the forces were all less than 5 mRy/a.u. The dimer bond has a length of 4.26 a.u., which is substantially shorter than the bulk bond length of 4.44 a.u. Since the formation of the dimer involves considerable strain, this indicates that the bond has substantial double bond character. The backbond for the lower atom in the dimer is 4.31 a.u. long, indicating that it is somewhat stronger than a regular bond. The backbond on the higher atom in the dimer is 4.40 a.u. long, which is only slightly shorter than the bulk bond length. The corresponding lengths obtained by Yin and Cohen are 4.25 a.u., 4.35 a.u., and 4.42 a.u. respectively [80]. Thus, we obtain excellent agreement with other theoretical calculations.

For the different k-point sampling schemes that we tried, we found that the dimer buckled in all cases. Even though the buckled dimer state has a lower energy, the energy of the symmetric state is important because it represents the activation barrier for conversion from one buckled state to the other (i.e. the "flipping" of the dimers seen in experiment). The energy difference between the buckled and symmetric dimer varied between 26 meV for an 8 inequivalent (16 total) k-point sampling scheme and 297 meV for a 2 inequivalent (8 total) k-point scheme. This large range of values obtained from different k-point sampling schemes reflects a well known difficulty in performing density functional calculations for metallic systems, which is aggravated in this case by the

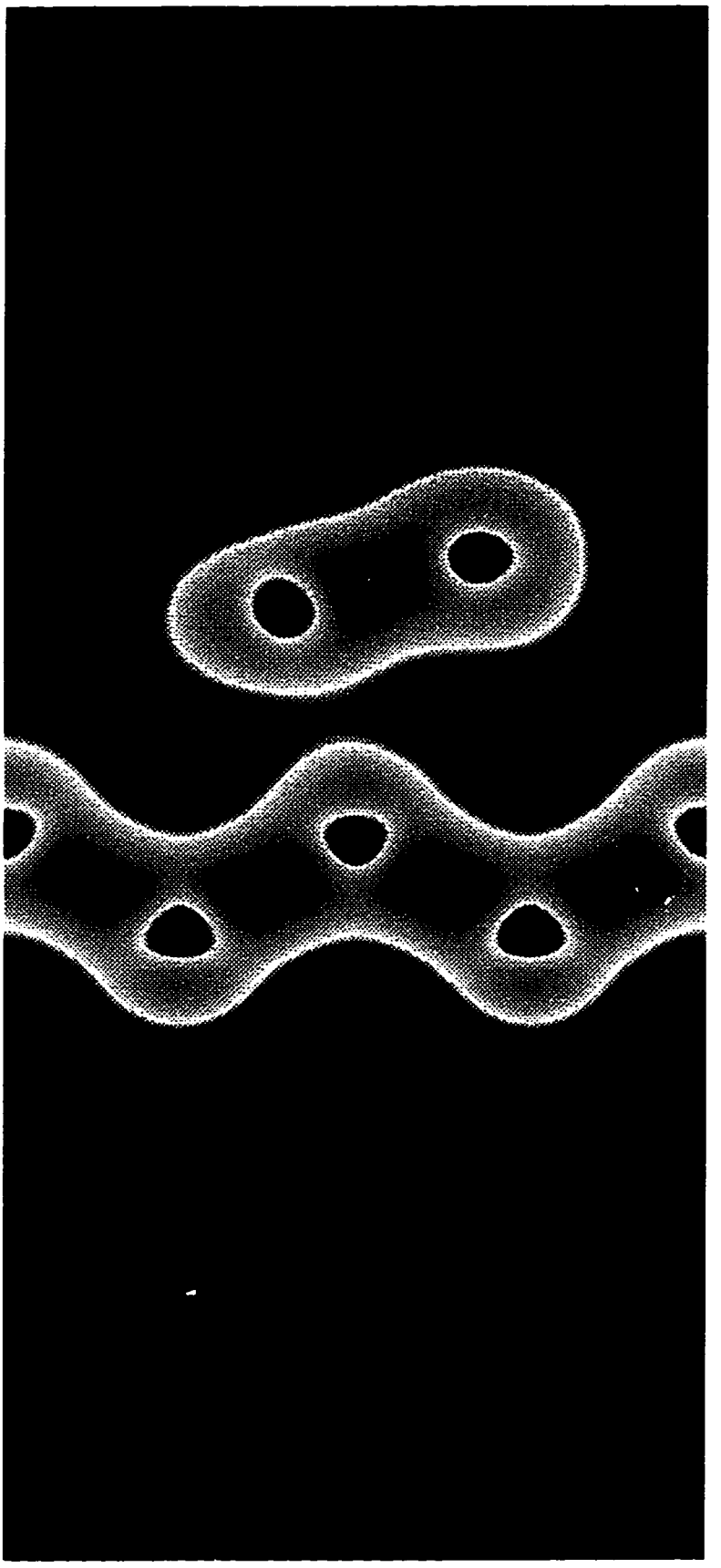

Figure 5.2: The charge density for a buckled dimer reconstruction of the Si (100) surface shown in a cross-section through the center of the dimer bond. We have concentrated on the interesting region near the surface by including only 5 layers of Si atoms and 1 layer of surface terminating H atoms in the calculation. Only 3 layers of Si atoms can be seen in this cross-section. There are 2 additional rows of Si atoms located out of the plane of the picture between the dimer and the top rows. The four H atoms that terminate the bonds on the bottom row of atoms are also located out of the plane with two atoms in front of the plane of the figure and and two behind.

small size of the energy difference between the two configurations. The most accurate calculation (15 inequivalent k-points) gave an energy difference of 140 meV. This is in good agreement with the 200 meV value obtained by Yin and Cohen [80] and the more recent result of 100 meV obtained by Dabrowski et al. [81].

5.4.3 Oxygenated Dimer

As a first attempt at including oxygen into the system, we put an oxygen atom into the middle of the symmetric dimer bond. This is likely to be a low energy location for the oxygen because it allows the dangling bonds to be as well satisfied as they are with the dimer, while releasing much of the stress due to the dimer formation. Fig. 5.3 shows the self-consistent electronic charge density that we obtained from this calculation. In order to show how the ACRES method applies to a complicated system like the one studied here, we include a cross-section of the grid that was used for this calculation in Fig. 5.4. Notice the mild enhancement of the grid density near the Si atoms, the larger enhancement for the H atoms, and the high density of points achieved at the O atom.

5.5 Discussion

The work described so far is an appealing demonstration of the capabilities of the ACRES method, but much remains to be done in this study of oxygenation of the Si(100) surface. Obvious additions include a better k-point sampling for the symmetric dimer and a calculation for a buckled dimer with oxygen included in the middle of the dimer bond. Numerous other sites where oxygen might be incorporated into the system also need to be examined in detail. The problem is quite difficult, with a very large configuration space that must be explored. We feel that these early results demonstrate that ACRES is the right tool to use for this difficult system, and we have great hopes for what this method will eventually achieve. Identifying the most probable sites for oxygen incorporation on Si surfaces, as well as pathways for diffusion between them, is

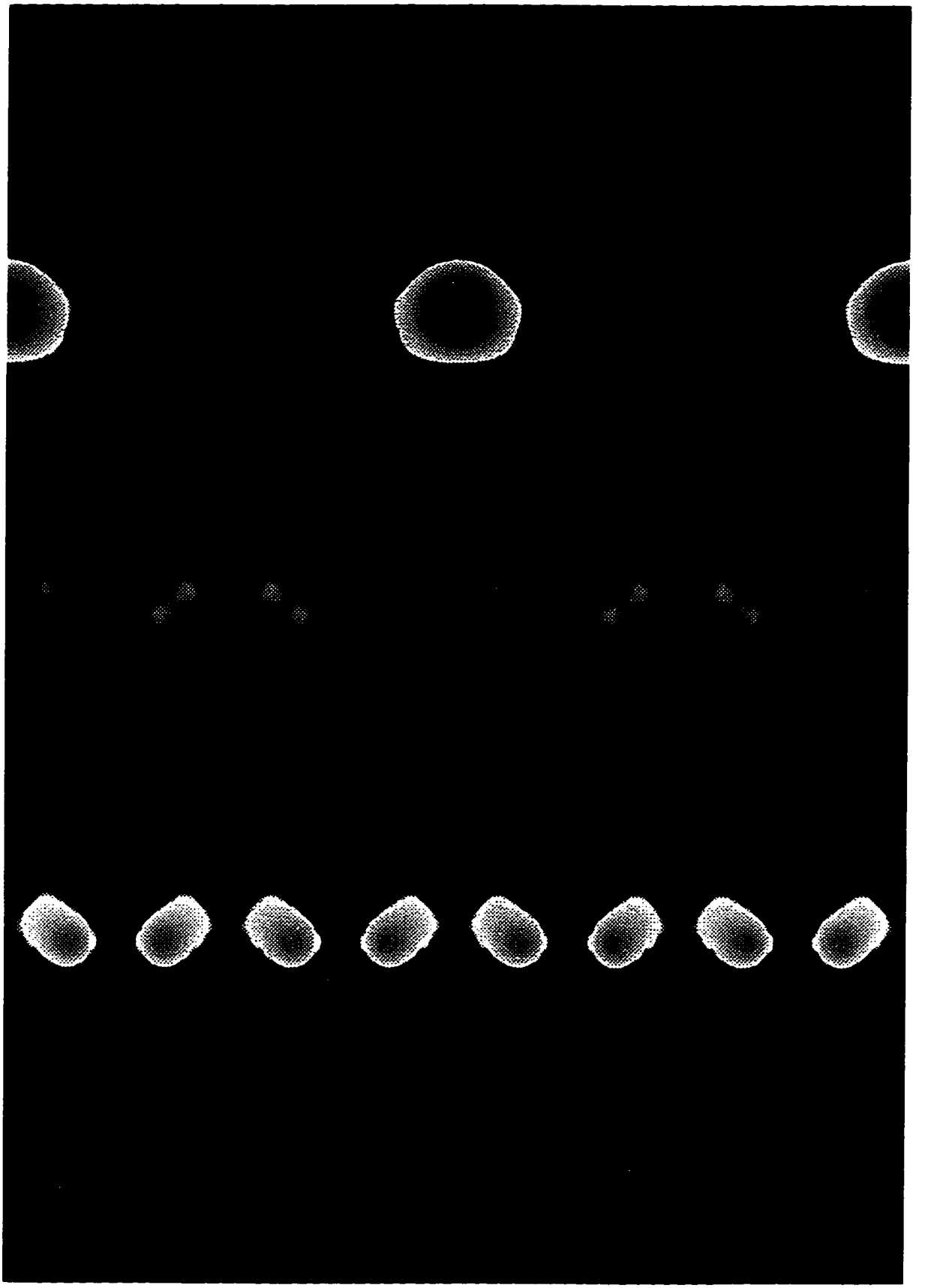

Figure 5.3: The charge density for a symmetric dimer reconstruction of the Si (100) surface with a half monolayer of oxygen inserted into the dimer bonds. The figure shows a cross-section through the center of the dimer bond. The size of the unit cell has been doubled in the direction parallel to the dimer in order to make it easier to see the dimer. The oxygen, 8 layers of Si atoms and 1 layer of surface terminating H atoms are included in the calculation. There are 2 rows of Si atoms located out of the plane of the picture between each of the sets of rows that are visible. The O atom appears as a very bright spot due to its very large charge density relative to other atoms, even the H atoms.

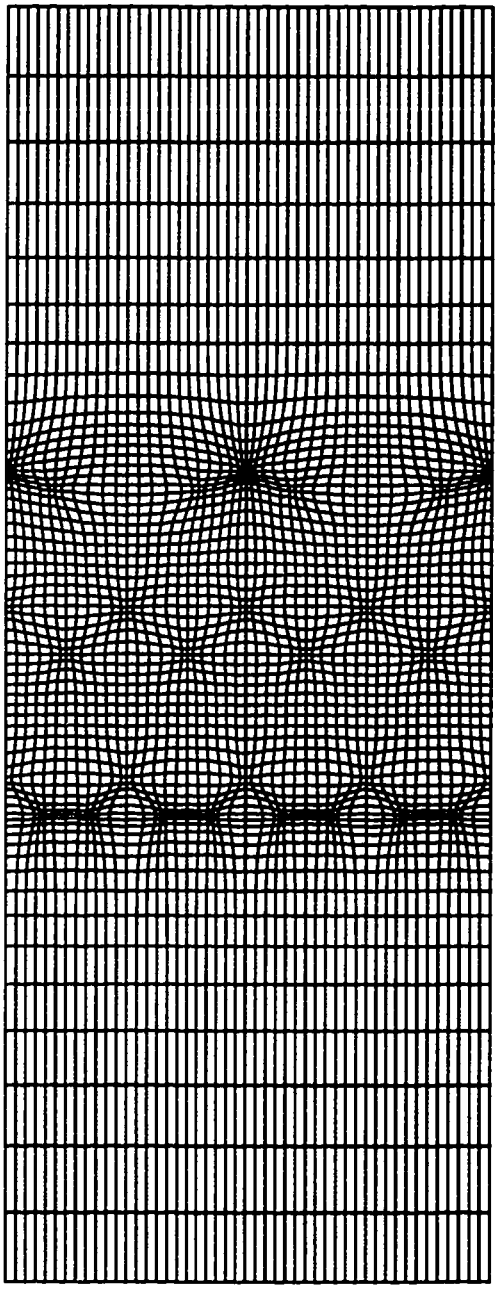

Figure 5.4: The grid used for the dimer with oxygen shown in cross-section through the center of the dimer bond. The size of the unit cell has been doubled in the direction parallel to the dimer in order to make it easier to see the dimer.

a task that we envisage completing in the near future.

Chapter 6

Conclusion and Prospects

We believe that the work discussed in this thesis demonstrates that an adaptive basis approach can substantially improve the efficiency and extend the applicability of quantum mechanical calculations. Although we obtained only moderate gains for the quantum spin systems discussed in Chapter 2 due to the extreme difficulty of solving these systems, the ACRES approach coupled with the algorithms of Chapter 4 promises to extend the usefulness of *ab initio* electronic structure computations significantly for inhomogeneous materials. The application of the ACRES method to the simulation of real physical systems that could not be studied practically using other methods is ongoing. The variational Hilbert space truncation approach would surely provide much more dramatic gains for systems whose basis states have a stronger hierarchy of importance. Our implementation of this approach waits only for a physically interesting problem of this type to come to our attention. The inverse iteration eigensolver discussed in Chapter 4 is very likely to provide excellent performance for a variety of problems that involve finding a few eigenvalues of a large sparse matrix, especially those with a wide spectrum. We look forward to applying it to other systems as well as continuing to optimize it for our ACRES calculations.

Although a substantial amount of effort was required in order to implement the

methods discussed in this thesis efficiently on parallel computers, we were eventually able to achieve good parallel efficiency, and we feel that the extra effort was justified by the extra power provided by massively parallel computation. The huge matrices associated with the larger spin clusters investigated in Chapter 2 could not have been diagonalized using conventional computers. Likewise, the power of parallel machines is helping our ACRES method to reach beyond systems that have been conventionally studied with *ab initio* techniques.

One remaining issue is that we have no way of guaranteeing that the ACRES method generates the best possible grid for determining the most important quantity in an electronic structure computation, the total energy. Picking a set of parameters that generates a nearly optimal grid for a complicated system still requires a considerable amount of experience and physical intuition. One exciting prospect that offers to remedy this situation is the combination of our variational truncation approach with an adaptive electronic structure method. The techniques discussed in Chapter 2 are immediately applicable to any local refinement scheme that maintains a variational principle, such as methods that use wavelets as a basis. The addition of basis states in order to refine the resolution in difficult regions and the removal of basis states in order to maintain a space of manageable size could be treated exactly as in our variational Hilbert space truncation approach. Unlike methods in which the resolution is enhanced to a certain extent within a predetermined radius of each atom, which can lead to incorrect treatment of atypical behavior in the bonding region, a variational approach would automatically achieve the optimal resolution everywhere in the system without any prejudice towards expected behavior. Furthermore, a variational approach would eliminate the tricky art of guessing the parameters that determine how the resolution varies in space.

As mentioned in Chapter 3, the problem with a local refinement electronic structure approach is that it is much more difficult to parallelize efficiently than the ACRES method. The difficulty of parallelizing a local refinement implementation efficiently is

closely related to the challenges encountered when parallelizing our variational Hilbert space truncation method. Although the techniques discussed in Chapter 2 achieve reasonable parallel performance, they do not approach the efficiency of the naturally parallel ACRES method. On the other hand, a variational basis truncation approach can not be directly combined with our ACRES method because the energy found using the finite difference approximation is not variational. One possible way to regain a variational energy is to replace the finite difference approximation with a finite element approach. By using the grid points determined by an ACRES type coordinate transformation as the nodes of a finite element mesh, the advantages of the ACRES method could be retained while, in principle, a variational energy could be maintained. Since the basis elements in such an approach would evolve continuously as the coordinate transformation was adjusted rather than being suddenly added or removed, the approach of Chapter 2 would have to be modified, but the principles should remain the same. Since such an approach would require the analytic evaluation of a large number of integrals involving the irregularly shaped finite element shape functions, it would likely require considerable coding effort. However, the finished product would almost surely be very powerful indeed.

Bibliography

[1] P. Hohenberg and W. Kohn, Phys. Rev. **136**, B864 (1964).

[2] W. Kohn and L. Sham, Phys. Rev. **140**, A1133 (1965).

[3] For a review of work on the HAFM, see E. Manousakis, Rev. Mod. Phys. **63**, 1 (1991).

[4] P. W. Anderson, Science **235**, 1196 (1987).

[5] F. C. Zhang and T. M. Rice, Phys. Rev. B **37**, 3759 (1988).

[6] D. W. Thompson, *On Growth and Form*, 2nd ed. (Cambridge University Press, Cambridge, 1942), Vol. II, Chap. IX, pp. 737–738.

[7] H. J. Schulz and T. A. L. Ziman, Europhys. Lett. **18**, 355 (1992).

[8] D. Sorensen, SIAM J. Matr. Anal. Apps. **13**, 357 (1992).

[9] N. Metropolis *et al.*, Journal of Chemical Physics **21**, 1087 (1953).

[10] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes*, 2nd ed. (Cambridge University Press, New York, 1992), Chap. 10.9.

[11] C. F. Bender and E. R. Davidson, Phys. Rev. **183**, 23 (1969).

[12] B. Huron, J. P. Malrieu, and P. Rancurel, J. Chem. Phys. **58**, 5745 (1973).

[13] S. Evangelisti, J.-P. Daudey, and J.-P. Malrieu, Chem. Phys. **75**, 91 (1983).

[14] D. Feller and E. R. Davidson, J. Chem. Phys. **90**, 1024 (1989).

[15] R. J. Harrison, J. Chem. Phys. **94**, 5021 (1991).

[16] D. Maynau and J.-L. Heully, Chem. Phys. Lett. **187**, 295 (1991).

[17] I. Shavitt, Chem. Phys. Lett. **192**, 135 (1992).

[18] W. Wenzel and K. G. Wilson, Phys. Rev. Lett. **69**, 800 (1992).

[19] M. M. Steiner, W. Wenzel, K. G. Wilson, and J. W. Wilkins, Chem. Phys. Lett. **231**, 263 (1994).

[20] R. J. Buenker and S. D. Peyerimhoff, Theoret. Chim. Acta **39**, 217 (1975).

[21] H. D. Raedt and W. von der Linden, Phys. Rev. B **45**, 8787 (1992).

[22] J. Riera and E. Dagotto, Phys. Rev. B **47**, 15346 (1993).

[23] M. R. Pederson and N. Laouini, Phys. Rev. B **48**, 2733 (1933).

[24] E. R. Gagliano, E. Dagotto, A. Moreo, and F. C. Alcaraz, Phys. Rev. B **34**, 1677 (1986).

[25] H. Q. Lin, Phys. Rev. B **42**, 6561 (1990).

[26] A. Pothen, H. D. Simon, and K.-P. Liou, SIAM J. Matrix Anal. Appl. **11**, 430 (1990).

[27] H. D. Simon, Comp. Sys. Eng. **2**, 135 (1991).

[28] Z. Johan, K. K. Mathur, S. L. Johnsson, and T. J. R. Hughes, Technical report, Thinking Machines (unpublished).

[29] D. Vanderbilt, Phys. Rev. B **32**, 8412 (1985).

[30] A. M. Rappe, K. M. Rabe, E. Kaxiras, and J. D. Joannopoulos, Phys. Rev. B **43**, 8861 (1991).

[31] N. Troullier and J. L. Martins, Phys. Rev. B **43**, 8861 (1991).

[32] J. R. Chelikowsky, N. Troullier, and Y. Saad, Phys. Rev. Lett. **72**, 1240 (1994); J. R. Chelikowsky, N. Troullier, K. Wu, and Y. Saad, Phys. Rev. B **50**, 11355 (1994).

[33] E. L. Briggs, D. J. Sullivan, and J. Bernholc, Phys. Rev. B **52**, R5471 (1995).

[34] S. Baroni and P. Giannozzi, Europhys. Lett. **17**, 547 (1992).

[35] K. A. Iyer, M. P. Merrick, and T. L. Beck, J. Chem. Phys. **103**, 227 (1995).

[36] T. Hoshi, M. Arai, and T. Fujiwara, Phys. Rev. B **52**, R5459 (1995).

[37] K. Cho, T. A. Arias, J. D. Joannopoulos, and P. K. Lam, Phys. Rev. Lett. **71**, 1808 (1993).

[38] S. Q. Wei and M. Y. Chou, preprint.

[39] S. R. White, J. W. Wilkins, and M. P. Teter, Phys. Rev. B **39**, 5819 (1989).

[40] E. Tsuchida and M. Tsukada, Sol. St. Comm. **94**, 5 (1995); Phys. Rev. B **52**, 5573 (1995).

[41] J. Bernholc, J.-Y. Yi, and D. J. Sullivan, Faraday Discuss. **92**, 217 (1991).

[42] E. J. Bylaska *et al.*, in *Proc. 6th SIAM Conf. Parallel Processing for Sci. Comput.* (SIAM, San Francisco, CA, 1995).

[43] W. H. Frey, Int. J. Numer. Methods Eng. **11**, 1653 (1977).

[44] C. D. Mobley and R. J. Stewart, J. Comput. Phys. **34**, 124 (1980).

[45] W. C. Thacker, Int. J. Numer. Methods Eng. **15**, 1335 (1980), and references therein.

[46] F. Gygi, Europhys. Lett. **19**, 617 (1992); Phys. Rev. B **48**, 11692 (1993); **51**, 11190 (1995); Private communication.

[47] D. R. Hamann, Phys. Rev. B **51**, 7337 (1995).

[48] A. Devenyi, K. Cho, T. A. Arias, and J. D. Joannopoulos, Phys. Rev. B **49**, 13373 (1994).

[49] F. Gygi and G. Galli, Phys. Rev. B **52**, R2229 (1995).

[50] R. W. Hockney and J. W. Eastwood, *Computer Simulations Using Particles* (Adam Hilger, Philadelphia, PA, 1988).

[51] We use the word *flat* to describe the trivial change of coordinates $\vec{x} = \vec{\xi}$, corresponding to a regular grid in $\vec{x}$ space. Strictly speaking, the Riemannian curvature $R^{\alpha}_{\beta\gamma\delta}$ is identically zero for any change of coordinates, i.e. our spaces are always flat.

[52] P. J. Davis and P. Rabinowitz, *Methods of Numerical Integration*, 2nd ed. (Academic Press, Orlando, FL, 1984).

[53] J. K. Cullum and R. A. Willoughby, *Lanczos Algorithms for Large Symmetric Eigenvalue Computations* (Birkhäuser, Boston, 1985), Vol. I, ,and references therein.

[54] J. P. Perdew and A. Zunger, Phys. Rev. B **23**, 5048 (1981).

[55] J. P. Perdew, in *Electronic Structure of Solids '91*, edited by P. Ziesche and H. Eschrig (Akademie Verlag, Berlin, 1991).

[56] G. B. Bachelet, D. R. Hamann, and M. Schlüter, Phys. Rev. B **26**, 4199 (1982).

[57] L. Kleinman and D. M. Bylander, Phys. Rev. Lett. **48**, 1425 (1982).

[58] B. G. Johnson, P. M. W. Gill, and J. A. Pople, J. Chem. Phys. **98**, 5612 (1993).

[59] K. P. Huber and G. Herzberg, *Molecular Spectra and Molecular Structure* (Van Nostrand Reinhold Company, New York, 1979), Vol. IV.

[60] J. P. Perdew *et al.*, Phys. Rev. B **46**, 6671 (1992).

[61] Y.-M. Juan and E. Kaxiras, Phys. Rev. B **48**, 14944 (1993); Private communication.

[62] C. E. Moore, *Atomic Energy Levels* (U. S. Government Printing Office, Washington, 1971), Vol. I.

[63] F. W. Kutzler and G. S. Painter, Phys. Rev. B **37**, 2850 (1988).

[64] G. Galli and M. Parrinello, Phys. Rev. Lett. **69**, 3547 (1992).

[65] F. Mauri, G. Galli, and R. Car, Phys. Rev. B **47**, 9973 (1993).

[66] X.-P. Li, R. W. Nunes, and D. Vanderbilt, Phys. Rev. B **47**, 10891 (1993).

[67] M. S. Daw, Phys. Rev. B **47**, 10895 (1993).

[68] P. Ordejon, D. A. Drabold, M. Grumbach, and R. Martin, Phys. Rev. B **48**, 14646 (1993).

[69] F. Mauri and G. Galli, Phys. Rev. B **50**, 4316 (1994).

[70] S. Goedecker and L. Colombo, Phys. Rev. Lett. **73**, 122 (1994).

[71] N. A. Modine, G. Zumbach, and E. Kaxiras, in *Materials Research Society Proceedings*, edited by E. Kaxiras, J. Joannopoulos, P. Vashishta, and R. Kalia (Materials Research Society, Pittsburgh, 1996), Vol. 408.

[72] G. Zumbach, N. A. Modine, and E. Kaxiras, in press (unpublished).

[73] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes*, 2nd ed. (Cambridge University Press, New York, 1992), Chap. 2.7.

[74] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes*, 2nd ed. (Cambridge University Press, New York, 1992), Chap. 19.6.

[75] R. Schlier and H. Farnsworth, J. Chem. Phys. **30**, 917 (1959).

[76] D. J. Chadi, Phys. Rev. Lett. **43**, 43 (1979).

[77] R. Tromp, R. Hamers, and J. Demuth, Phys. Rev. Lett. **55**, 1303 (1985).

[78] R. J. Hamers and R. M. T. J. E. Demuth, Phys. Rev. B **34**, 5343 (1986).

[79] R. A. Wolkow, Phys. Rev. Lett. **68**, 2636 (1992).

[80] M. T. Yin and M. L. Cohen, Phys. Rev. B **24**, 2303 (1981).

[81] J. Dabrowski and M. Scheffler, Appl. Surf. Sci. **56–58**, 15 (1992).

[82] K. Cho and J. D. Joannopoulos, Phys. Rev. Lett. **71**, 1387 (1993).

[83] J. Ihm, D. H. Lee, J. D. Joannopoulos, and J. J. Xiong, Phys. Rev. Lett. **51**, 1872 (1983).

[84] D. Kandel and E. Kaxiras, Phys. Rev. Lett. **75**, 2742 (1995).