2013-05-09

Cluster Expansion Models Via Bayesian Compressive Sensing

Lance Jacob Nelson
Brigham Young University - Provo

Cluster Expansion Models Via Bayesian Compressive Sensing

Lance J. Nelson

A dissertation submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Gus L. W. Hart, Chair
Bret Hess
Karine Chesnel
John Colton
David Neilsen

Department of Physics and Astronomy

Brigham Young University

May 2013

ABSTRACT

Cluster Expansion Models Via Bayesian Compressive Sensing

Lance J. Nelson
Department of Physics and Astronomy, BYU
Doctor of Philosophy

The steady march of new technology depends crucially on our ability to discover and design new, advanced materials. Partially due to increases in computing power, computational methods are now having an increased role in this disovery process. Advances in this area speed the discovery and development of advanced materials by guiding experimental work down fruitful paths. Density functional theory (DFT) has proven to be a highly accurate tool for computing material properties. However, due to its computational cost and complexity, DFT is unsuited to performing exhaustive searches over many candidate materials or for extracting thermodynamic information. To perform these types of searches requires that we construct a fast, yet accurate model. One model commonly used in materials science is the cluster expansion, which can compute the energy, or another relevant physical property, of millions of derivative superstructures quickly and accurately. This model has been used in materials research for many years with great success.

Currently the construction of a cluster expansion model presents several noteworthy challenges. While these challenges have obviously not prevented the method from being useful, addressing them will result in a big payoff in speed and accuracy. Two of the most glaring challenges encountered when constructing a cluster expansion model include: (i) determining which of the infinite number of clusters to include in the expansion, and (ii) deciding which atomic configurations to use for training data. Compressive sensing, a recently-developed technique in the signal processing community, is uniquely suited to address both of these challenges. Compressive sensing (CS) allows essentially all possible basis (cluster) functions to be included in the analysis and offers a specific recipe for choosing atomic configurations to be used for training data. We show that cluster expansion models constructed using CS predict more accurately than current state-of-the art methods, require little user intervention during the construction process, and are orders-of-magnitude faster than current methods. A Bayesian implementation of CS is found to be even faster than the typical constrained optimization approach, is free of any user-optimized parameters, and naturally produces error bars on the predictions made. The speed and hands-off nature of Bayesian compressive sensing (BCS) makes it a valuable tool for automatically constructing models for many different materials. Combining BCS with high-throughput data sets of binary alloy data, we automatically construct CE models for all binary alloy systems. This work represents a major stride in materials science and advanced materials development.

Keywords: cluster expansion, density functional theory (DFT), compressive sensing, Bayesian

ACKNOWLEDGMENTS

I, the author, wish to thank

... Gus L. W. Hart for being an outstanding mentor, an excellent scientist, and a kind friend. For guiding my research on what turned out to be a very fruitful path. For teaching me how to write and speak the language of science. For taking me on several scientific trips across both states and continents. For spending countless hours "driving" (coding) while I spouted off instructions. For putting up with my constant habit of pulling us off topic to some other interesting scientific question.

... My wife for always being my cheerleader, especially at times when I was feeling down and discouraged. For forgiving me for having my mind miles from home because I was still thinking about that hard enumeration problem. For acting excited about my scientific endeavors, even when it probably wasn't that exciting to you. For holding the fort down at home and being the wonderful mother that you are. I couldn't have accomplished this goal without you.

... My children for being excited when I come home, and for missing me when I was gone. For motivating me to be the best I can be so that you have someone worthy of emulation for your father.

... My parents for always encouraging me in my goals and dreams.

... Rodney Bain who once said to me, " I'll be disappointed in you if you don't go on and earn your PhD." Many times when I felt like quitting, the thought of disappointing Brother Bain

has given me the motivation to go on.

.. Richard Hatt who taught my very first physics class and thereby convinced me that this should be my field of study. Your explanations and knowledge of physics made me want to obtain an equivalent understanding.

Table of Contents

List of Tables viii

List of Figures ix

1 Introduction 1

2 Density functional theory 6

2.1 The many-body Schrödinger equation 6

2.2 Density functional theory: An alternative to solving the Schrödinger equation . . . 9

2.2.1 Proof of first H-K theorem 9

2.2.2 Proof of second H-K theorem 11

2.3 The energy functional 12

2.4 The exchange-correlation functional 14

2.5 Practical details 15

2.5.1 Plane wave basis and energy cutoff 16

2.5.2 Pseudopotentials 18

2.5.3 Brillioun zone integration 19

3 Cluster Expansion 23

3.1 Motivation for building a model 23

3.2 Definition of the basis 24

3.3 Using the basis . . . 28
3.4 Review of current techniques . . . 30
3.5 Conclusion . . . 33

4 Compressive Sensing **35**
4.1 Introduction . . . 35
4.2 Compressive sensing: an illustration . . . 36
4.3 Cluster Expansion . . . 39
4.3.1 Energy Model . . . 39
4.3.2 Compressive sensing cluster expansion (CSCE) . . . 41
4.4 Practical aspects of ℓ_1-based optimization . . . 43
4.4.1 Fixed-point continuation . . . 43
4.4.2 Bregman iteration . . . 44
4.4.3 Split Bregman iteration . . . 45
4.4.4 Choice of structures for CSCE . . . 46
4.4.5 Effect of noise and its relation to optimal μ . . . 48
4.5 Applications . . . 52
4.5.1 Short-ranged pair model with noise . . . 52
4.5.2 Actual alloy example: Ag-Pt . . . 54
4.5.3 Statistical analysis of Ag-Pt ECI's . . . 57
4.5.4 Protein folding application . . . 60
4.6 Conclusion . . . 62

5 Bayesian Compressive Sensing **63**
5.1 Introduction . . . 63
5.2 The cluster expansion . . . 64

5.3 Compressed sensing 66
5.4 A Bayesian Implementation 69
5.4.1 Enhancing the sparsity through re-weighted ℓ_1 norm minimization 70
5.5 Application 71
5.6 Conclusion 73

6 CEFlash: high-throughput CE model construction **75**
6.1 Introduction 75
6.2 Large-scale construction of cluster expansion models 78
6.2.1 Training set selection 78
6.2.2 Choice of *k*-points 85
6.3 Lattice models for all binary alloys: results from a few select systems 90
6.4 Summary and Outlook 92

7 Conclusion **94**
7.1 Summary 94
7.2 Outlook 95

Bibliography **97**

A Bayesian Statistics **103**
A.1 Bayesian compressive sensing 105

List of Tables

6.1 Model-quality results for the binary systems Ag-Pt, Ag-Al, Cu-Pt, and Ag-Pt on an fcc lattice. The rms, $\|J\|_1$, and $\|J\|_0$ are averages over 100 different choices of training set. 90

List of Figures

1.1 Publications in computational materials science over the last 37 years. The increased activity in this area can be attributed to increases in computing power and to the discovery of density functional theory, which made the many-body problem tractable. 2

2.1 Illustration of the self-consistent cycle used to solve the KS equations given by equation (2.35). 14

2.2 The pseudopotential (dashed, red line) is constructed to approximate the true potential (solid, black line) to a high degree of accuracy outside some cutoff radius. The justification for using such a pseudopotential is that core electrons do not contribute significantly to the chemical bonding in materials. 19

2.3 Illustration of the Monkhorst-Pack scheme for choosing *k*-points. Rectangular (left) and triangular (right) reciprocal unit cells are shown. For the rectangular unit cell, the *k*-points mesh is defined by dividing one lattice vector into 4 divisions and the other into 3, creating a mesh of uniform density. For the triangular unit cell, the mesh is defined by dividing both lattice vectors into 4 divisions. In each case the choice of division is dictated by the requirement that the density of the mesh be uniform. 21

2.4 Illustration of Froyen's equivalent scheme for choosing *k*-points. The black dots indicate the *k*-points mesh with the black and red polygons being reciprocal unit cells commensurate with the mesh chosen. The mesh of *k*-points shown will be used for both reciprocal unit cells depicted here. Using the same *k*-points mesh is theorized to reduce systematic error. 22

3.1 The number of unique, fcc-based derivative superstructures as a function of unit cell size, ranging from 1 atom/cell to 20 atoms/cell. Since DFT-based methods can be computationally costly, exhaustive exploration of derivative superstructures requires the use of a model. 24

3.2 Thirty-five derivative superstructures derived from a square lattice. The atoms of each configuration lie on the sites of a square lattice and are unique configurations. Fcc, bcc, and hcp-derived superstructures are commonly observed in nature, which motivates their study in computational research. 25

3.3 Illustration of all possible combinations of point functions on a nearest neighbor pair cluster on a square lattice. The (0,0) combination is always a constant regardless of atomic occupation. The $(0,1)$ and $(1,0)$ combinations are equivalent to the point cluster. The only unique pair cluster function here is the $(1,1)$ combination. . 27

3.4 Illustration of geometrically unique clusters on a square lattice. The cluster basis is constructed from two components: (i) geometrically distinct clusters of lattice sites (depicted here) and (ii) all possible combinations of point functions on those lattice sites. 28

3.5 Geometrically unique clusters of fcc lattice sites. The average distance from the center of mass of the cluster increases moving left to right. 29

3.6 Symmetrically equivalent versions of the nearest neighbor pair cluster on a square lattice. When constructing the cluster functions, the point function products are averaged over all symmetrically equivalent versions of the cluster. In this example there would be four terms in the average. 30

3.7 Illustration of how the basis function corresponding to the nearest neighbor pair cluster is averaged over all unique sites in the crystal. The atomic configuration shown has two unique sites and the unit cell is given by the rectangle. In the figure on the left, all rotationally equivalent versions of the pair cluster are constructed (see figure 3.6). In the figure on the right, the sites of the pair cluster are translated to the other unique lattice site. 31

3.8 Illustration of the linear algebra problem that emerges when constructing a cluster expansion. Most notably, the problem is heavily underdetermined due to the vast number of possibly-relevant basis functions. 32

4.1 (a) A sparse signal (blue line) like that of Eq. 4.2, uniform samples of the signal at the Nyquist frequency (red dots), and a few random samples (black circles). The signal is composed of only 3 non-zero frequencies. (b) Exact recovery of the frequency components of the signal using compressive sensing. 36

4.2 $\|J_{\text{exact}} - J_{\text{fit}}\|_1$(solid) and $\|J_{\text{fit}}\|_0$(dashed) vs $\log_{10}\mu$ for the short-ranged pair model with $M = 200$ (a) and $M = 400$ (b). Random uniform noise of $\sim 10\%$(blue circles), 20% (green squares), and 50% (red "x"s) of the noiseless energies was added to the fitting structures. (c) $\|J_{\text{exact}} - J_{\text{fit}}\|_1$ vs the number of fitting structures and the noise level. Each point represents an average over $\sim$100 different subsets of M structures. 49

4.3 Root-mean-square errors for the prediction set (black line with empty squares) and the leave-one-out cross-validation score (LOOCV, solid blue line) as functions of the parameter μ. LOOCV has been averaged over 10 randomly drawn sets of 100 (400) structures, and the error bars were calculated from the variance in the predicted LOOCV scores over these sets. Predictive errors for the hold-out set and the fitting errors for the training set were averaged over 500 different sets of 100 (400) structures; the corresponding error bars are smaller than the size of the symbols. 55

4.4 Results from compressive sensing and leave-one-out cross-validation for the fcc-based, Ag-Pt alloy system. The solid line gives the root-mean-square (RMS) errors for predictions made on a constant holdout set for CS (box and whisker) and leave-one-out cross-validation (squares). The dashed lines give the ℓ_1-norm of the solution vector for both methods. 57

4.5 Comparison of the interaction coefficients found using the DO method implemented in ATAT software and compressive sensing. The upper pane shows a comparison of two typical fits from CS and ATAT. The lower pane shows the coefficients that were found to be statistically relevant from both methods. The x-axis is the cluster radius, which is defined as the average distance from the center of mass of all cluster vertices. (Blue dots were placed on the x-axis even for clusters not found to be relevant to help the reader know the ordinal number of the relevant clusters.) Physical intuition suggests that shorter-radius, fewer-vertex clusters are the most important contributors in alloy energetics. Pair interaction coefficients found by both methods are similar. As the number of vertices increases, CS finds coefficients in harmony with physical intuition, while DO finds spurious, long-ranged three- and four-body interactions. CS solutions also demonstrate a convergence to one specific solution as the size of the fitting set increases. (note: Triplets and quadruplets are shown on a scale from -20 to 20 meV, different from the scale used for the pairs.) 58

4.6 Predicted CSCE formation energies obtained using the ECI's shown in Fig. 4.5; error bars are standard deviation due to different random choices of ≤ 400 structure subsets. Black solid line denotes the convex hull calculated from the average energies; only Ag, Ca_7Ge-type Ag_7Pt (barely, with a depth of less than 1 meV/atom), $L1_1$ AgPt, and Pt are predicted to be $T = 0$ K ground states. 60

4.7 Predictive performance of CS for protein energetics in the zinc-finger structure (shown in the inset). 61

5.1 Histogram of geometrically unique clusters on an fcc lattice. The x axis is the cluster radius, which is defined to be the average distance from the cluster center of mass to the cluster vertices. The number of unique clusters increases exponentially as the number of cluster vertices and cluster radius increase. This illustrates the magnitude of the challenge associated with truncating the cluster expansion. . . . 65

5.2 Illustration of constant ℓ_p norm surfaces in R^2. The circle is a constant ℓ_2 norm surface and the diamond is a constant ℓ_1 norm surface The straight line indicates the possible solutions to the underdetermined problem $10y + 7x = 20$. A sparse solution to this problem is the solution where one of the variables is zero and the other is not, in other words it is at the intersection of the straight line and the axes. Minimizing the ℓ_2 norm of this system will result in a dense solution, whereas minimizing the ℓ_1 norm will yield a sparse solution. 66

5.3 Comparison between re-weighted Bayesian compressive sensing and genetic algorithm methods for constructing a cluster expansion model for the binary systems Cu-Pt, Ag-Pt, and Ag-Pd. The solid curves indicate rmse values over a holdout set. The dashed curves represent the ℓ_1 norm of the solution vectors. Approximately 100 BCS fits were performed at each training set size, and the results of these fits are depicted using box-and-whiskers. Due to it's high computational cost, only 5 GA fits were performed, and hence GA results are not depicted using box-and-whiskers. 72

6.1 Simple one dimensional function representing a signal in time. The red dots are regular samples according the Nyquist's theorem. The figure on the right is a plot of the sensing matrix $\mathbb{A}$ from Eq. (6.5) 79

6.2 The red dots indicate regular samples over the function domain. The blue (black) curve is a cosine function with frequency 2 (28) Hz. Due to the sampling rate employed here, the information contained in the samples is redundant between the two basis functions shown. This is known as aliasing. 80

6.3 Sensing matrix constructed by sampling the function at random locations in its domain. The function on the right contained frequencies: 3, 4, and 7. Recovery of this signal with a 10 x 10 sensing matrix would not be possible with standard Fourier transform tequniques. However, by ensuring that the entries in the sensing matrix are random, the signal can be recovered exactly. 81

6.4 Histograms of the value of the 1st, 2nd, and 3rd, nearest neighbor pair cluster functions over all fcc-derived superstructures up to 12 atoms/cell. Most noteworthy is the fact that the cluster function values are not uniformly distributed. Also, note that there are regions of values which never occur over this set of structures. These points make it challenging to construct a sensing matrix composed of random, uniformly distributed entries. 82

6.5 Comparison of three different training set selection methods: choosing structures at random (top), the method of reference1 (middle), and the method discussed in this chapter (bottom). The matrices depicted on the left are cross correlation matrices, and the off-diagonal terms are indicative of how correlated structures are to one another. The histograms on the right show the distribution of off-diagonal cross correlation values. The method described in this work yields lower off-diagonal cross correlations and therefore lower-coherence sets of training structures. 84

6.6 Illustration of the Monkhorst-Pack scheme for choosing *k*-points. Rectangular (left) and triangular (right) reciprocal unit cells are shown. For the rectangular unit cell, the *k*-points mesh is defined by dividing one lattice vector into 4 divisions and the other into 3, creating a mesh of uniform density. For the triangular unit cell, the mesh is defined by dividing both lattice vectors into 4 divisions. . . . 86

6.7 Illustration of Froyen's equivalent scheme for choosing *k*-points. The black dots indicate the *k*-points mesh with the black and red polygons being reciprocal unit cells commensurate with the mesh chosen. The mesh of *k*-points shown will be used for both reciprocal unit cells depicted here. Using the same *k*-points mesh is theorized to reduce systematic error. 87

6.8 Comparison of systematic errors associated with the choice of *k*-point meshes. Three methods for choosing *k*-point meshes are depicted: MP with a density of 600 KPPRA, and 10,000 KPPRA and the equivalent scheme of Froyen. Clearly the equivalent scheme results in smaller systematic error. However for high enough densities the MP method appears to be sufficiently accurate. 88

6.9 Illustration of two different choices of unit cell for the same 2D atomic configuaration. The unit cell on the left has the shortest, most orthogonal lattice vectors. The unit cell on the left provides a perfectly correct description of the crystal, but this choice of unit cell should not be used in DFT calculations. 89

6.10 Ground state search over all fcc-derived superstructures up to 12 atoms/cell for the binary system Ag-Pt. The green line is the convex hull and indicates the ground states of the system. 91

6.11 This is a snippet of the file which provides a list of all relevant cluster functions and their coefficients. Information provided in this file includes the coordinates of the cluster vertices, point functions to be evaluated on each cluster vertex, and the associated model coefficient. Soon models for hundreds of alloy systems will become available to the general public. 92

A.1 Illustration of Bayes' rule. It is reasonable to assume that the distribution of University students' heights be a Normal (Gaussian) distribution. However, the location and width of this distribution are unknown. Shown are three possible Normal distributions that could represent these heights. Bayes' rule provides distributions on these values, the mean and width of the likelihood, indicating what values are likely for these parameters. Bayes' rule weighs both the data provided and the prior information about the parameters of interest. 104

Chapter 1

Introduction

The field of computational solid state physics/materials science is largely concerned with the understanding and predicting of the physical properties of solid material via computer calculations and simulations. The material in question could be something as small as a nanoparticle or as large as a bulk alloy. The drive for materials research in general stems from the ever-increasing demand for new technologies, which rely heavily on the discovery and successful fabrication of high-performing materials. Due to remarkable advances in computing power as well as significant methodological/algorithmic strides, the role of computation in materials research has increased dramatically over the last half century. Computational findings in the materials arena are a boon to metallurgists and experimental scientists, providing valuable direction for materials synthesis and avenues for future research.

In theory, all physical properties of a material can be obtained by solving the Schrödinger equation for the system comprised of all ions and electrons that make up the material. As any undergraduate physics student knows, the solution to this equation can be easily obtained for many simple problems (i.e., particle in a box, harmonic oscillator, hydrogen atom, etc). However, solving Schrödinger's equation for even the simplest of materials problems presents a considerable challenge. For example, if we were interested in a single molecule of CO_2, the electronic wavefunction would be 66-dimensional, 3 dimensions for each of the 22 electrons. Consideration of nanoclusters, which can consist of thousands of atoms, requires the determination of an electron wavefunction that has hundreds of thousands of dimensions. This should give the reader a first clue as to the complexity of this problem.

In 1998 Walter Kohn and John Pople were jointly awarded the Nobel prize in chemistry for their discovery of density functional theory (DFT) and quantum chemistry computer code development. The class of numerical methods they discovered has benefited the field of computational

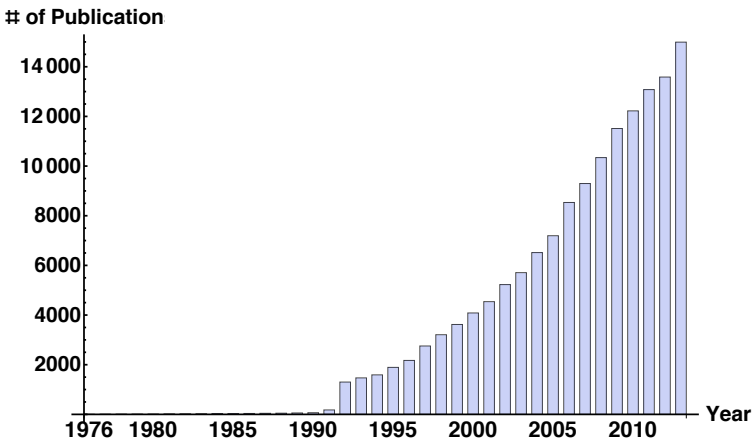


Figure 1.1: Publications in computational materials science over the last 37 years. The increased activity in this area can be attributed to increases in computing power and to the discovery of density functional theory, which made the many-body problem tractable.

materials research immensely. As a side note, it was the first time in history that a nobel prize was awarded for a numerical method rather than a purely scientific discovery. The theorems of DFT were put forth by Hohenberg, Kohn, and Sham in the 1960's, and reduced the many-interacting-electron problem to a system of independent electrons without any approximations. First, Hohenberg and Kohn proved that the ground state energy is a unique functional of the electron density and that this density can be found by minimizing the functional with respect to the density. To make the minimization tractable Kohn and Sham proposed considering a fictitious system of independent electrons. The variational principle is applied to the resulting functional to produce a Schrödinger-like equation for single electrons. All errors introduced by considering the electrons as classical, independent particles are wrapped into an exchange-correlation energy, which must be approximated. DFT has become a pillar of modern-day computational materials science and most modern-day computational materials research methods employ DFT-based calculations in some way (see Figure 1.1). Chapter 1 in this dissertation reviews modern-day density functional theory, the theorems that form its foundation, and some practical details.

Many materials problems of practical importance find themselves well beyond the scope of DFT-based methods. For example, one subset of computational materials problems involves exploring all crystal structures whose atoms are restricted to lie on a parent lattice, called derivative superstructures. The motivation for studying such groups of materials stems from the fact that many experimentally observed crystal structures fall into this category. However, the number of unique derivative superstructures for any given lattice increases rapidly with the unit cell size to

include millions of atomic configurations. The sheer size of such searches put them well beyond the reach of computationally-costly, DFT-based methods.

To perform large searches, one approach is to build a model, trained from DFT data, but which is much simpler mathematically and can therefore compute much faster. One such model commonly used to explore derivative superstructures is the cluster expansion. The cluster expansion expresses the energy of any atomic configuration (restricted to the parent lattice) as a linear combination of cluster energies, or energies of small clusters of atoms on the lattice. Due to its mathematical simplicity, the cluster expansion model can accurately compute the energies of millions of derivative superstructures in only minutes, a vast speed-up over DFT.

Assessing the thermodynamic stability and physical usefulness of a material at temperatures greater than zero requires consideration of the free energy: $F = U - TS$. This is beyond the scope of time-independent DFT, and a common approach for computing the free energy involves Monte-Carlo-like simulations which can require millions of energy calculations. These simulations would be unfeasible to perform without a fast model for computing energies, providing another key motivator for wanting a fast, accurate model.

Several noteworthy practical challenges currently exist in the cluster expansion construction process. The first is deciding how to truncate the expansion. This is challenging because the number of unique clusters is very large and there is no way to know *a priori* which terms will be dominant contributors for a given system. Most modern techniques for accomplishing this involve the use of physical intuition and/or complex algorithms. The second challenge is deciding which crystal structures to use as training data. A subtle point here is that these two challenges are not independent of one another, but must be addressed jointly. Design of robust, efficient methods to address these questions has remained a challenge for many years. Chapter 3 of this dissertation provides a description of the mathematical foundation of the cluster expansion and a summary of the prevailing methods for constructing such models.

Since the number of possibly-relevant basis functions (clusters) is much larger than the number of DFT calculations that are feasible to perform for a single system, the problem of constructing a cluster expansion model naturally emerges as an underdetermined linear algebra problem,

$$\mathbb{A}\mathbf{x} = \mathbf{b}, \tag{1.1}$$

where each column (row) in $\mathbb{A}$ corresponds to a basis function (training data point/DFT calculation), and the number of rows is much less than the number of columns. An underdetermined problem is challenging to solve because there are an infinite number of solutions that are consistent with the data provided.

A recently-developed technique from the signal processing community, compressive sensing (CS), proposes solving this underdetermined problem by constraining the solution search to solutions whose ℓ_1 norm is minimal. Of all the solutions that are consistent with the data, the ℓ_1 norm constraint identifies the solution with the fewest non-zero coefficients, or most sparse solution. This new paradigm presented by CS is uniquely well-suited to solve the above-mentioned cluster expansion challenges. CS solves the truncation problem by including essentially all possible basis functions. The mathematical theorems forming the foundation of CS also dictate the form of the matrix $\mathbb{A}$, providing a mathematically proven recipe for choosing training data. CS is robust and provides an efficient method for identifying relevant basis functions (out of a very large pool of contenders), and computing their associated coefficients.

Various mathematical implementations of CS currently exist. In a one-parameter formulation, a constrained minimization problem is recast as an unconstrained problem, with the single parameter controlling the sparseness of the solution. Another implementation of CS involves the use of Bayesian statistics. This implementation offers a parameterless framework (automatic), vast speed increases from current state-of-the art methods, and error bars on solutions. A weighted formulation of CS further enhances the sparsity of the solution and reduces the amount of training data needed. Chapters 4 and 5 in this dissertation provide further details about these implementations in the context of cluster expansion models. A comparison between CS-based CEs and other prevailing methods is also given. CS is found to produce cluster expansion models that predict more accurately than current state-of-the art methods, and are orders of magnitude faster than the current state-of-the-art.

One modern approach to uncovering high-performing materials is to simply compute, using DFT, the property of interest for all candidate materials (as stability is of fundamental importance, the chemical energy is routinely computed). By identifying those crystal structures which appear most frequently in nature, a database of candidate crystal structures can be assembled. Using an automatic framework for performing DFT calculations, the energy is computed for all crystal

structures in the database and for all possible combinations of atoms. The resulting database of first-principles data can then be mined for new, advanced materials. This approach is commonly referred to as *high throughput* and it relies heavily on being able to **automatically** perform large numbers of calculations with minimal human oversight. Results from high-throughput studies have been fruitful and beneficial [2].

Previously, the inclusion of materials models, like the cluster expansion, in high-throughput databases has not been possible. This is mostly because the aforementioned challenges have made the model building process cumbersome and human-time-intensive, requiring hours of user time to construct a high-quality model for a single system. However, the discovery of CS as a fast, efficient, and automatic way to build CE models has made the inclusion of material models in high-throughput databases feasible. Chapter 6 discusses the details of this endeavor and the asset this database will be to the materials science community.

Chapter 2

Density functional theory

2.1 The many-body Schrödinger equation

One common goal of materials scientists is to understand and predict the physical properties of material. While experimental results are insightful, the materials science theorist seeks to predict or compute materials properties starting from basic physical and mathematical theories. The material of interest could be as simple as a single atom or as complex as a bulk solid made up of a large array of atoms. This complex arrangement of atomic particles is inherently a quantum mechanical problem, and requires finding a solution to Schrödinger's equation. As we shall see, for everything except the most simple systems, a straight-forward solution to Schrödinger's equation is impossible.

Consider a system comprised of N atoms. To define where an atom is located requires that we define both where the nucleus and the atom's electrons are. A quantity of fundamental importance is the energy of the atomic configuration and perhaps how this energy changes as the atoms move to different positions. This energy can be found by finding a solution to the time-independent, non-relativistic Schrödinger equation which, for such a system, is given by

$$H\Psi(\mathbf{R_1},\mathbf{R_2}..\mathbf{R_N},\mathbf{r_1},\mathbf{r_2}..\mathbf{r_n}) = E\Psi(\mathbf{R_1},\mathbf{R_2}..\mathbf{R_N},\mathbf{r_1},\mathbf{r_2}..\mathbf{r_n}), \tag{2.1}$$

where R_i is the position of ion i and r_j is the position of electron j. A key realization is that the ions are orders of magnitude heavier than the electrons and therefore react much more slowly to changes in their environment. This allows the problem to be divided into two parts. First, the massive ions are assumed to remain fixed in their positions. The resulting problem is solved and the wavefunction describes the behavior of the electrons moving in the presence of the potential created by the ions and the other electrons. Changes in energy as the ionic positions change can be

explored by performing multiple calculations, each one with the ions fixed in different locations. The division of this problem into two parts is called the Born-Oppenheimer approximation, which reduces the dimensionality of the problem from $3M + 3MN$ (for M atoms and N electrons per atom) to $3N$. Under this approximation the Schrödinger equation becomes

$$H\Psi(\mathbf{r_1}, \mathbf{r_2}..\mathbf{r_N}) = E\Psi(\mathbf{r_1}, \mathbf{r_2}..\mathbf{r_N}), \tag{2.2}$$

where N is the number of electrons in the material. Notice that the wavefunction is a function of the electronic positions but not the ionic positions. The Born-Oppenheimer approximation reduces the dimensionality of the wavefunction to 3N variables, 3 coordinates for each electron, which is still an extremely large number even for few-atom materials such as a molecule or nanoparticle. The Hamiltonian operator, H, consists of three terms: The kinetic energy of the electrons, the interaction of the electrons with the external potential, and the electron-electron interaction

$$H = T + V_{\text{ext}} + V_{\text{ee}} \tag{2.3}$$

$$= \sum_{i}^{N} -\frac{\hbar^2 \nabla_i^2}{2m} + V_{\text{ext}} + \sum_{i<j} \frac{q^2}{|\mathbf{r_i} - \mathbf{r_j}|}. \tag{2.4}$$

For solids, the external potential is the interaction between the ions and electrons in the solid

$$V_{\text{ext}} = \sum_{i,k} \frac{Qq}{|\mathbf{r_i} - \mathbf{R_k}|}. \tag{2.5}$$

Here, $\mathbf{r_i}$ is the position of electron i and $\mathbf{R_k}$ is the location of ion k and the double sum is over all electrons and ions in the solid. Note that this term in the Hamiltonian is the only term involving interactions between particles other than electrons, and is the only way to distinguish between a lattice decorated with Ag and Pt atoms and a lattice decorated with Ni and Al atoms, for example. The operators T and V_{ee} are independent of the external potential as they only involve the electrons and not the ions.

Putting everything together yields the following many-body Schrödinger equation

$$\left(\sum_{i}^{N} -\frac{\hbar^2 \nabla_i^2}{2m} + \sum_{i,k} \frac{Qq}{|\mathbf{r_i} - \mathbf{R_k}|} + \sum_{i<j} \frac{q^2}{|\mathbf{r_i} - \mathbf{r_j}|} \right) \Psi(\mathbf{r_1}, \mathbf{r_2}..\mathbf{r_N}) = E\Psi(\mathbf{r_1}, \mathbf{r_2}..\mathbf{r_N}). \tag{2.6}$$

While conceptually simple, finding a solution to this equation using straight-forward techniques is practically impossible. Specifically, the second and third terms add considerable complexity because they involve physical interactions with electrons. In order to fully express these terms, the locations of the electrons must be known. But the locations of the electrons can only be obtained from the *yet-unknown* electron wavefunction. Thus, in addition to being extremely high dimensional, the exact form of the Hamiltonian can't even be fully expressed without first knowing the answer. In other words, the Schrödinger equation is a many-body problem.

One way to attack this problem is to assume that the electron-electron energy can be approximated as

$$V_H(\mathbf{r}) = e^2 \int \frac{n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|d\mathbf{r}'}. \tag{2.7}$$

This energy, called the Hartree energy, says that each electron feels the average, rather than instantaneous, effect of the other electrons. The many-body wavefunction is then constructed as a Slater determinant of one-particle orbitals, $\psi_j(x_i)$

$$\Psi(\mathbf{r_1}, \mathbf{r_2}..\mathbf{r_N}) = \psi_1(\mathbf{r}_1)\psi_2(\mathbf{r}_2)\cdots\psi_N(\mathbf{r}_N) - \psi_1(\mathbf{r}_2)\psi_2(\mathbf{r}_1)\cdots\psi_N(\mathbf{r}_N) + \cdots \tag{2.8}$$

$$= \begin{vmatrix} \psi_1(\mathbf{r}_1) & \psi_1(\mathbf{r}_2)\ldots & \psi_1(\mathbf{r}_3) \\ \psi_2(\mathbf{r}_1) & \psi_2(\mathbf{r}_2)\ldots & \psi_2(\mathbf{r}_3) \\ . & . & . \\ . & . & . \\ \psi_N(\mathbf{r}_1) & \psi_N(\mathbf{r}_2)\ldots & \psi_N(\mathbf{r}_3) \end{vmatrix}. \tag{2.9}$$

This construction ensures that the wavefunction satisfies the Pauli exclusion principle which requires that the sign of Ψ change when two of its arguments interchange. Use of this form of the wavefunction leads to a set of single-electron equations known as the Hartree-Fock equations. These equations include a complicated exchange term involving integrals of the form

$$\int V(\mathbf{r}, \mathbf{r}')\psi(\mathbf{r}')d\mathbf{r}'. \tag{2.10}$$

This term adds considerable complexity to the problem and generally makes a straightforward solution quite impossible.

2.2 Density functional theory: An alternative to solving the Schrödinger equation

In 1964 Hohenberg and Kohn proved two theorems that provided an alternative to solving the many-body Schrödinger equation while still remaining formally exact [3]. As the name suggests these theorems focus on finding the electron density instead of the many-body wavefunction and are the foundation of modern-day density functional theory (DFT). The following is a brief review of the mathematical foundation of DFT. The discussion employed here generally follows the same as Hart [4] and Sholl [5].

The first theorem states that the ground-state energy is a *unique* functional of the electron density. A functional is similar to a function, except instead of mapping a number to a number, a functional maps a function to a number. To distinguish it from a normal function, brackets instead of parenthesis are used to enclose the arguments of the functional. For example, the energy functional can be written as $E[n(\mathbf{r})]$ and takes as input the density function and returns the energy.

This theorem establishes a link between the three-dimensional electron density and the ground-state energy of the system of electrons. If the density corresponding to the ground-state of the system can somehow be determined, then from it can also be determined the ground-state energy. From this it follows that the Hamiltonian operator and ground-state wavefunction are also fully specified from a knowledge of the electron *density* alone. This theorem is useful because it shifts our focus away from searching for the many-electron wavefunction (3N-dimensional) and redirects it towards finding the electron density (3-dimensional).

2.2.1 Proof of first H-K theorem

Briefly stated, Hohenberg and Kohn's first theorem is that:

The ground state energy is a unique functional of the electron density

The proof of this theorem is straightforward and proceeds by *reductio ad absurdum*, which means that a false, or absurd result follows from a denial of the theorem. Let's assume that there are two potentials, $V_{\text{ext}}(\mathbf{r})$ and $V'_{\text{ext}}(\mathbf{r})$ which both result in the same electronic density $n_0(\mathbf{r})$. In other words, let's first assume that the theorem is incorrect, and see what emerges from this assumption. The Hamiltonian operators, H and H', for these two external potentials are

$$H = T + V_{\text{ee}}(\mathbf{r}) + V_{\text{ext}}(\mathbf{r}), \tag{2.11}$$

$$H' = T + V_{\text{ee}}(\mathbf{r}) + V'_{\text{ext}}(\mathbf{r}). \tag{2.12}$$

The ground-state energy of the unprimed Hamiltonian can be expressed as

$$E_0 = \langle \Psi_{\text{gs}} | H | \Psi_{\text{gs}} \rangle. \tag{2.13}$$

Now, if we replace the un-primed wavefunctions with the primed ones, we would expect that the result would be greater than the ground-state energy of the un-primed system

$$E_0 < \langle \Psi'_{\text{gs}} | H | \Psi'_{\text{gs}} \rangle. \tag{2.14}$$

We can write the un-primed Hamiltonian in terms of the primed Hamiltonian as:

$$H = H' - V' + V. \tag{2.15}$$

Inserting this into equation (2.14)

$$\begin{aligned} E_0 &< \langle \Psi'_{\text{gs}} | H' - V' + V | \Psi'_{\text{gs}} \rangle & (2.16)\\ &= E'_0 + \int n(\mathbf{r}) \left(V - V' \right) d^3\mathbf{r}. & (2.17) \end{aligned}$$

We can exchange the prime with the un-primed and get a similar result

$$\begin{aligned} E'_0 &< \langle \Psi_{\text{gs}} | H - V + V' | \Psi_{\text{gs}} \rangle & (2.18)\\ &= E_0 - \int n(\mathbf{r}) \left(V - V' \right) d^3\mathbf{r}. & (2.19) \end{aligned}$$

Adding these two inequalities together gives

$$\begin{aligned} E'_0 &< E_0 - \int n(\mathbf{r}) \left(V - V' \right) d^3\mathbf{r} & (2.20)\\ E_0 &< E'_0 + \int n(\mathbf{r}) \left(V - V' \right) d^3\mathbf{r} & (2.21)\\ & \qquad\qquad\qquad\qquad\qquad\qquad & (2.22)\\ E'_0 + E_0 &< E_0 + E'_0, & (2.23) \end{aligned}$$

which is obviously an absurdly false statement. Thus, the first H-K theorem is proven because to assume otherwise leads to an obviously false statement.

H-K's first theorem proves that the electron density uniquely determines all other important quantities, such as the energy, Hamiltonian operator and wavefunction, of the system. This theorem effectively trades the need to find the many-body wavefunction for the electron density. However, it doesn't specify what the energy functional looks like or provide a way to find the ground state electron density from it. Given an energy functional, the second H-K theorem provides a method for using the functional to find the ground state electron density. The second H-K theorem is crucial to making the first theorem useful. The second theorem involves defining a key property of the energy functional. It states that the electron density that *minimizes* the energy functional *is* the density corresponding to the solution to Schrödinger's equation for the ground-state of the system. Put briefly, the first theorem proves that an energy functional exists, and the second theorem provides a path towards using the functional to find the ground-state electron density. These two theorems form the foundation of modern density functional theory.

2.2.2 Proof of second H-K theorem

The second H-K theorem, which states that the density which minimizes the energy functional is the true density associated with the ground-state wavefunction, is also simple to prove. Suppose that $|\Psi\rangle(|\Psi'\rangle)$ is the ground-state wavefunction having density $n(\mathbf{r})(n'(\mathbf{r}))$ corresponding to the Hamiltonian $H(H')$. The ground-state energy of the unprimed system is given by

$$E_{\mathrm{gs}}[n(\mathbf{r})] = \langle\Psi|H|\Psi\rangle \tag{2.24}$$

$$= \langle\Psi|T + V_{\mathrm{ee}} + V_{\mathrm{ext}}|\Psi\rangle. \tag{2.25}$$

Since this energy is a minimum , we know that if we insert the unprimed wavefunction into this expression the result will be greater than this energy

$$E_{\mathrm{gs}}[n(\mathbf{r})] = \langle\Psi|H|\Psi\rangle \tag{2.26}$$

$$< \langle\Psi'|H|\Psi'\rangle, \tag{2.27}$$

which means that

$$E_{\rm gs}[n(\mathbf{r})] < E_{\rm gs}[n'(\mathbf{r})]. \tag{2.28}$$

This is really a trivial result, but reassures us that finding the density which minimizes the energy functional will yield the density corresponding to the ground-state of the system.

2.3 The energy functional

The energy functional proved by H-K to exist can be written as

$$E[n(\mathbf{r})] = T[n(\mathbf{r})] + V_{\rm ee}[n(\mathbf{r})] + V_{\rm ext}[n(\mathbf{r})]. \tag{2.29}$$

Here, the first term is the kinetic energy of the electrons, the second term is the energy associated with the interacting electrons and the third term is the energy associated with the interaction between the electrons and the ions. The form of the first two functionals are unknown, with the third functional having the form

$$V_{\rm ext}[n(\mathbf{r})] = \int n(\mathbf{r}) V_{\rm ext}(\mathbf{r}) d^3\mathbf{r}. \tag{2.30}$$

Even though H-K proved that an energy functional exists, we don't know the exact form of the functional, and even if we did, we have no well-defined recipe for minimizing this functional with respect to the electron density. To provide further traction to this problem Kohn and Sham proposed solving a slightly different problem, that of *N non-interacting* electrons [6]. Under this approximation, the kinetic energy functional can be written as

$$T_{\rm s}[n(\mathbf{r})] = -\frac{\hbar^2}{2m}\sum_i^N \langle \phi_i | \nabla^2 | \phi_i \rangle, \tag{2.31}$$

The subscript "s" is to remind us that this is not the true kinetic energy but rather the kinetic energy of *N* non-interacting electrons. A considerable portion of the electron-electron interaction energy is classical Coulomb,or Hartree energy, and can be written as

$$V_{\rm H}[n(\mathbf{r})] = \frac{1}{2}\int \frac{n(\mathbf{r_1})n(\mathbf{r_2})}{|\mathbf{r_1}-\mathbf{r_2}|} d\mathbf{r_1} d\mathbf{r_2}. \tag{2.32}$$

Putting all these pieces into the total energy functional gives

$$E[n(\mathbf{r})] = T_s[n(\mathbf{r})] + V_{\mathrm{H}}[n(\mathbf{r})] + V_{\mathrm{ext}}[n(\mathbf{r})] + E_{\mathrm{xc}}[n(\mathbf{r})], \tag{2.33}$$

where the last term is called the exchange-correlation energy and is defined as

$$E_{\mathrm{xc}}[n(\mathbf{r})] = (T[n(\mathbf{r})] - T_s[n(\mathbf{r})]) - (V_{\mathrm{ee}}[n(\mathbf{r})] - V_{\mathrm{H}}[n(\mathbf{r})]). \tag{2.34}$$

In words, the exchange-correlation energy is the sum of the errors introduced by using a non-interacting electron kinetic energy operator and by treating the electron-electron interaction classically.

Applying the variational principle to this functional yields a Schrödinger-like equation for single electrons

$$\left[-\frac{\hbar^2}{2m}\nabla^2 + V_{\mathrm{ext}}(\mathbf{r}) + V_H(\mathbf{r}) + E_{\mathrm{xc}}(\mathbf{r})\right]\psi_i(\mathbf{r}) = \varepsilon_i\psi_i(\mathbf{r}). \tag{2.35}$$

This set of equations is commonly referred to as the Kohn-Sham (KS) equations, and they are similar in form to equation (2.6). The main difference is that there is no sum over electrons and the equation involves single-electron wavefunctions instead of a many-body wavefunction.

Hohenberg and Kohn's theorems, which shift the focus away from the many-body wavefunction and towards minimizing an energy functional, combined with Kohn and Sham's approach for minimizing the energy functional, have resolved most of the complexities associated with solving equation (2.6). The many-body Schrödinger equation has been replaced with a set of single-electron Schrödinger-like equations, which is mathematically tractable and can be solved using traditional numerical techniques.

However, you may notice that there is something circular about the KS equations. To define the operator in equation (2.35) we need the electron density. But to find the density we must find the KS wavefunctions. To break this circle, the KS equations are solved using an iterative process (illustrated in figure 2.1). An initial electron density is chosen and used to construct the operator. With the operator constructed, the differential equation can then be solved, and the single-particle wavefunctions found. These wavefunctions are then used to compute a new electron charge density. If the new density is equal to the the previous density, then the problem is solved. Otherwise, a new

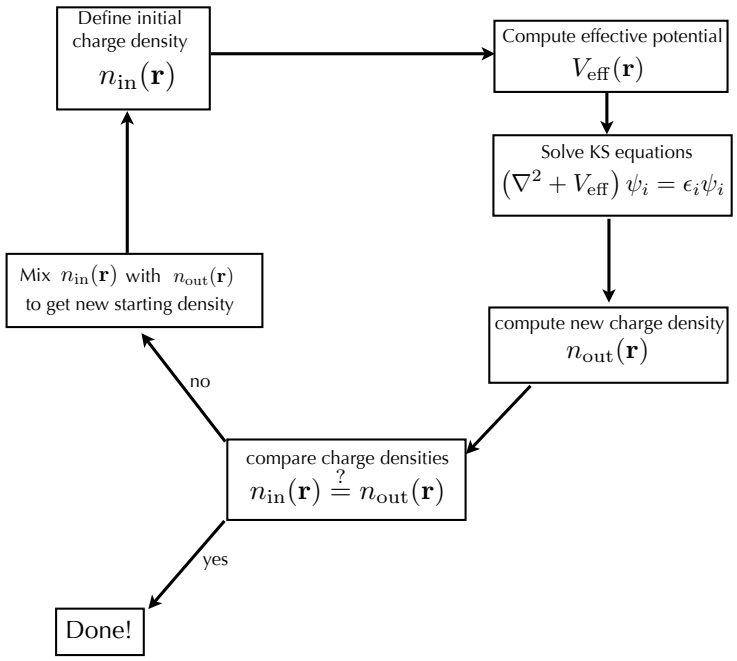


Figure 2.1: Illustration of the self-consistent cycle used to solve the KS equations given by equation (2.35).

density is constructed, typically by mixing the old and new densities somehow, and the algorithm starts over. This is called a self-consistent cycle because the convergence criteria is whether the starting point was consistent with the end result.

Let's review what we have learned so far. We'd like to find the ground-state energy of a complex collection of ions and electrons but a straightforward solution is impossible because it is a many-body problem. The theorems of Hohenberg, Kohn, and Sham show that an alternative to solving the many-body Schrödinger equation is to minimize an energy functional with respect to the electron density, and that this can be done by solving a set of Schrödinger-like equations for single electrons. With the exception of the exchange-correlation energy, $E_{\text{xc}}(\mathbf{r})$, all terms in the KS Hamiltonian operator can be easily written down.

2.4 The exchange-correlation functional

All errors introduced by considering the electrons as classical, independent particles are accounted for in the exchange-correlation energy, and to proceed we must specify this functional. Since everything that we know has already been written down, defining this energy is not a trivial task. The truth is that the true form of the exchange-correlation function is simply not known and work to find an approximate functional has been ongoing for many years.

One common way to approximate the exchange-correlation potential is to use the exchange-correlation potential of the uniform electron gas ($n(\mathbf{r}) = \text{constant}$)

$$E_{\text{xc}}(\mathbf{r}) = E_{\text{xc}}^{\text{uniform electron gas}}(\mathbf{r}). \tag{2.36}$$

While the uniform electron gas may seem unimportant for materials problems of interest, it provides a situation where the exact form of the exchange-correlation can be calculated. Since only the local electron density is employed in the approximation, it is called the local density approximation (LDA). The exchange energy of a uniform electron gas is known analytically to be

$$E_x^{\text{LDA}}[\rho] = -\frac{3}{4}\left(\frac{3}{\pi}\right)^{1/3}\int \rho(\mathbf{r})^{4/3} d\mathbf{r}. \tag{2.37}$$

However, the correlation energy is not known analytically except in the high and low density regimes and approximating this functional has been accomplished through the use of quantum Monte Carlo simultions. Various different parameterizations of the LDA are commonly used [7–10]. For magnetic systems a local spin density approximation has been employed [11]. The limitations of this approximation are well known, and generally speaking the LDA performs well for systems whose electron density is close to uniform or that is slowly-varying. LDA-based calculations typically underestimate the bandgap in semiconductors. The LSDA has incorrectly predicted the groundstates for certain magnetic compounds.

Improving upon the LDA has been attempted by including information about the gradient of the electron density in the functional, i.e. $E_{\text{xc}}(\mathbf{r}) \rightarrow E_{\text{xc}}(\mathbf{r}, \nabla\mathbf{r})$. Functional paramaterizations of this type are called generalized gradient approximations (GGA), and dozens of paramaterizations for doing this exist in the literature [12–16]. No matter the functional employed, it is important to remember that all are approximations to the true functionals and hence solutions to equation (2.35) obtained by using these approximate functionals are only approximate solutions.

2.5 Practical details

One particular implementation of DFT is contained in the VASP software, which stands for Vienna ab-initio simulation package. Since VASP was the primary tool for performing DFT

calculations used in this work, a few practical details associated with its use will be given here. However, some of the topics discussed here are general and used in many DFT implementations.

2.5.1 Plane wave basis and energy cutoff

The self-consistent approach for minimizing the energy functional can be broken into two parts. First, the KS equations (Eq. 2.35) are solved and the corresponding density is computed. Then the density is used to compute a new potential, which is then used when solving the updated KS equations. The second part, using the density to assemble the differential operator, is trivial, but the first part requires solving an ordinary differential equation.

One numerical approach for solving this equation involves expanding the wavefunction using a set of basis functions. From Bloch's theorem we know that the solutions to the equation

$$\left[-\frac{\hbar^2}{2m}\nabla^2 + V_{\text{ext}}(\mathbf{r}) + V_H(\mathbf{r}) + E_{\text{xc}}(\mathbf{r})\right]\psi_i(\mathbf{r}) = \varepsilon_i\psi_i(\mathbf{r}) \tag{2.38}$$

subject to periodic boundary conditions have the form

$$\psi_{n,\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k}\cdot\mathbf{r}}u_{n,\mathbf{k}}(\mathbf{r}), \tag{2.39}$$

where $u_{n,\mathbf{k}}(\mathbf{r})$ is a periodic function of the lattice, i.e., $u_{n,\mathbf{k}}(\mathbf{r}) = u_{n,\mathbf{k}}(\mathbf{r}+\mathbf{R})$ where $\mathbf{R}$ is a lattice vector. Stated in words, Bloch's theorem indicates that each single-electron state is a product of a planewave times a function periodic in the lattice. The periodic function is indexed by two variables, n, and $\mathbf{k}$. States associated with a single value of n vary continuously with the vector $\mathbf{k}$ and form a band of states. The index n is the so-called band index because for each value of n there are a band of electronic states. Solving equation (2.38) yields a set of n states, one for each band.

The unknown part of the Bloch function is the periodic function $u_{n,\mathbf{k}}(\mathbf{r})$ and we can express this function using a set of basis functions, $\phi_j(\mathbf{r})$

$$u_{n,\mathbf{k}}(\mathbf{r}) = \sum_{\mathbf{j}} c_{j,n,\mathbf{k}}\phi_j(\mathbf{r}). \tag{2.40}$$

Inserting this into the Bloch functions gives

$$\psi_{n,\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k}\cdot\mathbf{r}} \sum_{\mathbf{j}} c_{j,n,\mathbf{k}} \phi_j(\mathbf{r}) \tag{2.41}$$

$$= \sum_{\mathbf{j}} c_{j,n,\mathbf{k}} \bar{\phi}_{j,\mathbf{k}}(\mathbf{r}), \tag{2.42}$$

where $\bar{\phi}_{j,\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k}\cdot\mathbf{r}}\phi_j(\mathbf{r})$. In bra-ket notation this expansion can be written as

$$|\psi_{n,\mathbf{k}}\rangle = \sum_{\mathbf{j}} c_{j,n,\mathbf{k}} |\bar{\phi}_{j,\mathbf{k}}\rangle. \tag{2.43}$$

Substituting this into the KS equations gives

$$\mathbf{H} \sum_{\mathbf{j}} c_{j,n,\mathbf{k}} |\bar{\phi}_{j,\mathbf{k}}\rangle = \varepsilon_{\mathrm{nk}} \sum_{\mathbf{j}} c_{j,n,\mathbf{k}} |\bar{\phi}_{j,\mathbf{k}}\rangle. \tag{2.44}$$

Now multiplying both sides by $\langle\bar{\phi}_{i,\mathbf{k}}|$ gives

$$\langle\bar{\phi}_{i,\mathbf{k}}|\mathbf{H} \sum_{j} a_{j,n,\mathbf{k}} |\bar{\phi}_{j,\mathbf{k}}\rangle = \varepsilon_{\mathrm{nk}} \langle\bar{\phi}_{i,\mathbf{k}}| \sum_{j} a_{j,n,\mathbf{k}} |\bar{\phi}_{j,\mathbf{k}}\rangle \tag{2.45}$$

$$\sum_{j} a_{j,n,\mathbf{k}} \langle\bar{\phi}_{i,\mathbf{k}}|\mathbf{H}|\bar{\phi}_{j,\mathbf{k}}\rangle = \varepsilon_{\mathrm{nk}} \sum_{j} a_{j,n,\mathbf{k}} \langle\bar{\phi}_{i,\mathbf{k}}|\bar{\phi}_{j,\mathbf{k}}\rangle. \tag{2.46}$$

$$\tag{2.47}$$

This can be viewed as the following generalized matrix eigenvalue problem

$$\mathbb{H}\mathbf{a} = \varepsilon \mathbb{S}\mathbf{a}, \tag{2.48}$$

where $\mathbb{H}$ is a matrix with the ith row and jth column given by $\langle\bar{\phi}_{i,\mathbf{k}}|\mathbf{H}|\bar{\phi}_{j,\mathbf{k}}\rangle$, the vector $\mathbf{a}$ contains the expansion coefficients, and $\mathbb{S}$ is the overlap matrix between basis function i and j, $\langle\bar{\phi}_{i,\mathbf{k}}|\bar{\phi}_{j,\mathbf{k}}\rangle$. Notice that for each choice of the vector $\mathbf{k}$, the Bloch functions are slightly different and therefore the matrices $\mathbb{H}$ and $\mathbb{S}$ are different also. For each value of $\mathbf{k}$ the solutions to equation (2.48) yield n eigensolutions.

The choice of basis function $\phi_j(\mathbf{r})$ depends on the situation under consideration. It is desirable to use a basis that can approximate the solution well with a small number of functions. For atoms and molecules, atomic-like orbitals are commonly used. For periodic, bulk solids a common choice is plane waves since they have an infinite extent. However, the nature of the electronic wavefunction varies from the interstitial region to the core. Electrons in the interstitial region vary very slowly, whereas in the core region the large ionic potential causes the wavefunction to oscillate rapidly. Because of these rapid oscillations, to approximate electronic states near the core requires the use of many planewaves. Since the energy

$$E = \frac{\hbar^2}{2m}|\mathbf{k}|^2 \tag{2.49}$$

can be associated with the wavefunction $e^{i\mathbf{k}\cdot\mathbf{r}}$, one way to specify the number of plane waves is to specify a cutoff energy. All plane waves with energy below the specified cutoff are used in the expansion when constructing equation (2.48). For the VASP software, the tag used to specify the cutoff energy is ENCUT. If a value is not specified, a default value, based on the depth of the ionic potentials is chosen.

2.5.2 Pseudopotentials

The dimensionality of the matrices $\mathbb{A}$ and $\mathbb{S}$ in equation (2.48) are determined by the number of basis functions included in the expansion of the single-electron wavefunctions. As these matrices get bigger, the computational cost of solving the problem increases. As was mentioned, the core electronic states (electrons who spend their time close to the nucleus) tend to oscillate rapidly due to the large electrostatic attraction to the nucleus. Approximating these states to a high degree of accuracy requires that many basis functions be included. However, most of the interesting and pertinent physical interactions in solids occur between valence states, with the core states remaining inert. With this in mind, one way to reduce the computational burden associated with solving equation (2.48) is to replace the true ionic potential, which diverges at the origin, with a "pseudo"-potential. The "pseudo"-potential approximates the ionic potential very accurately beyond some cutoff radius but replaces the diverging potential well near the origin with

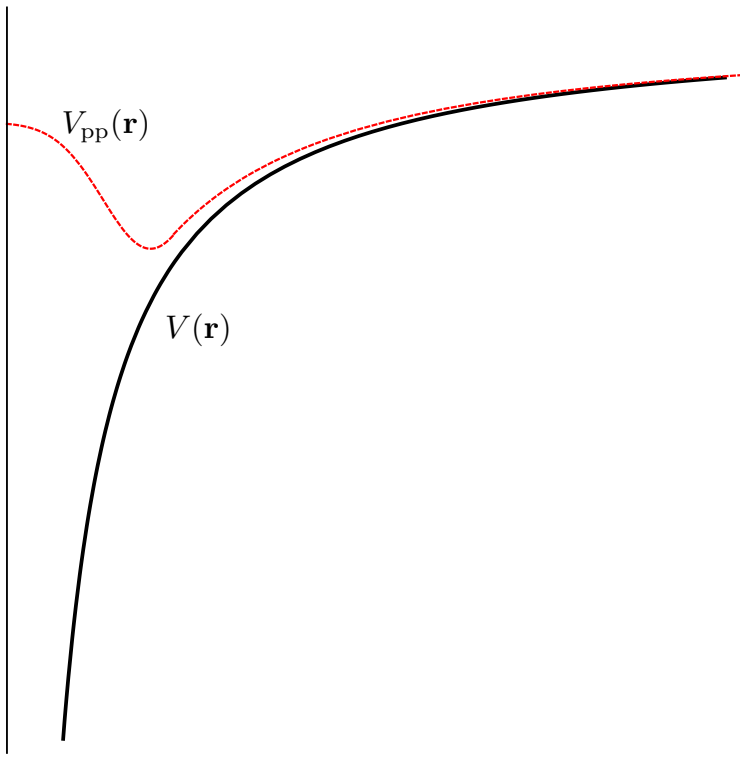


Figure 2.2: The pseudopotential (dashed, red line) is constructed to approximate the true potential (solid, black line) to a high degree of accuracy outside some cutoff radius. The justification for using such a pseudopotential is that core electrons do not contribute significantly to the chemical bonding in materials.

a more shallow one (see Figure 2.2). This allows the electronic wavefunctions to be accurately approximated with a relatively few number of plane waves.

Practically speaking, a pseudopotential is generated by considering an isolated atom of a single element. The resulting pseudopotential is transferrable, meaning it can be used without modification in situations where the atom is placed in a complex chemical environment. Pseudopotentials are classified according to how many planewave basis functions are needed to resolve the core states. Pseudopotentials with a very shallow core potential require few planewaves and are called soft, while pseudopotentials with deeper core potentials require more planewaves and are termed "hard". The most commonly used pseudopotentials are the ultrasoft pseudopotentials (USPP) of Vanderbilt [17]. In the VASP software, a default value for the variable ENCUT is included with each pseudopotential and is used when no value for ENCUT is explicitly provided.

2.5.3 Brillioun zone integration

The reciprocal vector **k** introduced in Bloch's theorem denotes the "crystal momentum" and is a *continuous* quantum variable with unique values being restricted to the unit cell in reciprocal

space. For each choice of this vector, a different matrix $\mathbb{A}$ and overlap matrix $\mathbb{S}$ can be formed and the solution to equation (2.48) yields a new set of one-electron states. Many important quantities involve integrating over all unique values of the quantum number $\mathbf{k}$

$$\int_{\mathrm{BZ}} g(\mathbf{k}) d\mathbf{k}, \tag{2.50}$$

where BZ indicates that the integral is the first Brillouin zone, or primitive unit cell in reciprocal space. For example, the electron density can be computed from the single-electron wavefunctions as

$$\sum_n \int_{\mathrm{BZ}} \left(f_{n,\mathbf{k}} \cdot |\psi_{n,\mathbf{k}}|^2 \right) d\mathbf{k}, \tag{2.51}$$

or the sum of occupied eigenvalues

$$\sum_n \int_{\mathrm{BZ}} \left(f_{n,\mathbf{k}} \cdot \varepsilon_{n,\mathbf{k}} \right) d\mathbf{k}. \tag{2.52}$$

Here $f_{n,\mathbf{k}}$ is the occupation number and the discrete sum over the band index is simply a sum over all eigensolutions to the equation (2.48). The integral, however, is over the entire reciprocal unit cell, and to approximate it we must employ numerical integration techniques. This requires that we define a set of reciprocal space points, or k-points at which to find solutions to the one-electron Schrödinger-like equations. At each chosen point, new matrices $\mathbb{A}$ and $\mathbb{S}$ are constructed and a set of solutions to the equation (2.48) are found. Numerical integration techniques are then used to interpolate between the chosen points and approximate the integral.

As these integrals define important physical quantities, great thought regarding the choice of k-points and the methods for numerically evaluating these integrals has been expended. Two of the most common methods used to construct the k-points grids are the Monkhorst-Pack [18] (named after Hendrick J. Monkhorst and James D. Pack) and the equivalent scheme suggested by Froyen [19]. The Monkhorst-Pack scheme subdivides each reciprocal lattice vector into a specified number of divisions, with the density of the resulting mesh being uniform. An example of Monkhorst-Pack k-points scheme is given in figure 2.3. The figure on the left shows a rectangular reciprocal unit cell whose reciprocal lattice vectors have been divided into 4 and 3 divisions. The figure on the right shows a hexagonal reciprocal unit cell whose reciprocal lattice vectors have been

divided into 4 divisions. In each case, the specific geometry of the reciprocal unit cell dictated the mesh chosen.

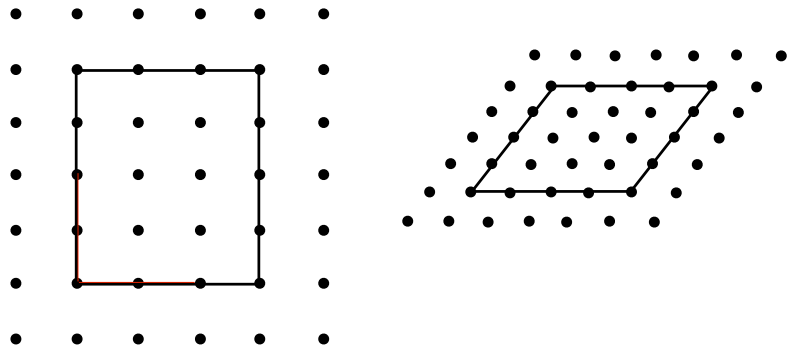


Figure 2.3: Illustration of the Monkhorst-Pack scheme for choosing k-points. Rectangular (left) and triangular (right) reciprocal unit cells are shown. For the rectangular unit cell, the k-points mesh is defined by dividing one lattice vector into 4 divisions and the other into 3, creating a mesh of uniform density. For the triangular unit cell, the mesh is defined by dividing both lattice vectors into 4 divisions. In each case the choice of division is dictated by the requirement that the density of the mesh be uniform.

The "equivalent" method was suggested by Froyen in cases where the comparison of two energy calculations is to be considered. For example, the formation enthalpy of a binary mixture of two elements is given by

$$H_{\text{formation}} = E_{\text{alloy}} - (E_{\text{A}} x_{\text{A}} + E_{\text{B}}(1 - x_{\text{A}})), \tag{2.53}$$

where E_{AB} is the energy per unit cell of the mixture configuration, N_{AB} is the number of atoms in the unit cell of the mixture, and x_A is the concentration of atom type A in the mixture. The formation enthalpy is a quantity of fundamental importance as it determines the energetic stability of a mixture. This calculation will require three first principles calcuations to be performed, and Froyen suggests that using the same mesh for all three calculations will result in a cancelation of systematic error and therefore a lower overall error in the formation enthalpy of the mixture. Under the "equivalent" scheme for generating k-point meshes a set of vectors in reciprocal space are first defined. The mesh is constructed by adding multiples of these vectors together. The chosen mesh

must be commensurate with the reciprocal unit cell. An illustration of the equivalent scheme for constructing k-points meshes is shown in figure 2.4.

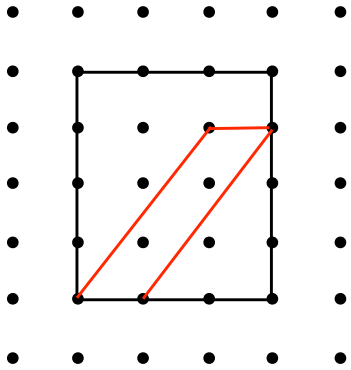


Figure 2.4: Illustration of Froyen's equivalent scheme for choosing k-points. The black dots indicate the k-points mesh with the black and red polygons being reciprocal unit cells commensurate with the mesh chosen. The mesh of k-points shown will be used for both reciprocal unit cells depicted here. Using the same k-points mesh is theorized to reduce systematic error.

In VASP, the file used to define the set of k-points is called KPOINTS. An example of a KPOINTS file defining a mesh of k-points under the equivalent scheme is given by:

```
Equivalent Kpoints   16 x 16 x 16
0
C
0.0625 0.0 0.0
0.0 0.0625 0.0
0.0 0.0 0.0625
.5 .5 .5
```

The first line is a comment, or title. The second line is the number of explicitly defined k-points being supplied, and is set to 0 here since we are not providing any explicitly defined points. The next line indicates the coordinate system that will be used to define the vectors. The next three lines are the vectors that will be used to construct the mesh of points. The final line is also a vector and is an offset or an amount that the mesh will be shifted. A KPOINTS file for the MP scheme is given by:

```
KPOINTS File  [KPPRA=6000]
0
Monkhorst-Pack
9 9 11
0 0 0
```

The only difference between this file and the previous one is the replacement of the three vectors with a single set of numbers. Each number indicates the number of divisions that the corresponding reciprocal lattice vector will be divided into. These three numbers are typically chosen to ensure that the mesh density is uniform.

Chapter 3

Cluster Expansion

3.1 Motivation for building a model

The DFT theorems represent a major stride in solid state theory, and DFT-based calculations are routinely used in computational materials research. However a single DFT calculation can easily occupy multiple computer processors for days or weeks depending on the complexity of the crystal structure, although most elemental crystal structures can be computed in a few minutes. This immediately precludes the use of DFT for performing large, exhaustive searches over many crystal structures. Furthermore, first-principles calculations focus on the zero temperature properties of a material and provide no insight into the stability or usefulness of a material at temperatures greater than zero Kelvin. These points illustrate that methods which use DFT only are unsuited to explore many materials problems of practical interest.

One way to extend the reach of computational methods to include these types of calculations is to use a handful of DFT data to construct a model. The model is typically much simpler mathematically than DFT and can therefore compute much faster. This speed enables large searches over many crystal structures to be performed. Thermodynamic simulations for assessing finite-temperature properties and which require millions of energy calculations, also become accessible once a fast, accurate model becomes available.

One class of materials problems of practical interest involves exploring all crystal structures whose atoms are constrained to lie on the sites of a parent lattice. Such materials are called derivative superstructures, and many experimentally observed crystal structures fall into this category. Crystal structure enumeration algorithms indicate that the number of unique atomic configurations increases exponentially with the size of the unit cell (see Figure 3.1), making it unfeasible to study such groups of crystal structures using computationally-costly DFT-based methods alone.

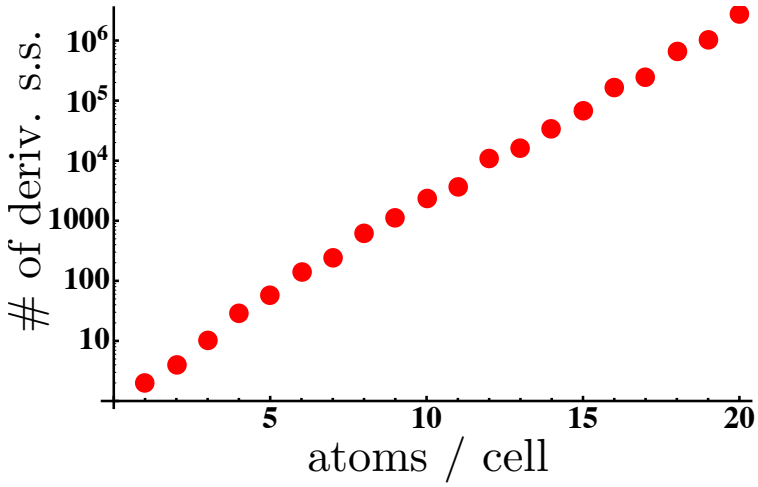


Figure 3.1: The number of unique, fcc-based derivative superstructures as a function of unit cell size, ranging from 1 atom/cell to 20 atoms/cell. Since DFT-based methods can be computationally costly, exhaustive exploration of derivative superstructures requires the use of a model.

One model that is commonly used to investigate substitutional order on a lattice (derivative superstructures) is the cluster expansion, which is an Ising-like model for atomic configurations restricted to a parent lattice. Similar to a Fourier, or Taylor series, the cluster expansion expresses a function as a linear combination of basis functions. Instead of taking a single number as an argument, the cluster expansion takes an atomic configuration and returns a number, typically a physical property of the configuration. The cluster expansion is different from these canonical expansions only in that it uses a different set of basis functions. (This seems appropriate and necessary when you consider that the domain of our function (all atomic configurations) is much more complex than the simple Cartesian space used in Fourier series.)

3.2 Definition of the basis

The mathematical formalism for the cluster expansion was developed by Sanchez, Ducastelle, and Gratias in 1984 [20] and is the foundation of modern-day cluster expansion methods. Other noteworthy mathematical work on the topic of cluster expansion methodology can be found in references [21] and [22]. The following is a review of the cluster expansion basis as put forth by Sanchez *et al.*

Begin with a set of N lattice points, each site being occupied by one of M possible atomic types. An occupation variable, σ_i, is then assigned to each lattice site, i, depending on the type of atom sitting there. The allowed values for the occupation variables are $\pm m, \pm(m-1), \ldots, \pm 1, 0$,

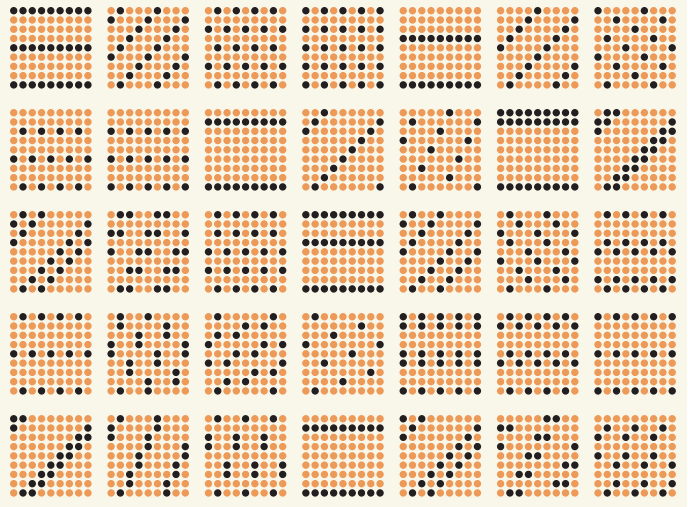


Figure 3.2: Thirty-five derivative superstructures derived from a square lattice. The atoms of each configuration lie on the sites of a square lattice and are unique configurations. Fcc, bcc, and hcp-derived superstructures are commonly observed in nature, which motivates their study in computational research.

where $m = \frac{M}{2}$ (or $\frac{M-1}{2}$). Any atomic configuration on the lattice may be fully specified from the vector of occupation variables, σ.

The scalar product between any two functions of σ, $f(\sigma)$ and $g(\sigma)$, is defined as

$$\langle f, g \rangle = \frac{1}{M^N} \sum_{\text{all configs.}} f(\sigma) \cdot g(\sigma), \tag{3.1}$$

where the sum is over all M^N configurations on the lattice. For an M-component system, an orthonormal basis with respect to this inner product can be constructed from M other functions, called point functions. These point functions take a single occupation variable as an argument (not the entire vector of occupation variables needed to specify the atomic configuration), and should also form an orthonormal set. The inner product for the point functions is very similar to equation 3.1, with the sum over all configurations being replaced with a sum over all allowed occupation variables.

$$\langle f(\sigma_p), g(\sigma_p) \rangle = \frac{1}{M} \sum_{\sigma_p=-m}^{m} f(\sigma_p) \cdot g(\sigma_p). \tag{3.2}$$

A logical choice for these functions is powers of the occupation variables: 1, σ_i, σ_1^2 .... Using the definition of the inner product to orthogonalize the first three (for example) powers of σ_i via Gram-Schmidt yields the following three point functions

$$\Theta_0(\sigma_i) = 1, \qquad \Theta_1(\sigma_i) = \sqrt{\frac{3}{2}}\sigma_i, \qquad \Theta_2(\sigma_i) = \sqrt{2} - \frac{3}{\sqrt{2}}\sigma_i^2. \tag{3.3}$$

Additionaly point functions can be added to this set by orthogonalizing over higher powers of σ_i. It can be easily verified that these point functions form an orthonormal set over the space of all possible occupation variables

$$\langle \Theta_n(\sigma_p), \Theta_{n'}(\sigma_p) \rangle = \frac{1}{M} \sum_{\sigma_p=-m}^{m} \Theta_n(\sigma_p)\Theta_{n'}(\sigma_p) = \delta_{nn'}. \tag{3.4}$$

An orthonormal set of functions, $\Pi_\alpha^{(s)}(\sigma)$, in the space of all M^N configurations on the lattice can now be constructed as products of point functions, $\Theta_n(\sigma_\alpha)$, for all possible combinations of the index n and of lattice points α. So for a cluster of lattice sites $\alpha = \{1, 2, \ldots |\alpha|\}$, and a vector of allowed point function indices, $s = \{n_1, n_2, \ldots n_l\}$ the basis functions are given by

$$\Pi_\alpha^{(s)}(\sigma) = \Theta_{n_1}(\sigma_1)\Theta_{n_2}(\sigma_2)\ldots\Theta_{n_l}(\sigma_\alpha). \tag{3.5}$$

Once again it can be easily shown that these functions form an orthonormal set

$$\langle \Pi_\alpha^{(s)}, \Pi_\beta^{(s')} \rangle = \delta_{\alpha\beta}\delta_{ss'}. \tag{3.6}$$

To better understand how these basis functions are constructed, let's consider the square lattice shown in figure 3.3. Remember that the functions $\Pi_\alpha^{(s)}$ are constructed by assembling products of point functions $\Theta_n(\sigma_\alpha)$ for **all possible indices** n **and all possible combinations of lattice sites** α. The combinations of indices determine which point functions are evaluated and the combinations of lattice sites determines on which lattice sites these point functions will be evaluated. So, for example, one possible combination of lattice sites is the two sites neighboring one another on the square lattice (see Figure 3.3). If only the zeroth and first point functions are considered then the possible point function products are:

$$\Pi_{nn}^{(0,1)}(\sigma) = \Theta_0(\sigma_1)\Theta_1(\sigma_2) = \sqrt{\frac{3}{2}}\sigma_2 \tag{3.7}$$

$$\Pi_{nn}^{(1,0)}(\sigma) = \Theta_1(\sigma_1)\Theta_0(\sigma_2) = \sqrt{\frac{3}{2}}\sigma_1 \tag{3.8}$$

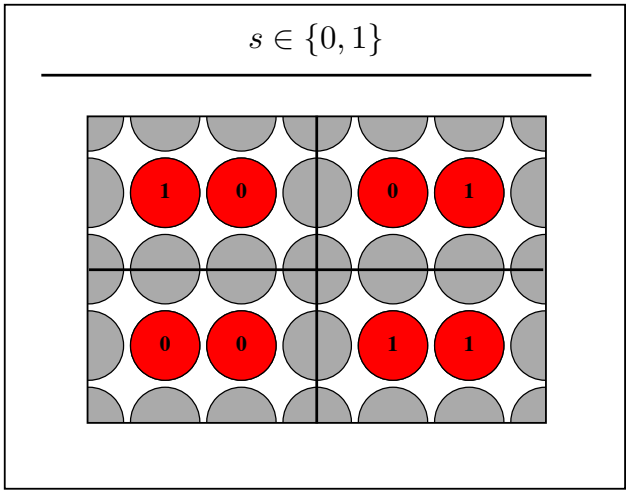


Figure 3.3: Illustration of all possible combinations of point functions on a nearest neighbor pair cluster on a square lattice. The (0,0) combination is always a constant regardless of atomic occupation. The $(0,1)$ and $(1,0)$ combinations are equivalent to the point cluster. The only unique pair cluster function here is the $(1,1)$ combination.

$$\Pi_{nn}^{(0,0)}(\sigma) = \Theta_0(\sigma_1)\Theta_0(\sigma_2) = 1 \tag{3.9}$$

$$\Pi_{nn}^{(1,1)}(\sigma) = \Theta_1(\sigma_1)\Theta_1(\sigma_2) = \frac{3}{2}\sigma_2\sigma_1 \tag{3.10}$$

In words, the basis function $\Pi_{nn}^{(0,1)}$, for example, indicates that this basis function is composed of the zeroth point function evaluated on the first site in the nearest neighbor pair cluster multiplied by the first point function evaluated on the second site in the nearest neighbor pair cluster (see figure 3.3).

As mentioned previously, for an M-component system, the first M point functions are needed to construct the basis. Theoretically, an infinite number of these functions can be constructed by identifying more symmetrically unique clusters of lattice sites, and enumerating all possible combinations of M point functions to be evaluated over those lattice sites. Figure 3.4 shows the first few unique clusters of lattice sites for the square lattice. As shown in the figure, the zero-body, or empty cluster which is not a physical cluster of lattice sites is formally included in the basis. A similar figure for clusters of lattice sites on an bcc lattice is given in figure 3.5. The assembling of all unique clusters of lattice sites when constructing the basis is the origin of the name cluster expansion. As the number of lattice sites in and spatial extent of a cluster increase the number of unique lattice-site clusters increases dramatically.

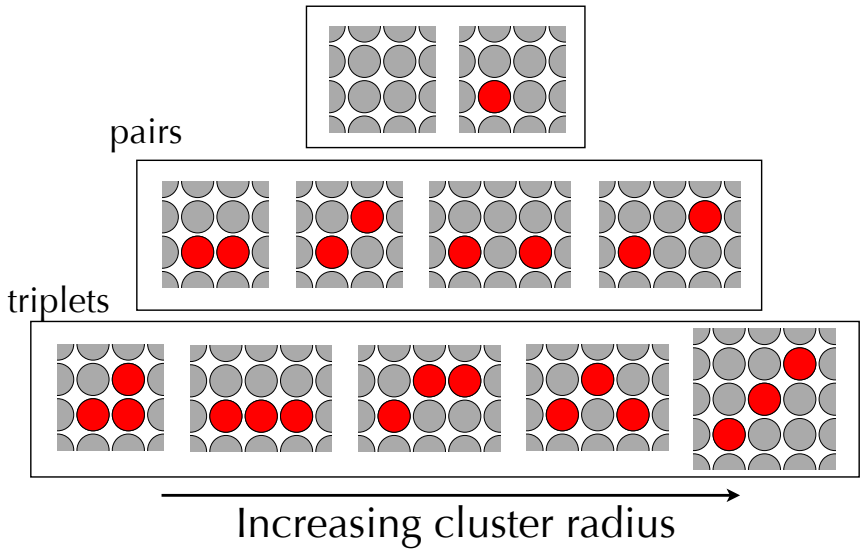


Figure 3.4: Illustration of geometrically unique clusters on a square lattice. The cluster basis is constructed from two components: (i) geometrically distinct clusters of lattice sites (depicted here) and (ii) all possible combinations of point functions on those lattice sites.

While the first M point functions are formally needed to describe an M-component system, the inclusion of the zeroth point function, $\Theta_0(\sigma_i) = 1$, always results in a redundant basis function. This can be seen in the example above by noticing that the basis functions $\Pi_{nn}^{(0,1)}(\sigma)$ and $\Pi_{nn}^{(1,0)}(\sigma)$ are equivalent to the point or on-site cluster $\Pi_{on-site}^{(1)}(\sigma)$ and the basis function $\Pi_{nn}^{(0,0)}(\sigma)$ is equivalent to the zero-body or "empty" cluster. For the example above, the function $\Pi_{nn}^{(1,1)}(\sigma)$ is the only unique basis function. In general, for an M-component system the point functions $\{\Theta_1, \Theta_2, \ldots, \Theta_{M-1}\}$ are needed to construct the basis. This means that for binary systems, only the first point function is needed, for a ternary system, the first and second point functions are needed, etc.

3.3 Using the basis

With the basis defined, any function of configuration σ can be expressed as a linear sum over these functions

$$E(\sigma) = \sum_a \sum_{(s)} J_a^{(s)} \bar{\Pi}_a^{(s)}(\sigma). \tag{3.11}$$

The $\bar{\Pi}_a^{(s)}$ are averages over symmetrically equivalent versions of the $\Pi_a^{(s)}$ functions (see figure 3.6) and over all unique sites in the crystal, all sites inside the unit cell (see figure 3.7). The $J_a^{(s)}$ are

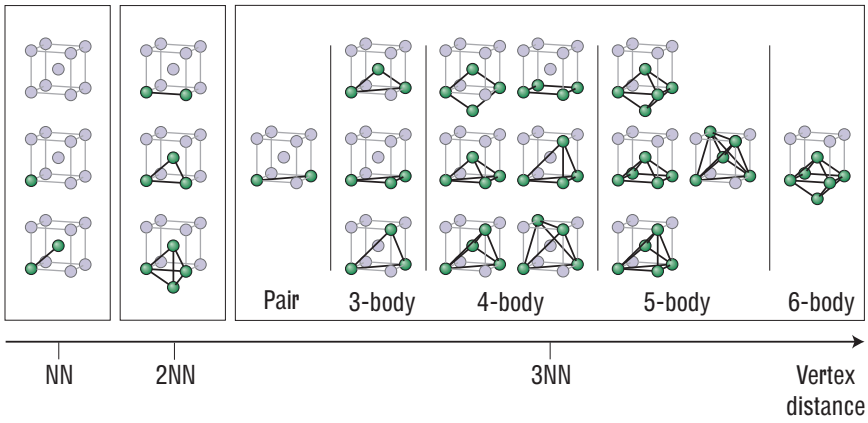


Figure 3.5: Geometrically unique clusters of fcc lattice sites. The average distance from the center of mass of the cluster increases moving left to right.

the coefficients associated with the clusters, and determining their values is the main goal when constructing the model.

Constructing a cluster expansion is essentially a linear algebra problem

$$\bar{\Pi}\mathbf{J} = \mathbf{E}, \tag{3.12}$$

where the matrix $\bar{\Pi}$ is composed of the cluster (basis) functions, $\bar{\Pi}_a^{(s)}(\sigma)$, evaluated at the crystal structures chosen as training data. Each row in this matrix corresponds to a crystal structure and each column corresponds to a cluster function. The vector $\mathbf{E}$ contains the first-principles energies of the crystal structures used for training data and the vector $\mathbf{J}$ contains the sought-after model coefficients, sometimes referred to as ECIs or effective cluster interactions.

Since the number of possibly-relevant basis functions is much larger than the number of DFT data points that is feasible to compute for a single system, the problem of constructing a cluster expansion naturally emerges as an underdetermined problem. Without knowing how to constrain the solution search, solving an underdetermined problem is hard because there are an infinite number of solutions that are consistent with the data provided. For this reason the most popular techniques for constructing cluster expansion models enforce that the number of data points (rows in matrix $\bar{\Pi}$) be greater than or equal to the number of basis functions considered (columns in matrix $\bar{\Pi}$). This creates a determined or overdetermined system which can be solved by inverting the matrix $\bar{\Pi}$ [23] or through standard linear algebra techniques such as singular value decomposition, etc.

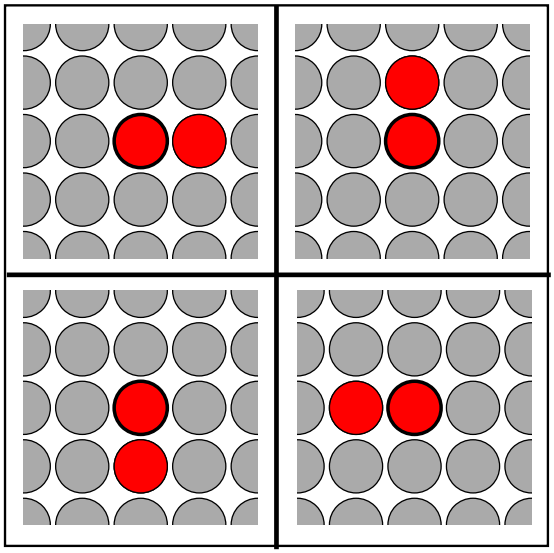


Figure 3.6: Symmetrically equivalent versions of the nearest neighbor pair cluster on a square lattice. When constructing the cluster functions, the point function products are averaged over all symmetrically equivalent versions of the cluster. In this example there would be four terms in the average.

Restricting the matrix $\bar{\Pi}$ to not have more columns (basis functions) than rows (DFT data points) makes the efficient construction of a robust, predictive model very challenging. While the number of relevant basis functions is probably not large, knowing *a priori* which, of the thousands of candidates, should be included is not possible. Inevitably, the chosen truncation will include irrelevant terms and exclude relevant ones, and the predictive quality of the model will suffer as a result. Many approaches to solving this problem have been invented, and most modern methods typically involve using physical intuition and/or complex algorithms to truncate the expansion. Here we give a short review of modern techniques for truncating the expansion.

3.4 Review of current techniques

A popular method for truncating the expansion involves monitoring the predictive capacity of the model as basis functions are added/removed from the expansion. The method begins with an initial set of DFT data and a pool of candidate basis functions. Basis functions are added to the expansion only if their inclusion increases the predictive capability of the model. The algorithm terminates when none of the attempted changes produce an improvement in the model. This algorithm is *NP*-hard and is referred to as direct optimization (DO) because it seeks to optimize the quality of the model directly by trying every possible solution (or at least all those solutions that can be explored in a reasonable amount of time). Unfortunately, due to the enormity of the search space, there is no guarantee that this process will truly result in the optimal set of ECI's.

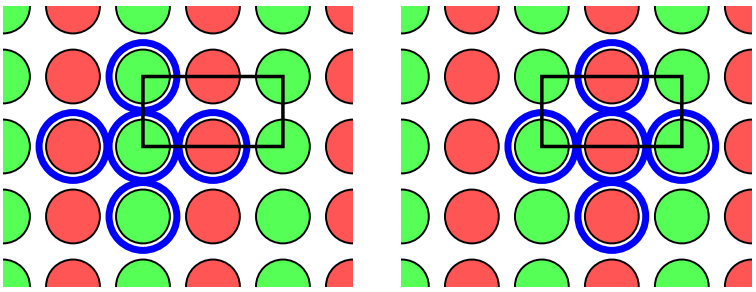


Figure 3.7: Illustration of how the basis function corresponding to the nearest neighbor pair cluster is averaged over all unique sites in the crystal. The atomic configuration shown has two unique sites and the unit cell is given by the rectangle. In the figure on the left, all rotationally equivalent versions of the pair cluster are constructed (see figure 3.6). In the figure on the right, the sites of the pair cluster are translated to the other unique lattice site.

As a result DO may miss important clusters or add clusters that should not have been included in the model, which may result in non-physical ECI values. Additionally, adding/removing clusters one-by-one is computationally expensive, requiring days to finish when considering very large pools of candidate clusters.

At each iteration of the DO procedure, the quality of the current model must be assessed. This is typically done using k-fold cross validation [24], a method devised to avoid having to construct a costly holdout set of data for validation. K-fold cross validation is done by first dividing the data set into k subsets. One at a time, each of the subsets is withheld from the fitting procedure and saved to validate the model. The reduced data set is then used in the fitting procedure and the resulting model is used to predict over the validation set. The root-mean-square error (rmse) of the predictions made on the validation set is then computed. This is done k times, each time using a different set for validation and computing the rmse of the validation set. The final fit-quality measure is then computed as the average of the rmse values.

Genetic algorithms have been used with some success to select relevant clusters [25, 26]. In the genetic algorithm paradigm, the poor-quality model evolves toward a high-quality model through a series of matings and mutations of the solution. In this approach each cluster in the pool is thought of as a single gene in a biological genome. Each gene can take on a value of 1 or 0, where 1 indicates that the corresponding cluster is "on" or is being used in the expansion and a 0 indicates that it is being excluded ("off"). The algorithm proceeds by constructing a population, or pool of genomes, and then mating genomes amongst each other to create children

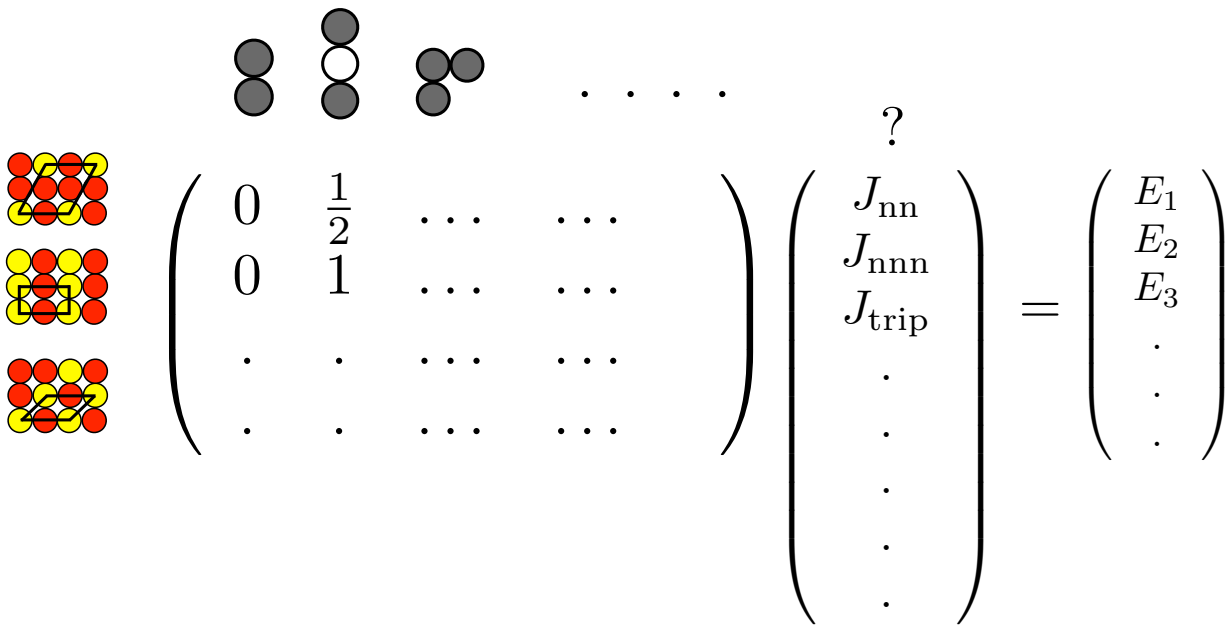


Figure 3.8: Illustration of the linear algebra problem that emerges when constructing a cluster expansion. Most notably, the problem is heavily underdetermined due to the vast number of possibly-relevant basis functions.

genomes. The "on" genes, or clusters, are then used when fitting to the data, and a cross validation scheme is used to assign a quality to each genome. This approach has had some success but the time required to run increases rapidly with the size of the cluster pool and the number of fitting structures. Additionally, this method utilizes various user-tuned parameters, further complicating the model-building process by requiring intensive human time.

The DO and GA methods are two commonly-used ways to optimize the cross validation score, one measure of the quality of the model. Other methods for optimizing the CV score have also been suggested [27, 28], and they perform reasonably well. However, all of these methods are limited in the fact that the number of basis functions that can be considered must be less than or equal to the number of data points available. This amounts to a heavy truncation, with many possibly-relevant clusters being left out of consideration. While the number of relevant clusters probably does not exceed the number of data points typically available, the probability that the chosen truncation will include all relevant terms is very low.

Bayesian-statistics-based methods have been used to estimate ECI's and have shown to outperform several common methods in low-symmetry situations [29,30]. However, these methods

require the incorporation of detailed physical intuition about the relative strength of the Js, adding laborious, time-consuming, and system-specific requirements to the model construction process.

The second noteworthy challenge in the cluster expansion construction process is choosing which structures to use as training data. Since the information content of a single crystal structure varies with the truncation choice, this problem is coupled with the cluster selection problem. Most modern efforts in cluster expansion theory have focused on the truncation problem, with little thought regarding the choice of training structures. However, several researchers have proposed structure selection methods aimed at minimizing the variance in the fit energies [31, 32]. These methods typically involve iterative procedures where a fit is constructed (using costly DO or GA algorithms) and then a set of structures are selected to be added to the training set. First-principles calculations for the chosen structures must then be performed. This process continues until the variance is minimized to some threshold.

With the exception of recent CE techniques based on Bayesian inference [1, 29, 30], the model-building process of contemporary techniques are essentially the same: An initial set of training data is generated and a fit is calculated. The predictive accuracy of the model is assessed, and more training data is generated and added to the set of training data. This results in a more refined, and typically more complex, model. This process is continued, with more and more terms being included in the expansion, until a model with the desired predictive accuracy is achieved.

3.5 Conclusion

The cluster expansion model is a useful tool for exploring many material problems of practical interest, such as exhaustive searches over many candidate crystal structures and thermodynamic simulations used for assessing finite-temperature stability and usefulness. The mathematical foundation of the CE basis was proven by Sanchez *et. al.*, and has been reviewed here. The basis is constructed by first constructing a set of point functions that take a single occupation variable as an arguement, and then forming products of these point functions over geometrically unique clusters of lattice sites.

To date, the process of constructing a CE model has been arduous at best, requiring parameter tuning, lengthy iterative procedures, and complex algorithms. The most widely-used methods for finding relevant clusters have involved direct optimization procedures or genetic algorithms.

Both techniques fall short of providing an efficient, robust, and automatic method for constructing these models.

Chapter 4

Compressive Sensing

4.1 Introduction

Physical intuition and experience suggest that many important properties of materials are primarily determined by just a few key variables. For instance, the crystal structures of intermetallic compounds have been successfully classified into groups (so-called structure maps) according to the properties of the constituent atoms. [33–36] The widely known Miedema rules relate alloy formation energies to atomic charge densities and electronegativities. [37] Most magnets can be described using a Heisenberg model with only a few short-ranged exchange interactions, [38] and the formation energies of multicomponent alloys can be efficiently parameterized using generalized Ising models (cluster expansions) with a finite number of pair and multibody interactions. [20–22] In all these cases, enormous gains in efficiency and conceptual clarity are achieved by building models which express the quantity of interest (typically, total energy) in a simple, easy-to-evaluate functional form. These models can then be used to perform realistic simulations at finite temperatures, on large systems, and/or over long time scales, significantly extending the reach of current state-of-the-art quantum mechanics based methods.

The conventional approach to model building starts by selecting a small, physically motivated basis set which describes the configuration of the system. The target properties are then expressed in terms of these basis functions and the unknown coefficients are determined by performing least-squares fits to the calculated or experimentally measured data. While conceptually simple, this method is often difficult to use in practice. First, the number of unknown coefficients has to be smaller than the number of data points, which precludes the use of very large basis sets. Second, least-squares fitting is susceptible to noise, and there is often the possibility of "overfitting"—the model is trained to reproduce the fitting data, but performs poorly in a predictive capacity. Finally, finding the the optimal finite basis set is an *NP*-hard problem, i.e., the solution time

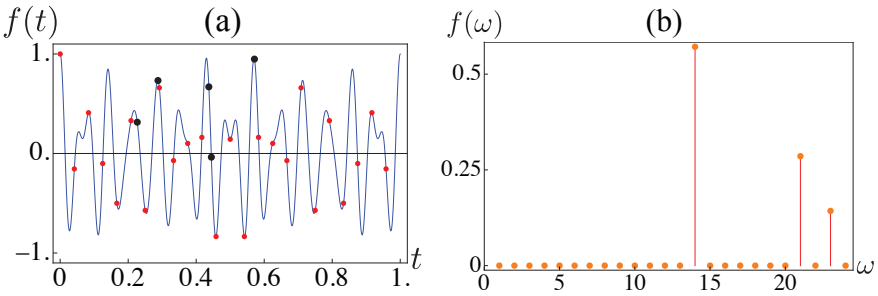


Figure 4.1: (a) A sparse signal (blue line) like that of Eq. 4.2, uniform samples of the signal at the Nyquist frequency (red dots), and a few random samples (black circles). The signal is composed of only 3 non-zero frequencies. (b) Exact recovery of the frequency components of the signal using compressive sensing.

increases faster than polynomial with the number of possible basis functions. To keep the number of coefficients smaller than the amount of data, one must choose, based on physical intuition, which basis functions to keep. This physical intuition in many cases may be unavailable and/or difficult to develop; hence there is no clear path to achieve systematic improvement. Recent years have seen numerous attempts to use machine learning algorithms (genetic programming, neural networks, Bayesian methods, etc.) to decrease the role of intuition in model-building. [29, 30, 39–44]

We show that a recently developed technique in the field of signal processing, compressive sensing (CS), [45] provides a simple, general, and efficient approach to model-building. [46] Instead of attempting to develop physical intuition for which coefficients will be most relevant, the CS framework allows the inclusion of essentially all possible basis functions. Using very large basis sets eliminates the need to use physical intuition to construct smaller ones. Furthermore, CS is computationally efficient for very large problems, robust even for very noisy data, and its models predict more accurately than current state-of-the-art approaches.

4.2 Compressive sensing: an illustration

Before demonstrating the power of compressive sensing for building physical models, we first illustrate the concept itself with a simple time series. Discussion of compressive sensing

requires the definition of ℓ_p norms:

$$\|u\|_p = \left(\sum_i |u_i|^p \right)^{1/p}, \tag{4.1}$$

of which the ℓ_1 (taxicab or Manhattan distance) and ℓ_2 (Euclidean; subscript 2 often omitted) norms are special cases. The number of non-zero elements of $\vec{u}$ is often (improperly) referred to as the ℓ_0 "norm" even though it is not a norm in a strict mathematical sense.

In the signal processing community, compressive sensing is used to recover sparse signals *exactly* with far fewer samples than required by standard spectral techniques, such as the well-known Fourier and Laplace transforms. Consider a signal like that shown in Fig. 4.1(a) which has the functional form:

$$f(t) = \sum_{n=1}^{N} u_n e^{i2\pi nt}, \tag{4.2}$$

where most of the coefficients, u_n, are zero (i.e., the signal is sparse). The Fourier transform is mathematically equivalent to solving the matrix equation

$$\mathbb{A}\vec{u} = \vec{f}, \tag{4.3}$$

where the matrix $\mathbb{A}$ is formed by the values of the Fourier basis functions at the sampling times t_m, i.e., it consists of rows of n terms of the form $A_{mn} = e^{i2\pi nt_m}$, and $f_m \equiv f(t_m)$ is the sampled signal. The solution vector $\vec{u}$ contains the relative amounts of the different Fourier components, as shown in Fig. 4.1(b). Capturing all relevant frequency components of the signal using Fourier transform techniques requires the signal to be sampled regularly and at a frequency at least as high as the Nyquist frequency [shown as red points in Fig. 4.1(a)], a severe restriction stemming from the requirement that the linear system Eq. (4.3) should not be underdetermined.

However, the main idea of compressive sensing is that,when the signal is sparse, one should be able to recover the exact signal with a number of measurements that is proportional to the number of nonzero components, i.e., with far fewer samples than given by the Nyquist frequency. Conceptually, this could be done by searching for a solution that reproduces the measured time signal exactly *and* has the minimum number of non-zero Fourier components. Unfortunately, this formulation results in a discrete optimization problem, which cannot be solved in polynomial time.

Compressive sensing recasts the problem as a simple minimization of the ℓ_1 norm of the solution, subject to the constraint given by Eq. (4.3) above:

$$\min_u \{\|\vec{u}\|_1 : \mathbb{A}\vec{u} = \vec{f}\}, \tag{4.4}$$

where $\|\vec{u}\|_1 = \sum_i |u_i|$ is the ℓ_1-norm defined in Eq. (4.1). In other words, one seeks to minimize the sum of the components of the solution vector $\vec{u}$ subject to the condition that the measured signal is reproduced exactly; this constitutes the so-called basis pursuit problem. Eq. (4.4) is a convex optimization problem which can be solved efficiently (see Sec. 4.4). We note here that optimization of the common sum-of-squares (ℓ_2) norm of $\vec{u}$ would result in a dense solution which may deviate considerably from the original signal. [45]

As a simple illustration, the exact decomposition of an example function, shown in Fig. 4.1, was possible via compressive sensing with only 5 random samples (black dots in figure 4.1) of the signal, instead of the 24 equally-spaced samples (red dots in figure 4.1) needed for a discrete Fourier transform. Quite generally, a mathematical theorem proven by Candes, Romberg, and Tao [47] guarantees that, with an overwhelming probability, any sparse signal with S nonzero components can be recovered from $M \sim S \log N$ random measurements, where N is the total number of sensing basis functions. This very powerful result is the mathematical foundation of compressive sensing.

Another practically important feature of compressive sensing is the ability to tolerate noise in the input data and to deal with signals that are only approximately sparse, i.e., are dominated by a few large terms, but also contain a large number of smaller contributions; this is the case in almost all physics applications. It has been proven that, if the sensing matrix $\mathbb{A}$ obeys the so-called *restricted isometry property* (RIP), an accurate reconstruction of the signal from highly under-sampled measurements can be achieved also in the presence of both random and systematic noise. [45,47] The RIP criterion is automatically satisfied if the measurements are chosen randomly. (see Sec. 4.4.4 for a detailed discussion)

When applying compressive sensing to model building, two tasks must be accomplished: (i) a basis must be chosen, and (ii) the coefficients associated with each basis function must be determined. Mathematically, the problem is analogous to the simple Fourier example considered

above, with the sensing matrix $\mathbb{A}$ being determined by the values of the basis functions at the chosen measurement points. Below we illustrate the use of compressive sensing on two cluster expansion (CE) models of configurational energetics: [20] (i) Ag-Pt alloys on a face-centered cubic (fcc) lattice, and (ii) protein folding energies in the so-called zinc finger motif. CE is chosen as an example because it is conceptually simple, mathematically rigorous, and widely used in the materials community to calculate temperature-composition phase diagrams. Furthermore, CE is a stringent test case for compressive sensing because a significant amount of effort has been expended developing advanced model building techniques, which have been implemented in sophisticated general-purpose computer codes. [24, 26, 29, 30, 44, 48–50]

4.3 Cluster Expansion

4.3.1 Energy Model

Since a formal mathematical description of CE can be found in the literature, here we only restate its main features and refer the reader to Refs. 20–22 for detailed explanations. The CE method uses a complete set of discrete basis functions, defined over clusters of lattice sites, which describe the occupation of each site and thus the entire atomic configuration on the crystal. The total energy is given by

$$E(\sigma) = E_0 + \sum_f \bar{\Pi}_f(\sigma) J_f, \tag{4.5}$$

where f represents symmetrically distinct clusters of lattice sites (points, pairs, triplets, etc.), σ denotes the atomic configuration, usually expressed by a collection of pseudo-spin variables $\{S_i\}$ describing the type of atom at each lattice site, and the cluster correlations $\bar{\Pi}_f(\sigma)$ are formed as symmetrized averages of products of these pseudo-spin variables. The key quantities in this approach are J_f, the effective cluster interactions (ECI's): Given the ECI's, the energy of *any* atomic configuration on the lattice can be calculated rapidly from Eq. (6.1). Physical intuition based on the concept of "near-sightedness"of screened interatomic interactions suggests that only clusters within a limited range and involving a limited number of sites will have significant ECI's. The goal then is to determine which of the clusters f, out of the myriads of possible choices, contribute significantly to the total energy of the system and to calculate the values of these coefficients.

Currently, the most popular approaches are based on the so-called structure inversion method (SIM), [23] where a limited number of quantum-mechanics-based total energy calculations are used to determine $E(\sigma)$ on the left-hand side of Eq. (6.1). The cluster interactions J_f are truncated according to some recipe and their values are determined by least-squares fitting to the training set energies $E(\sigma)$. The accuracy of the resulting CE depends crucially on the chosen truncation method. Including too few interactions leads to poor predictive power because important interactions are not accounted for ("under-fitting"), while choosing too many parameters J_f results in spurious interactions and an associated decrease in predictive accuracy ("over-fitting"). Use of least-squares fitting necessarily requires that the number of structures must exceed the number of candidate ECIs, which is the CE analogue of the Nyquist frequency in signal processing.

In modern practice, the trial ECI's are chosen by scanning over many possible sets of clusters while attempting to minimize the predictive error. Ideally, the predictive error should be calculated as the root mean square (RMS) deviation between the density functional theory (DFT) and CE-predicted energies over a separate "hold-out" set of structures that are not used in fitting. This approach would require tens or hundreds of additional DFT calculations and is therefore seldom used in practice. Leave-one-out cross-validation (LOOCV) or k-fold cross-validation scores are commonly used as proxies for the predictive error since they do not require the construction of a separate hold-out set. [24]

Starting from an initial set of ECI's (e.g., empty, point, and nearest-neighbor pair clusters), a typical procedure for improving the model attempts to add and/or substitute clusters into the current set, keeping changes if the predictive error is found to decrease. The procedure is terminated when none of the attempted changes produce an improvement in the predictive accuracy. Unfortunately, there is no guarantee that this process will truly result in the optimal set of ECI's because it is practically impossible to solve the *NP*-hard discrete optimization (DO) problem, especially if the number of candidate ECI's is large, such as required for very accurate CE's or in situations of low symmetry (e.g., near defects, surfaces, nano-particles). As a result, with DO, one may miss important clusters or add clusters that should not have been included in the model, which may result in non-physical ECI values. Additionally, adding/removing clusters one-by-one is computationally expensive, requiring days to finish when considering very large pools of candidate clusters.

Genetic algorithms have been used with some success, but they also require large amounts of time to complete, especially for large cluster pools and fitting sets, and employ a host of tunable paramters. [25,26] Other methods for optimizing the CV score have been proposed [27,28] but are limited in the number of basis functions they can consider and require a heavy initial truncation of the basis.

Other researchers, in an attempt to avoid predictive errors associated with incomplete discrete optimization, have sought to devise direct minimization methods that automatically select ECIs only if they are required to reproduce the energies of the training set. The first such approach was proposed by Laks, Wei, and Zunger for pair interactions, [51] who added a distance-weighted ℓ_2 norm of the pair interactions to the objective function. However, this approach usually results in dense sets of long-ranged pair ECI's and, more importantly, is difficult to extend to multibody interactions. [52–54] Recently, a method based on Bayesian statistics was introduced to automatically estimate ECI's and shown to outperform several common methods in low-symmetry situations. [29] However, it makes use of physical intuition to construct informative prior distributions, which are required for estimating the ECI values. It is desirable to develop methods that avoid the use of intuition since heuristic rules, derived from experience with a few specific systems, may not be universally valid. Design of efficient, numerically robust and physically accurate methods for selecting the physically significant ECIs remains a challenging problem.

4.3.2 Compressive sensing cluster expansion (CSCE)

Here, we show that compressive sensing can be used to select the important ECI's and determine their values *in one shot*. The applicability of compressive sensing to CE is based on the mathematical theorem of Candes, Romberg, and Tao, [47] which guarantees that sparse ECI's can be recovered from a limited number of DFT formation energies given certain easy-to-satisfy properties of the matrix $\bar{\Pi}$ in Eq. (6.1). Adopting the common assumption that the "true" physical ECI's are approximately sparse, this theorem guarantees that a good approximation will be found even in cases when the data has both random and systematic noise, e.g., due to numerical errors in the DFT calculations or due to interactions beyond the chosen energy resolution, see Sec. 4.4.5.

There are two possible formulations for compressive sensing cluster expansion (CSCE), both of which enforce the requirement that the cluster expansion should be as sparse as possible,

while resulting in a certain level of accuracy for the training set. In the first approach, one may determine the optimal set of ECI's from

$$J = \arg\min_{J} \left\{ ||J||_1 : ||E - \bar{\Pi} J|| \leq \varepsilon \right\}, \tag{4.6}$$

where the ℓ_1 norm of J's is used as a proxy for the number of nonzero ECI's. Solving the so-called LASSO problem Eq. (4.6) [55, 56] offers a mathematically strict way of constructing a minimal cluster set that reproduces the training set with a given accuracy. Of course, over- (under-) fitting is still an issue if ε is chosen too small (large), but it is physically reasonable that, given the physical properties of the system and the size of the training set, an optimal ε always exists. Following common practice, optimal ε could be found either by minimizing the LOOCV score or the predictive RMS error over a hold-out set.

Since the inequality constraint is inconvenient to enforce during calculations, [55] here we follow common practice in signal processing and use an unconstrained approach which minimizes the sum of an ℓ_1 norm of the ECI's and a least-squares sum of the fitting errors:

$$J = \arg\min_{J} \mu ||J||_1 + \frac{1}{2} ||E - \bar{\Pi} J||^2, \tag{4.7}$$

where μ is a parameter that controls the accuracy of the fit versus the sparseness of the solution: higher values of μ will result in sparser solutions and larger fitting errors (under-fitting), while very small μ values will lead to dense solutions and degraded predictive accuracy (over-fitting). It will be shown below in Sec. 4.4.5 that the optimal value of μ is proportional to the level of noise (random and systematic) in the calculated formation energies. Just like ε in Eq. (4.6), an optimal μ to avoid over- or under-fitting can be chosen either by minimizing the LOOCV score or by minimizing the rms prediction error for a separate hold-out set; it is shown below that both approaches result in very similar values of optimal μ. Furthermore, in Sec. 4.5 we demonstrate that CSCE is not particularly sensitive to the precise value of μ and show that there is usually a range of μ's that give ECI's of similar predictive accuracy.

The main advantage of CSCE, Eqs. (4.6) & (4.7), over current CE methods is that the *NP*-hard discrete optimization of the truncated ECI set is replaced by convex optimization problems for which exact solutions may be found in polynomial time. Furthermore, the minimization of the

ℓ_1 norm of the solution also serves to decrease the magnitude of the ECI's, leading to "smoother" ECI's, increased numerical stability with respect to the noise in the training data, and eventually more accurate predictions. In addition, the CSCE is simple to implement and use, which will facilitate its widespread adoption in solid state physics and other fields where configurational energetics play a role. In Sec. 4.5 below, we illustrate the superior performance of CSCE using examples from bulk alloys (Ag-Pt) and biology (protein folding energetics).

4.4 Practical aspects of ℓ_1-based optimization

In what follows, we review methods for solving the unconstrained minimization problem given by Eq. (4.7), which we rewrite as:

$$\min_{u} \mu \|\vec{u}\|_1 + \tfrac{1}{2}\|\mathbb{A}\vec{u} - \vec{f}\|^2. \tag{4.8}$$

Eq. (4.8) is referred to as the basis pursuit denoising problem. It has a tunable parameter, μ, which controls the sparseness of the solution: smaller (larger) values of μ produce less (more) sparse solutions.

4.4.1 Fixed-point continuation

The fixed-point continuation (FPC) method of Hale, Yin, and Zhang [57] is an iterative algorithm that starts from $\vec{u}^0 = \mathbf{0}$ and attempts to improve the objective function by following the gradient of the ℓ_2 term:

$$\vec{g}^k = \mathbb{A}^T(\mathbb{A}\vec{u}^k - \vec{f}) \tag{4.9}$$

$$u_n^{k+1} = \text{shrink}\left(u_n^k - \tau g_n^k, \mu\tau\right) \tag{4.10}$$

where $k = 0, 1, 2, \ldots$ is the iteration number and the shrinkage operator is defined as

$$\text{shrink}\,(y, \alpha) := \text{sign}(y)\max\left(|y| - \alpha, 0\right). \tag{4.11}$$

In other words, shrinkage decreases the absolute magnitude of y by α and sets y to zero if $|y| \leq \alpha$. The iterations are stopped when the ℓ_∞ norm, or maximum component value, of the gradient drops

below the shrinkage threshold,

$$\frac{1}{\mu}||\vec{g}||_\infty - 1 < \delta_g, \tag{4.12}$$

and the change in the solution vector is sufficiently small,

$$\frac{||\vec{u}^{k+1} - \vec{u}^k||}{||\vec{u}^k||} < \delta_u. \tag{4.13}$$

The sensing matrix should be normalized in such a way that the largest eigenvalue α_A of $\mathbb{A}^T\mathbb{A}$ is less than or equal to 1; this is easily accomplished by dividing both $\mathbb{A}$ and $\vec{f}$ by $\sqrt{\alpha_A}$. The step size τ in Eq. (4.10) is given by

$$\tau = \min(1.999, -1.665\frac{M}{N} + 2.665), \tag{4.14}$$

where M and N are the number of equations and the number of expansion coefficients, respectively.

4.4.2 Bregman iteration

While the FPC algorithm is generally applicable to any problem of type Eq. (4.8) and is guaranteed to converge, in practice it has a serious shortcoming: very small values of μ are needed to recover the exact solution to the basis pursuit problem without noise, Eq. (4.4), which cause an associated increase in the number of FPC iterations. To alleviate the need to use small μ's, Yin *et al.* [58] proposed an efficient iterative denoising algorithm for finding the solution to Eq. (4.8), which has the additional benefit of yielding the exact solution to the basis pursuit problem Eq. (4.4) for zero noise. This so-called Bregman iteration involves the following two-step cycle:

$$\vec{f}^{k+1} = \vec{f} + (\vec{f}^k - \mathbb{A}\vec{u}^k), \tag{4.15}$$

$$\vec{u}^{k+1} = \arg\min_u \mu\|\vec{u}\|_1 + \tfrac{1}{2}\|\mathbb{A}\vec{u} - \vec{f}^{k+1}\|^2, \tag{4.16}$$

starting from $\vec{f}^0 = \mathbf{0}$ and $\vec{u}^0 = \mathbf{0}$. A key feature of the algorithm is that the residual after iteration k is added back to the residual vector $\vec{f}^{k+1}$ for the next iteration, resulting in efficient denoising and rapid convergence. [58] Each minimization in Eq. (4.16) can be performed using the fixed-point continuation (FPC) method proposed by Hale, Yin, and Zhang. [57] The main advantages of the

Bregman iteration are faster convergence and the ability to use μ values that are several orders of magnitude larger than those required for direct application of the FPC method.

4.4.3 Split Bregman iteration

For very large problems (i.e., large sensing matrices $\mathbb{A}$), the FPC optimization steps in the Bregman iterative method progress very slowly. The problem becomes severe when the condition number computed from the nonzero eigenvalues of $\mathbb{A}^T\mathbb{A}$ becomes large. Indeed, FPC is essentially a steepest descent method combined with an ℓ_1 shrinkage step, and the number of required steepest descent iterations increases linearly with the ratio of the largest-to-smallest eigenvalues of $\mathbb{A}^T\mathbb{A}$. [59] An improved Bregman algorithm, which eliminates the hard-to-solve mixed ℓ_1 and ℓ_2 minimization problem in Eq. (4.16), was proposed by Goldstein and Osher. [60] It carries the name of "split Bregman" iteration because it splits off the ℓ_1 norm of the solution from the objective function and replaces it with the variable $\vec{d}$, which is designed to converge towards the ℓ_1 term, $\lim_{k\to\infty}(\vec{d}^k - \mu\vec{u}^k) = \mathbf{0}$. A least-squares ℓ_2 term is added to the objective function to ensure that $\vec{d} = \mu\vec{u}$ in the limit:

$$\vec{u} = \arg\min_{u,d} \|\vec{d}\|_1 + \tfrac{1}{2}\|\mathbb{A}\vec{u} - \vec{f}\|^2 + \tfrac{\lambda}{2}\|\vec{d} - \mu\vec{u}\|^2. \tag{4.17}$$

A key advantage of this formulation is that the minimization involving the quadratic form $\|\mathbb{A}\vec{u} - \vec{f}\|^2$ does not contain ℓ_1 terms and can be performed efficiently using standard convex optimization techniques, such as Gauss-Seidel or conjugate gradients (CG), [59] while the ℓ_1 minimization with respect to $\vec{d}$ at a fixed $\vec{u}$ contains an ℓ_2 term that is diagonal in the components of $\vec{d}$ and can be solved easily (see below). The full split Bregman iterative algorithm proceeds as follows:

$$\vec{u}^{k+1} = \arg\min_{u} \tfrac{1}{2}\|\mathbb{A}\vec{u} - \vec{f}\|^2 + \tfrac{\lambda}{2}\|\vec{d}^k - \mu\vec{u} - \vec{b}^k\|^2, \tag{4.18}$$

$$\vec{d}^{k+1} = \arg\min_{d} \|\vec{d}\|_1 + \tfrac{\lambda}{2}\|\vec{d} - \mu\vec{u}^{k+1} - \vec{b}^k\|^2, \tag{4.19}$$

$$\vec{b}^{k+1} = \vec{b}^k + \mu\vec{u}^{k+1} - \vec{d}^{k+1}, \tag{4.20}$$

starting from $\vec{d}^0 = \mathbf{0}$, $\vec{b}^0 = \mathbf{0}$, and $\vec{u}^0 = \mathbf{0}$. We use the conjugate gradient method to perform the ℓ_2 minimization in Eq. (4.18). The second step, Eq. (4.19), separates into individual vector

components and can be solved explicitly using shrinkage as

$$d_n^{k+1} = \text{shrink}\left(\mu u_n^{k+1} + b_n^k, 1/\lambda\right). \tag{4.21}$$

The final step of the split Bregman cycle, Eq. (4.20), adds back the residual deficit in the ℓ_1 term, in complete analogy with the Bregman iteration Eq. (4.15). The results do not depend on the value of the parameter λ, although an unsuitable choice will lead to very slow or failed convergence. We find that in practice an optimal λ can easily be found from a few trial runs at a fixed value of μ, and then kept fixed for any μ. Just like FPC, the split Bregman iteration provides an exact solution to the basis pursuit denoising problem Eq. (4.8), but in contrast to the Bregman approach of Sec. 4.4.2, small values of μ may be needed to solve the noiseless basis pursuit problem Eq. (4.4). In practice, we find that the convergence rate of the split Bregman method is almost always faster than those of the Bregman or FPC algorithms, and greatly so for large, ill-conditioned sensing matrices.

4.4.4 Choice of structures for CSCE

An important practical question regards the best strategy for choosing structures σ to include in the training set. Mathematical theorems from compressive sensing provide a definite answer to this question. The key idea is the notion of coherence between the measurement and representation basis. The representation basis $\Phi = \{\phi_j\}$ is used to express the signal as a sparse series expansion (e.g., plane waves form the representation basis for the Fourier series), while the measurement basis $\Psi = \{\psi_k\}$ contains all possible measurements. For the Fourier example in Sec. 4.2, the measurement basis is given by delta functions, i.e., signal values at certain points in time. Assuming that both ψ_j and ϕ_k are normalized and orthogonal, the coherence is defined as the maximum overlap between them: [61]

$$\nu(\Phi, \Psi) = \sqrt{N} \max_{j,k} |\langle \phi_j, \psi_k \rangle|. \tag{4.22}$$

In the Fourier example of Sec. 4.2, the scalar products are all $|\langle \phi_j, \psi_k \rangle| = N^{-\frac{1}{2}}$, which corresponds to the lowest possible coherence, $\nu = 1$. In contrast, the highest possible value $\nu = \sqrt{N}$ would be obtained by directly measuring the amplitudes of the individual sinusoidal components of the

signal, i.e., if plane waves were chosen as the measurement basis functions. Coherence is key in determining the number of measurements required to recover a given sparse signal with S nonzero components: the higher the coherence, the higher the required number of measurements. More quantitatively, the probability of correct signal recovery from M measurements exceeds $1-\delta$ if the number of measurements satisfies $M \geq C\nu^2(\Phi,\Psi)S\log(N/\delta)$, where C is a constant and S is the number of nonzero components; [62, 63] a similar result holds for compressive sensing in the presence of noise. This expression shows that the worst possible strategy for recovering sparse signals is to choose the same measurement basis as the one used in sparse representation ($\nu(\Phi,\Psi) \approx \sqrt{N}$), since this would require a number of measurements equal to the number of unknown coefficients, N.

In cluster expansion, the representation basis are formed by symmetry-distinct cluster types and the measurements are represented by structures σ. The corresponding representation basis functions are Kronecker deltas, $\phi_g(f) = \delta_{fg}$, where f and g are cluster numbers. The measurements are represented by symmetry-inequivalent structures σ, and the corresponding basis functions are given by normalized rows of the cluster correlation matrix, i.e., $\psi_\sigma(f) = \bar{\Pi}_f(\sigma)/\sqrt{\sum_{f'}\bar{\Pi}_{f'}(\sigma)^2}$. The coherence is given by the maximum scalar product between the two, which is

$$\nu(\Phi,\Psi) = \sqrt{N}\max_{\sigma,f}\frac{|\bar{\Pi}_f(\sigma)|}{\sqrt{\sum_{f'}\bar{\Pi}_{f'}(\sigma)^2}}. \tag{4.23}$$

Because random matrices with independent identically distributed (i.i.d.) entries are incoherent with almost any representation basis, they occupy a special place in compressive sensing. If the possible measurements are designed by selecting N uniformly distributed random vectors on the unit sphere, followed by subsequent orthogonalization, the coherence between Φ and Ψ is on the order of $\sqrt{2\log N}$. [45] This suggests the following simple strategy for selecting structures for CSCE:

- Generate M uniformly distributed random vectors $\psi_\sigma(f)$ on the unit sphere ($\sigma = 1,\ldots,M$)
- Orthogonalize $\psi_\sigma(f)$
- Match each $\psi_\sigma(f)$ onto a real structure σ with normalized correlations $\bar{\Pi}_f(\sigma)/\sqrt{\sum_{f'}\bar{\Pi}_{f'}(\sigma)^2}$ approximating $\psi_\sigma(f)$ as closely as possible

The last step can be conveniently performed by enumerating all possible ordered structures up to a certain size of the unit cell using the methods of Refs. 64, 65 and then choosing the best matches from this list. We stress that the somewhat counterintuitive strategy of selecting random structures follows from the general mathematical properties of ℓ_1-based compressive sensing and represents the best possible method for choosing structure sets for CSCE.

Parenthetically we note that selecting structures at random makes for a remarkably simple approach to generating input data. The "structure selection" problem, that is, deciding which structure to use to train the model, has been a vexing problem in the cluster expansion community since cluster expansions first began to be trained with first-principles data. At first, structures that were easy to calculate (few atoms per unit cell) were selected. In later years, more sophisticated approaches came to be used [32, 66][1] , but a simple, easy-to-implement solution has remained elusive. Compressive sensing not only solves the "cluster selection" problem (because it makes unbiased selections from a huge set of clusters) but also overcomes the structure selection problem because it dictates that the best strategy is to select ordered structures with pseudorandom correlations.

4.4.5 Effect of noise and its relation to optimal μ

The lone adjustable parameter, μ, should be chosen to achieve the optimal balance between the sparseness of the ECI's and the RMS fitting error for the training set. The effect of μ on the calculated ECI's is most transparently seen by analyzing the FPC equations (4.9) and (4.10), which show that μ controls the energy cutoff for the gradient of the ℓ_2 norm of the residuals: components of $\vec{g}$ with absolute values $|g_f| \leq \mu$ will be set to zero by the shrinkage operator and therefore will be excluded from the model. In what follows, we show that the optimal value for μ is proportional to the level of noise (random and systematic) in the training data.

We first consider the relation between the normalized sensing matrix $\mathbb{A}$ in Eq. (4.9) and the CSCE correlation matrix $\bar{\Pi}$: they are related by $\mathbb{A} = \bar{\Pi}/\sqrt{\alpha_{\bar{\Pi}}}$, where $\alpha_{\bar{\Pi}}$ is the largest eigenvalue of $\bar{\Pi}\bar{\Pi}^T$. The corresponding relation for the measurement vectors is $\vec{f} = E/\sqrt{\alpha_{\bar{\Pi}}}$. The distributions of the extremal eigenvalues for ideal random matrices are known from the theory of principal compo-

[1] Teck Tan and Duane Johnson recently developed an as-yet-unpublished method for structure selection that applies fractional factorial design to the structure selection problem (private communication)

nent analysis. [67] However, it is not immediately clear that the eigenvalue distributions found for i.i.d. random matrices will be directly applicable to the CSCE correlation matrices $\bar{\Pi}$ because the correlation values for real structures are neither independent nor identically distributed, and hence the entries of $\bar{\Pi}$ are only approximately i.i.d. We have numerically calculated the distribution of the largest eigenvalue of $\bar{\Pi}\bar{\Pi}^T$ using subsets of $1 \leq M \leq 500$ fcc-based ordered structures with 12 or fewer atoms in the unit cell. [64] We considered $N = 986$ correlations (up to six-body terms) and averaged the calculated eigenvalues over 1000 subsets randomly drawn from the above list of 10850 structures. We find that, for a fixed N, the average value of $\alpha_{\bar{\Pi}}$ increases linearly with the number of structures M. Therefore, $\mathbb{A} \propto \bar{\Pi}/\sqrt{M}$.

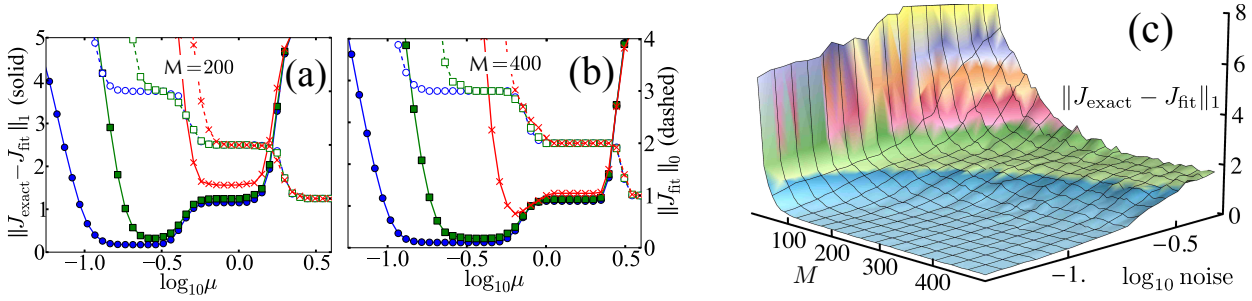


Figure 4.2: $\|J_{\text{exact}} - J_{\text{fit}}\|_1$(solid) and $\|J_{\text{fit}}\|_0$(dashed) vs $\log_{10}\mu$ for the short-ranged pair model with $M = 200$ (a) and $M = 400$ (b). Random uniform noise of $\sim 10\%$(blue circles), 20% (green squares), and 50% (red "x"s) of the noiseless energies was added to the fitting structures. (c) $\|J_{\text{exact}} - J_{\text{fit}}\|_1$ vs the number of fitting structures and the noise level. Each point represents an average over $\sim$100 different subsets of M structures.

Random noise: Here we demonstrate that CSCE is not only stable with respect to noise in the input data, but that it can also filter out the effects of noise on the calculated ECI's. We assume that the DFT formation energies $E(\sigma)$ contain random noise which is represented by a vector $\vec{\eta}_\varepsilon$ of length M and i.i.d random components with variance $\varepsilon^2_{\text{rand}}$. The contribution of $\vec{\eta}_\varepsilon$ to the FPC gradient in Eq. (4.9) is given by

$$\delta g_f \propto -\frac{1}{M}\sum_{\sigma=1}^{M} \bar{\Pi}_f(\sigma)\eta_\varepsilon(\sigma), \tag{4.24}$$

where the factor $1/M$ comes from the fact that both the sensing matrix $\mathbb{A}$ and the measurement vector $\vec{f}$ are related to the correlation matrix $\bar{\Pi}$ and input energies E by a normalization factor $1/\sqrt{\alpha_{\bar{\Pi}}}$. If the structures are chosen randomly according to the prescription outlined in Sec. 4.4.4, then $\bar{\Pi}_f(\sigma) \in [-1,1]$ are approximately i.i.d. Hence, the individual terms under the summation sign in Eq. (4.24) will be randomly distributed with a mean of zero and a variance proportional to $\varepsilon^2_{\text{rand}}$. To deduce the behavior of δg_f in the limit of large M, one can apply the central limit theorem (CLT) of classical statistics, which states that the average of M random terms is normally distributed with a variance that is given by the variance of the individual terms divided by M, i.e., the variance of δg_f is proportional to $\varepsilon^2_{\text{rand}}/M$. It then follows from the properties of the normal distribution that the average ℓ_1 norm of the noise term in the gradient decreases with the size of the training set as

$$||\delta g_f||_1 \propto \frac{\varepsilon_{\text{rand}}}{\sqrt{M}}. \tag{4.25}$$

This relation demonstrates an important noise-tolerance aspect of CSCE, which guarantees that the true physical ECI's will be recovered even if the training data sets contains uncorrelated random noise of arbitrary magnitude, provided that the number of data points is sufficiently large. The practical significance of this feature cannot be overstated: not only is CSCE stable with respect to random noise, but an absolute numerical accuracy in the DFT energies is not even needed to recover the correct ECI's![2] Equation (4.25) also offers guidance for choosing μ to smooth the effect of random noise: as long as $\mu \simeq ||\delta g_f||_1$, the contribution of noise to the gradient will be zeroed out in the shrinkage step [Eq. (4.10)] and will not affect the calculated ECI's. In practice, however, the optimal value of μ is difficult to determine using Eq. (4.25) because the level of noise in the DFT formation energies is not known *a priori,* and approaches based on optimizing the predictive error are more practical.

Systematic noise: We next consider the effect of systematic noise due to errors in the ECI's, which we denote by δJ_f. These errors contribute a term $\delta E(\sigma) = \sum_f \bar{\Pi}_f(\sigma)\delta J_f$ to the residual, and the corresponding error in the FPC gradient is given by

$$\delta g_f \propto -\sum_{f'} \langle \bar{\Pi}_f \bar{\Pi}_{f'} \rangle \delta J_{f'}, \tag{4.26}$$

[2]This applies only to random numerical errors in the DFT formation energies and excludes systematic errors, such as those due to the approximate nature of the exchange-correlation functionals.

where we have introduced a correlation matrix for cluster correlations $\bar{\Pi}$ calculated over the training set:

$$\langle \bar{\Pi}_f \bar{\Pi}_{f'} \rangle = \frac{1}{M} \sum_{\sigma=1}^{M} \bar{\Pi}_f(\sigma) \bar{\Pi}_{f'}(\sigma). \tag{4.27}$$

This matrix is of fundamental importance for CSCE because it describes how the value of one ECI is affected by errors in the other ECI's, or the degree of cross-contamination between systematic ECI errors. Minimum sensitivity to cross-contamination is achieved when $\langle \bar{\Pi}_f \bar{\Pi}_{f'} \rangle$ is diagonal, but the latter case is impossible to realize in practice due to the fact that there are rather pronounced correlations between the cluster averages in real structures. In the best case scenario, the correlation matrix $\langle \bar{\Pi}_f \bar{\Pi}_{f'} \rangle$ will be approximately diagonal if the training set structures are chosen randomly according to the algorithm proposed in Sec. 4.4.4. Indeed, if the average cluster correlations $\bar{\Pi}_f(\sigma)$ are approximately i.i.d., the off-diagonal elements of the correlation matrix $\langle \bar{\Pi}_f \bar{\Pi}_{f'} \rangle$ tend to zero with increasing M, while the diagonal elements remain $O(1)$:

$$\langle \bar{\Pi}_f \bar{\Pi}_{f'} \rangle = \begin{cases} \langle \bar{\Pi}_f^2 \rangle \text{ for } f = f' \\ O\left(\frac{1}{\sqrt{M}}\right) \text{ for } f \neq f' \end{cases} . \tag{4.28}$$

Hence, in the limit of large M, CSCE based on a randomly chosen training set cleanly separates the contributions of the systematic ECI errors to the gradient, i.e., the ECI error for cluster f only affects the component f of the gradient, enabling accurate recovery of the correct solution. This is an important feature for any physics model-building approach because it guarantees the stability of the solution with respect to the interactions that are not represented within the chosen basis set. Furthermore, these considerations offer another insight into the physical meaning of the parameter μ: it can be used to filter out the cross-contamination due to effects of systematic noise if chosen as

$$\mu \sim \frac{||\delta \vec{J}||_\infty}{\sqrt{M}}, \tag{4.29}$$

where $||\delta \vec{J}||_\infty$ is the magnitude of the largest error in the cluster interactions. Since the diagonal contribution to the gradient remains constant with increasing M, successively smaller ECI's can be extracted by increasing the size of the training set M and simultaneously decreasing the value of μ according to Eq. (4.29). Unfortunately, the practical value of this expression is limited because the

ECI errors are not known, and approaches based on minimizing the prediction errors or CV scores should be used instead.

The preceding analysis shows that μ can be interpreted as a parameter controlling the filtering of the noise in the calculated energies, including both random noise due to numerical errors in the DFT formation enthalpies and systematic noise due to cluster interactions that are not recoverable using the given structure set. Expressing the total noise level as a sum of random and systematic contributions, $\varepsilon^2 = \varepsilon^2_{\text{rand}} + \varepsilon^2_{\text{syst}}$, the effect of both is expected to decrease as the inverse of the size of the training set, and the optimal value of μ is expected to vary as $\frac{1}{\sqrt{M}}$. We note here that the Bregman and split Bregman iterations contain additional noise-filtering steps [Eqs. (4.15) and (4.20)] which add back the residual to the residual of the next iteration. As a result, the optimal value of μ will in general vary between the different ℓ_1 optimization approaches, even though the solutions and the predictive errors are practically the same. [58, 60]

4.5 Applications

4.5.1 Short-ranged pair model with noise

We first work with an ad-hoc cluster expansion example where we choose a set of sparse coefficients and then use them to compute the energies of various crystal structures for use as input to CSCE. The advantage of this approach is that knowing the exact solution *a priori* allows us to easily determine the accuracy of the solution found by CSCE and determine how numerical noise influences the performance of the algorithm. While this example is certainly not representative of any real alloy system, it clearly illustrates some key features of the method, particularly how CS performs with noisy data.

Using the UNCLE [26] framework the following clusters on an fcc lattice were enumerated: 141 pairs, 293 triplets, 241 four-bodies, 87 five-bodies, and 222 six-bodies (986 clusters in total, including the onsite and empty clusters). The coefficients of the three shortest nearest-neighbor pairs were chosen as 10, 4, and 1, respectively; all other coefficients were set to zero. Uniformly distributed random noise equal to $\sim$ 10%, 20%, and 50% of the noiseless energies was added to the computed energies $E(\sigma)$. We emphasize that these noise levels *significantly* exceed typical

numerical errors in the calculated formation enthalpies from state-of-the-art quantum mechanics codes.[3]

The values of each of the 986 basis functions were computed for all structures in the training set, thus forming the sensing matrix, $\mathbb{A}$. The rows of the sensing matrix, $\mathbb{A}$, which each represent a training set structure, were constructed by drawing randomly from a uniform distribution on $[-1,1]$. For real systems, such as Ag-Pt in the next section, these rows should be mapped onto real crystallographic configurations as described in Sec. 4.4.4. However since the quality of the fit for the short-ranged pair case was found to be unaffected by this mapping, either favorably or adversely, we chose to simply use the random vectors themselves in order to simplify computations.

Figures 4.2(a) and 4.2(b) illustrate the performance of CS by showing two quantities: 1) the ℓ_1-norm of the difference between the exact and fitted coefficients ($\|J_{\text{exact}} - J_{\text{fit}}\|_1$), and 2) the number of non-zero coefficients (ℓ_0-norm of the solution, $\|J_{\text{fit}}\|_0$). We varied μ to investigate it's optimal values for a given noise level. Each data point in Fig. 4.2 was obtained by averaging over approximately 100 different sets, each of size $M = 200$ or 400.

The curves in Fig. 4.2 exhibit a series of plateaus, each one indicating a region over which the extracted solution remains practically unchanged. Notice, for example, the plateau located between $\log_{10}\mu = -0.75$ and $\log_{10}\mu = -0.4$ in the $\|J_{\text{fit}}\|_0$ *vs.* μ curve for $M = 200$ and the lowest noise content (circle markers). This plateau indicates that CSCE has extracted three non-zero coefficients. Furthermore, the value of $\|J_{\text{exact}} - J_{\text{fit}}\|_1$ drops close to zero in this range, indicating that CSCE has found essentially the exact answer. Using values of μ below the optimal range results in sharp increases in both the number of nonzero coefficients and in the error $\|J_{\text{exact}} - J_{\text{fit}}\|_1$, indicating overfitting.

Conversely, μ values above the optimal range result in fewer non-zero coefficients and an incremental increase in $\|J_{\text{exact}} - J_{\text{fit}}\|_1$, probably indicating underfitting. As a function of increasing μ, one first obtains a plateau where the CS reproduces the two largest expansion coefficients (10 and 4), followed by another plateau where only the largest coefficient is reproduced. This example illustrates the important point that CS is largely *insensitive* to the choice of μ—the ability to recover

[3] In our estimation, numerical errors in the calculated DFT formation energies are only a few meV/atom for the case of Ag-Pt compounds considered in Sec. 4.5.2.

the correct solution does not depend on the exact value of μ, as long as it lies within an optimal, but broad, range.

Upon increasing the noise in the fitting data at a fixed data set size [compare the curves marked by circles and squares in Fig. 4.2(a)], the plateaus in $\|J_{\text{fit}}\|_0$ vs μ become narrower until the highest plateau, corresponding to full recovery of the true solution, disappears completely ("x" markers in Fig. 4.2). At the same time, the minimum in the error $\|J_{\text{exact}} - J_{\text{fit}}\|_1$ vs. μ is increasing incrementally. This displays the robustness and stability of CS—even at a very high noise level we are able to recover the majority of the signal content.

The shift towards higher values of optimal μ upon increasing noise level in Fig. 4.2 is consistent with the physical interpretation of μ as the threshold for noise filtering given in Sec. 4.4.5. We also note that an increase in the number of structures M tends to slightly lower the optimal μ, which can be attributed to a fuller recovery of the correct solution and an associated decrease in the systematic noise.

Figure 4.2(c) displays $\|J_{\text{exact}} - J_{\text{fit}}\|_1$, averaged over approximately 100 random subsets, as a function of M, the number of fitting structures, and the noise level. Here we see the same plateau structure found in Fig. 4.2(a), with the lower (blue) plateau indicating essentially an exact fit. This plot demonstrates that, for all noise levels considered (up to as high as 50% of the noiseless energies!), there remains a training set size for which the exact solution will be recovered.

4.5.2 Actual alloy example: Ag-Pt

Having explained the basic properties of CSCE for a model system, we now test its performance on real DFT data for binary Ag-Pt alloys on a face-centered cubic (fcc) lattice. Ag-Pt was chosen due to a report of unusual ordering tendencies [68] which are non-trivial to reproduce with current state-of-the-art CE methods. The energies of more than 1100 Ag-Pt fcc-based crystal structures[4] were calculated from the density-functional theory (DFT) using the VASP software. [69, 70] We used projector-augmented-wave (PAW) potentials [71] and the generalized gradient approximation (GGA) to the exchange-correlation functional proposed by Perdew, Burke and Ernzerhof. [12] To reduce random numerical errors, equivalent k-point meshes were used for Brillioun

[4] Such a large number of structures was only chosen to test the performance of different CE methods and is several times larger than typical training set sizes used in state-of-the-art CE methods.

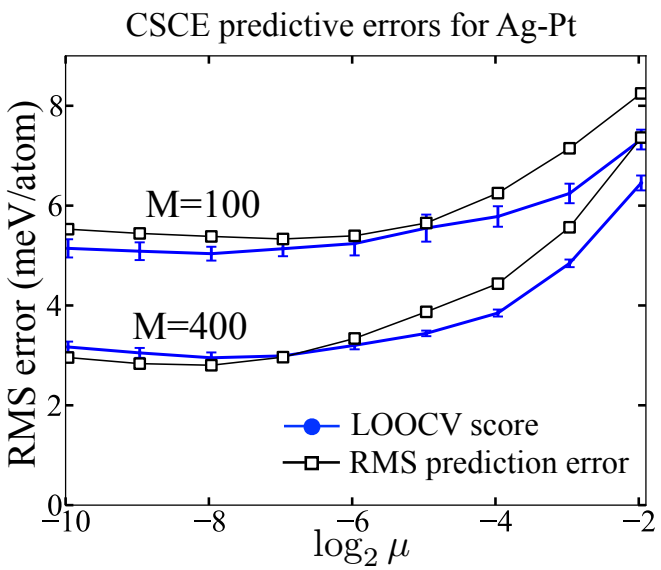


Figure 4.3: Root-mean-square errors for the prediction set (black line with empty squares) and the leave-one-out cross-validation score (LOOCV, solid blue line) as functions of the parameter μ. LOOCV has been averaged over 10 randomly drawn sets of 100 (400) structures, and the error bars were calculated from the variance in the predicted LOOCV scores over these sets. Predictive errors for the hold-out set and the fitting errors for the training set were averaged over 500 different sets of 100 (400) structures; the corresponding error bars are smaller than the size of the symbols.

zone integration. [19] Optimal choices of the unit cells, using a Minkowski reduction algorithm, were adopted to accelerate the convergence of the calculations. [72] The effect of spin-orbit coupling was not included in our calculations because it's effect was shown to be a simple tilt of the calculated energies, as explained in Ref. 73.

Out of a total of approximately 1100 structure energies calculated for this system, 250 were chosen at random to be held out of the fitting process and used for prediction. This "holdout" set remained unchanged for all fitting sets chosen. Of the remaining 850 data points available for fitting, subsets of up to $N = 400$ were chosen to be used as CSCE training data.

We start by illustrating the performance of two different methods for selecting the optimal value of μ. First, we varied μ and calculated the standard LOOCV score over 10 different randomly drawn subsets of M structures; the results are shown by blue curves in Fig. 4.3. It is seen that the LOOCV scores reach their minima at $\mu \approx 4$ and 2 meV/atom for $M = 100$ and 400, respectively, which we interpret as the optimum μ's providing maximum predictive power. Second, we calculated the average prediction errors for all structures left out of the fitting set, which are represented by the black dotted lines in Fig. 4.3. We see that the RMS errors for the prediction set largely follow the same behavior as the LOOCV scores, reaching minima at nearly identical μ values.

As expected, fitting errors for the training set (not shown here) decrease monotonically with decreasing μ and are significantly smaller than either the LOOCV scores or prediction errors for the hold-out set. The leveling off in both the prediction errors and the LOOCV score at small values of μ can be explained by noting that CSCE fits the training set perfectly and further decrease of μ does not bring about noticeable changes in the calculated ECI's. We note that this behavior is different from the short-ranged pair model in the previous section, where decreasing μ below the optimal range caused a rapid deterioration in the accuracy of the calculated ECI's. We attribute this difference to the lower level of noise in the Ag-Pt case, so that the range of μ's that leads to acceptable ECI's is much wider than at the 20-50% noise level for the short-ranged pair model.

To compare the performance of CSCE with other established methods, a discrete optimization (DO) scheme as implemented in the state-of-the-art ATAT software package, [24,48] was used. Note that the ATAT program is capable of employing advanced algorithms beyond minimization of the LOOCV score to ensure that the ground state line is reproduced correctly and to determine which structures should be used as input. In order to make a straightforward comparison between CSCE and DO and to ensure a reasonable fit construction time for this problem, we only used the LOOCV-based DO functionality of ATAT. Since the DO method for $N = 986$ clusters on a training set of a few hundred structures takes several days to complete, averages were taken over only 10 training sets of size M (except for $M = 400$ when we used 42 different training sets to perform statistical analysis of the calculated ECI's). In order to simulate building a complicated unknown model, we deliberately avoided applying physical intuition (e.g., picking short-range interactions) and simply performed the optimizations with minimal restrictions. The maximum number of reported ECI's was capped to $M/4$ for ATAT-based DO. For CSCE, we used a fixed $\mu = 8$ meV/atom and computed solutions for 500 randomly chosen training sets of M structures.

Figure 4.4 shows a box and whisker plot of the RMS errors over the prediction set for CS solutions and the mean RMS values for the DO solutions (box-and-whiskers were not used for DO solutions due to the small number of DO fits). Each box and whisker represents RMS values for approximately 500 different fits. We see that CSCE achieves an RMS error value much lower (2.8 meV/atom) than LOOCV-based DO (6.8 meV/atom). Furthermore, Fig. 4.4 shows that the ℓ_1 norm of the solution increases almost linearly for the DO fit, while it levels off for the CSCE

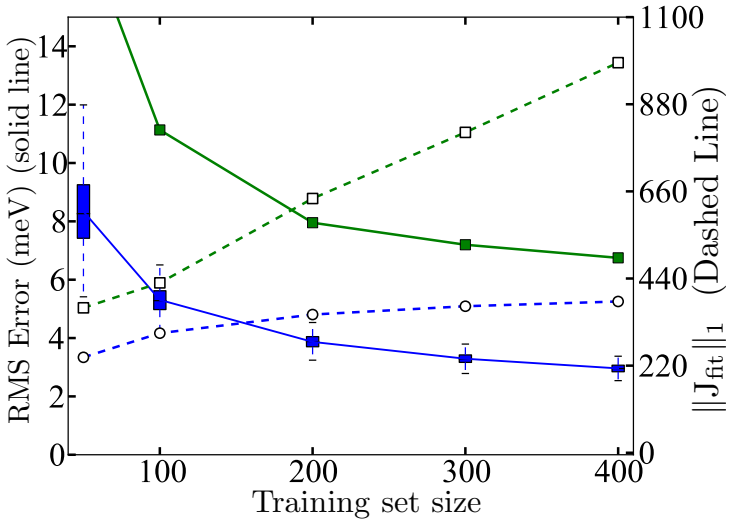


Figure 4.4: Results from compressive sensing and leave-one-out cross-validation for the fcc-based, Ag-Pt alloy system. The solid line gives the root-mean-square (RMS) errors for predictions made on a constant holdout set for CS (box and whisker) and leave-one-out cross-validation (squares). The dashed lines give the ℓ_1-norm of the solution vector for both methods.

fit, indicating that the latter is converging towards a stable solution, while the former keeps adding large ECI's, a behavior suggestive of over-fitting.

4.5.3 Statistical analysis of Ag-Pt ECI's

Because CSCE is fast, thousands of fits for many different training sets can be computed in a few minutes. The results of all these fits can be analyzed statistically to determine which coefficients are consistently identified as contributors and to eliminate artifacts due to a particular choice of the training set. This functionality, the ability to gather enough data in a reasonable amount of time to perform statistical analyses, is a significant advantage of CSCE over (slower) DO methods that can be used to gain insight into the probability distributions for the cluster interactions. These distributions can be used to quantify the uncertainty in the CSCE predictions for physical properties that go beyond a simple LOOCV score or an RMS prediction error. For instance, one can draw ECI's from the calculated distributions and generate ground state convex hulls with statistical error bars on each structure, quantifying the uncertainty in the predicted $T = 0$ K phase diagrams.

CSCE fits for 500 different fitting set choices were computed for Ag-Pt. Most of the resulting distributions had only one sharp peak at zero, indicating that, independently of the choice of the training set, they were almost never selected by CSCE and therefore should be set to zero. Several ECI's exhibited a unimodal distribution with nonzero mean, which were interpreted as strongly significant nonzero interactions. Finally, a fraction of the ECI's showed bi-modal distri-

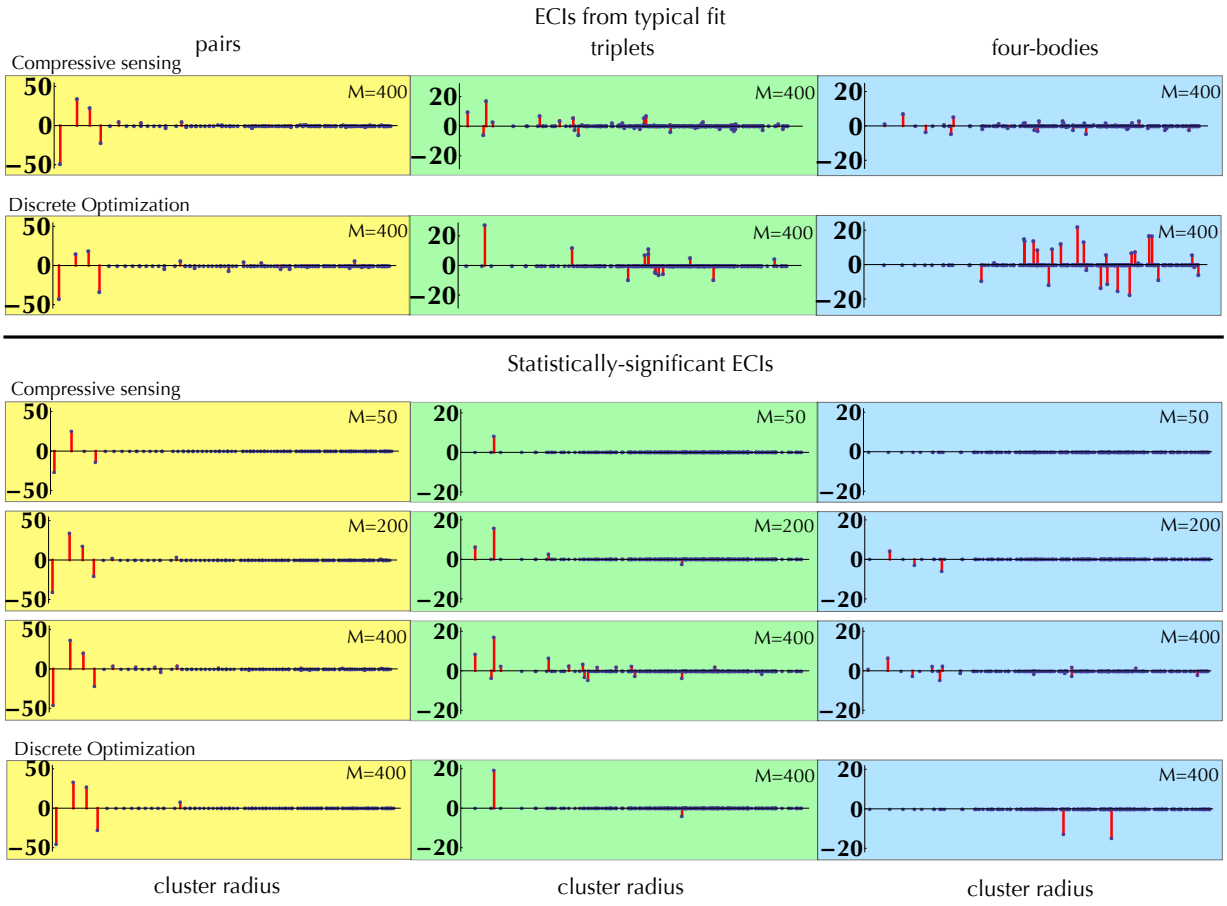


Figure 4.5: Comparison of the interaction coefficients found using the DO method implemented in ATAT software and compressive sensing. The upper pane shows a comparison of two typical fits from CS and ATAT. The lower pane shows the coefficients that were found to be statistically relevant from both methods. The x-axis is the cluster radius, which is defined as the average distance from the center of mass of all cluster vertices. (Blue dots were placed on the x-axis even for clusters not found to be relevant to help the reader know the ordinal number of the relevant clusters.) Physical intuition suggests that shorter-radius, fewer-vertex clusters are the most important contributors in alloy energetics. Pair interaction coefficients found by both methods are similar. As the number of vertices increases, CS finds coefficients in harmony with physical intuition, while DO finds spurious, long-ranged three- and four-body interactions. CS solutions also demonstrate a convergence to one specific solution as the size of the fitting set increases. (note: Triplets and quadruplets are shown on a scale from -20 to 20 meV, different from the scale used for the pairs.)

butions with two peaks of comparable weight and one of the peaks centered at zero energy. Since the latter ECI's were selected by CSCE with an approximately 50% probability, they belong to the class of "marginal" interactions which were counted as significant only if their distribution mean was greater than one standard deviation. To make a fair comparison between CSCE and the DO method implemented in the ATAT program, the same statistical criteria for determining relevant coefficients was used for the DO fits, even though data for only 42 fits were available.

Figure 4.5 gives a comparison of the CS-determined coefficients and those found by DO. The upper pane compares a typical DO fit with a typical CSCE fit, while the lower pane gives a comparison of statistically relevant ECI's from both methods. The CSCE-derived ECI's appear to evolve towards one specific solution as the size of the fitting set increases, indicating convergence of the solution. Notice also that the magnitudes of the CSCE coefficients decrease as the spatial extent of the cluster increases and as the number of cluster vertices increases (note that triplets and quadruplets are shown on a scale from -20 to 20 meV, as opposed to -50 to 50 meV for pairs). This is in harmony with long-standing claims in the CE community, and it confirms that a stable solution has been found. DO-determined clusters follow this pattern for pair clusters only. At higher vertex numbers, a typical DO fit finds non-physical, spurious coefficients for three- and four- body interactions. The set of statistically-relevant DO coefficients appear to be lacking several important interactions, specifically short-ranged three- and four-body interactions. This indicates that: (i) current DO methods are much too slow to be able to gather enough statistics to do a meaningful statistical analysis, and/or (ii) current DO methods are very sensitive to the choice of the training set and fall short in their ability to identify physically relevant interactions without user guidance.

Note that the mathematical framework of CS has no knowledge of the spatial extent or geometry of the cluster functions. Remarkably, the dominant expansion coefficients, regardless of spatial extent, are efficiently retrieved using CS. In cases where a purely real-space cluster expansion fails to converge, CS may fail to construct a suitable model, but it could be combined (as has been done with other approaches) with reciprocal-space fomulations. [51, 54, 74, 75]

Figure 4.6 shows the results of a ground state search performed by using the statistically significant $M = 400$ coefficients to predict the energies of all fcc-based superstructures up to 12 atoms. Error bars were calculated from randomly drawn sets of $M = 400$ structures. The ground

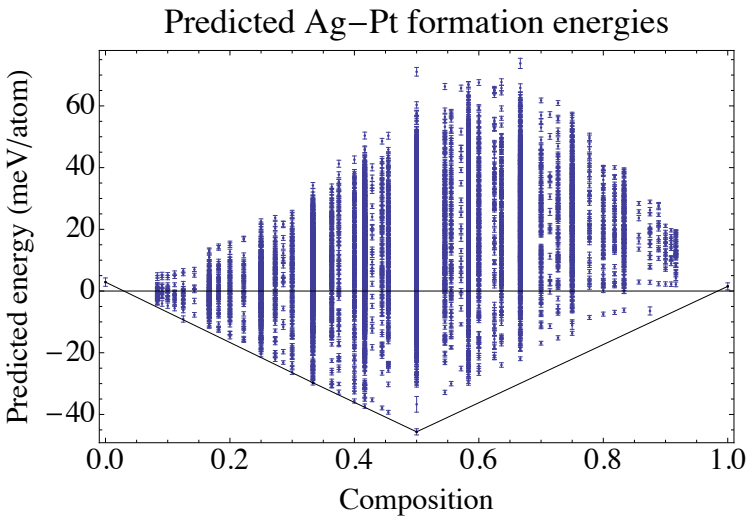


Figure 4.6: Predicted CSCE formation energies obtained using the ECI's shown in Fig. 4.5; error bars are standard deviation due to different random choices of $\leq$ 400 structure subsets. Black solid line denotes the convex hull calculated from the average energies; only Ag, Ca_7Ge-type Ag_7Pt (barely, with a depth of less than 1 meV/atom), $L1_1$ AgPt, and Pt are predicted to be $T = 0$ K ground states.

state line in this figure is consistent with first-principles data for this system, which finds the same ground states as in Fig. 4.6, with a few degenerate structures lying on the convex hull between $c = 0.4$ and 0.5.

This example shows that, in comparison with traditional cluster selection methods, CS is not only simpler and faster (less than a minute on a single CPU for CS versus *days* for LOOCV at $M = 400$), but also produces more physical solutions that result in a significant improvement in physical accuracy.

4.5.4 Protein folding application

We now turn to a technically much more challenging case—that of protein design in biology. Modeling the protein folding energies in the zinc-finger motif represents a technically difficult test case with applications in biology. [76, 77] One of the key problems in protein design is to find the sequence of amino acids (AAs) which stabilizes a particular 3D structure, or *folding*. Physics-based energy functionals are considered to be some of the most-promising methods in protein design since they link the stability of the the folded 3D structure to the total free energy, accurately accounting for electrostatics, van der Waals interactions, and solvation effects. However, their use is problematic due to the astronomical number of possible AA sequences for even very short proteins. It was shown [76, 77] that the CE model can be generalized to describe protein energetics, allowing very fast direct evaluation of the protein energy as a function of its sequence.

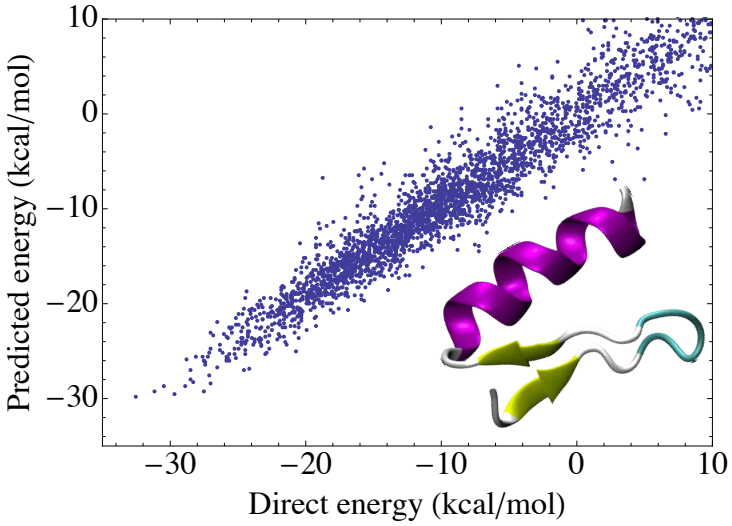


Figure 4.7: Predictive performance of CS for protein energetics in the zinc-finger structure (shown in the inset).

Here, we use the data from Ref. 76 for the so-called zinc-finger protein fold and closely follow exactly the same computational procedures as employed in that study. The fitting is done using a basis of approximately 76,000 clusters and energies of 60,000 AA sequences; a separate set of 4,000 AA sequence energies is used to test the predictive power of the CE model. The very large size of the problem presents a severe test to the conventional LOOCV-based model building approach, requiring running times of several weeks on parallel computers with user-supervised partial optimization. [76] We chose the highly efficient split Bregman iteration [60] for solving the basis pursuit denoising problem in Eq.(4.8), which allows us to perform a full optimization in approximately 30 minutes on a single 2.4 GHz Intel Xeon E5620 processor. Figure 4.7 shows that for the physically important negative-energy configurations, we are able to achieve an RMS predictive error of 2.1 kcal/mol with 3,100 model parameters, significantly better than the RMS error of 2.7 kcal/mol with approximately 6,000 parameters obtained using the LOOCV method in Ref. 76. Since the predictive errors are Gaussian-distributed with a mean of zero, the statistical uncertainty in the predictive error due to the finite size of the prediction set (> 1000 negative-energy structures) can be calculated using standard statistical formulas for the χ^2-distribution; they are found to be less than 1% of the calculated RMSE. These results show that the computational efficiency, conceptual simplicity and physical accuracy of the ℓ_1-based minimization shows promise for future applications in protein design.

4.6 Conclusion

In conclusion, compressive sensing can be straightforwardly adopted to build physical models that are dominated by a relatively small number of contributions drawn from a much larger underlying set of basis functions. Compressive sensing is applicable to any "sparse" basis-expansion problem, a broad class of problems in physics, chemistry and materials science. Compressive sensing allows the identification of relevant parameters from a large pool of candidates using a small number of experiments or calculations—a real paradigm shift from traditional techniques. Furthermore, many other scientific problems that do not appear to be a basis pursuit problem may be recast as one, in which case CS could efficiently provide accurate and robust solutions with relatively little user input. With the huge amount of experimental and computational data in physical sciences, compressive sensing techniques represent a promising avenue for model building on many fronts including structure maps, empirical potential models, tight binding methods, and cluster expansions for configurational energies, thermodynamics and kinetic Monte Carlo.

In the arena of cluster expansion, compressive sensing provides a simple solution to two challenges: "cluster selection" and "structure selection." Cluster selection is effectively solved because compressive sensing can select clusters efficiently from a very large set (thousands or tens of thousands). Essentially, it allows the user to specify a cluster set so large that it encompasses every physically-conceivable interaction. The second challenge, structure selection, is overcome by the fact that compressive sensing *requires* that input structures simply be chosen randomly from configuration space.

Chapter 5

Bayesian Compressive Sensing

5.1 Introduction

Technological advances are driven by the discovery and development of high-performing materials. Discovering these materials is perhaps the single largest bottleneck to technological developments. Due in large part to advances in computing power, computational methods play an increasingly important role in the discovery process. Results from calculations and simulations guide experimental work and provide insight into avenues for future materials research.

The well-known density functional theory (DFT) is an example of a recent methodological stride in computational materials research. Developed in the 1960's, this theory paved the way to accurate and efficient calculations of materials' properties. Steady advances in computing power have made these calculations more affordable computationally, and therefore more viable as a way to probe nature for high-performing materials. This is manifest by recent high-throughput studies that identify new materials and uncover new properties through brute-force calculation of all likely candidates. Results from these studies have been fruitful and illustrative. [78–80]

Although useful for some purposes, high-throughput DFT studies are far from exhaustive in their scope of search, and provide no thermodynamic information about the material. To extend computation's reach, a common approach is build a model, trained from DFT data but which is much faster. These models can quickly calculate important physical quantities for millions of candidate structures. A whole host of thermodynamic simulations also become available once a fast, reliable model is constructed.

Here we employ a Bayesian implementation of compressive sensing (BCS) to construct cluster expansion models. BCS addresses, in a mathematically rigorous fashion, two major and long-standing challenges in the cluster expansion community, namely the basis selection problem and the choice of training data problem. BCS provides a parameterless framework, considerable

speed up over current techniques, and error estimates on coefficient values. A re-weighting scheme (section 5.4.1) is wrapped around the BCS framework to further enhance the sparsity (quality) of the solutions and reduce start-to-finish time. Most impressive is the fact that re-weighted-BCS-constructed cluster expansion models exhibit a convergence of the solution to a very physical model that predicts more accurately than all other modern-day methods.

5.2 The cluster expansion

One commonly used model for exploring substitutional order in materials is the cluster expansion, which provides a fast, accurate way to compute the total energy of all atomic configurations on a parent lattice. [20–22] The cluster expansion is constructed by first assigning each atomic type a pseudo-"spin" variable. Any atomic configuration on the parent lattice can then be specified using a vector of pseudo-spin variables. The physical quantity of interest is then expressed as a linear combination of basis functions, an idea very analogous to a Taylor or Fourier expansion

$$E(\sigma) = E_0 + \sum_f \bar{\Pi}_f(\sigma) J_f. \tag{5.1}$$

Here the argument to the function, σ, is a vector of pseudo-spin variables indicating the atomic occupation on the parent lattice sites. The vector σ represents a specific structure (unit cell and atomic configuration). The $\bar{\Pi}_f$ are the basis functions, often referred to as cluster functions, with each function corresponding to a cluster of lattice sites. For binary systems, these basis functions are evaluated by averaging over products of pseudo-spin variables (For higher component systems, the basis is more complex). The J_f are the expansion coefficients and finding their values is the critical task when constructing a cluster expansion.

The cluster expansion is essentially a linear algebra problem

$$\bar{\Pi}\mathbf{J} = \mathbf{E}, \tag{5.2}$$

with $\mathbf{E}$ containing the first-principles training data, and $\mathbf{J}$ the sought-after coefficients (ECI's). Early in the development of cluster expansion, the ECI's were found by directly inverting Eq. 6.5. This so-called structure inversion method (SIM) [23] is conceptually appealing, but in practice the resulting model has poor predictive capability. As the CE method developed, the best practice that

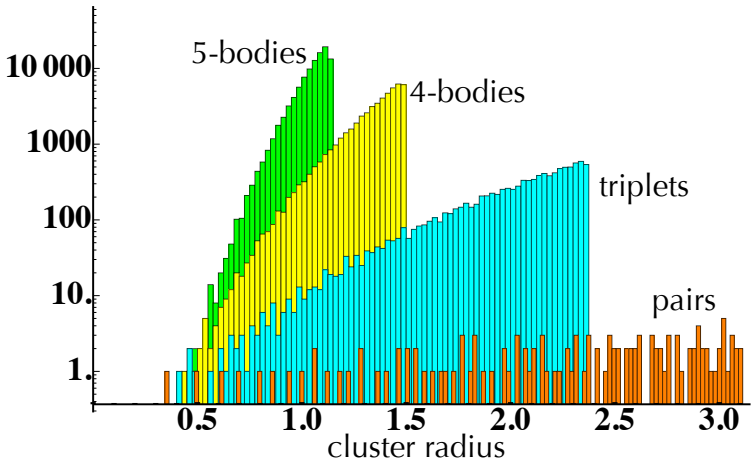


Figure 5.1: Histogram of geometrically unique clusters on an fcc lattice. The x axis is the cluster radius, which is defined to be the average distance from the cluster center of mass to the cluster vertices. The number of unique clusters increases exponentially as the number of cluster vertices and cluster radius increase. This illustrates the magnitude of the challenge associated with truncating the cluster expansion.

emerged was to generate more fitting data than fitting variables (more elements in the vector **E** than in the ECI's vector **J**). This results in an overdetermined problem that can be solved by singular value decomposition or related methods. Before discussing fitting approaches in more detail, we point out that whatever the details of the fitting procedure are, any method must deal with two difficulties: (1) The expansion given in Eq. 6.1 must be truncated to a finite (and typically small) number of terms, and (2) a choice must be made about which structures (among a practically infinite set) should be used as training data (to generate the vector **E**). The expansion must be severely truncated so that it has fewer terms than the number of training structures (maintaining an overdetermined problem), and the training structures should be chosen to minimize the predictive errors. Mathematically speaking, the choice of the training structures is not independent of the truncation.

Both of these difficulties are challenging. The first is difficult because the number of relatively short-ranged clusters is enormous (see Fig. 5.1) so a robust distance- or hierarchy-based truncation method is not apparent. It is difficult to avoid truncating relevant terms inadvertently. There are several contemporary approaches to truncation problem [24–29, 51–54]. The second challenge, choosing the structures to be used as training data, depends on the first. The optimal choice of training structures depends on the truncation. Some approaches attempt to choose training structures so as to minimize the variance in predictive errors. [31, 32, 50] Others, based on the early work of Garbulsky, [81] attempt to bias the training set to reproduce the correct ordering of low-energy states. [26]

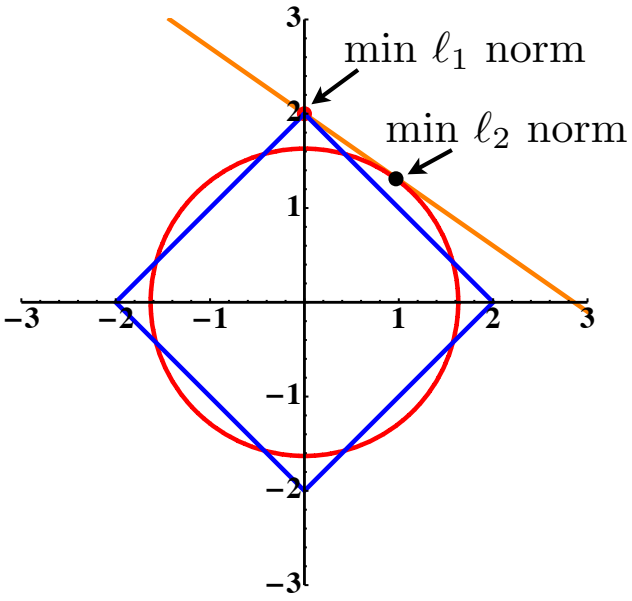


Figure 5.2: Illustration of constant ℓ_p norm surfaces in R^2. The circle is a constant ℓ_2 norm surface and the diamond is a constant ℓ_1 norm surface The straight line indicates the possible solutions to the underdetermined problem $10y + 7x = 20$. A sparse solution to this problem is the solution where one of the variables is zero and the other is not, in other words it is at the intersection of the straight line and the axes. Minimizing the ℓ_2 norm of this system will result in a dense solution, whereas minimizing the ℓ_1 norm will yield a sparse solution.

With the exception of recent CE techniques based on Bayesian inference [1, 29, 30], the model-building process of contemporary techniques are essentially the same: An initial set of training data is generated and a fit is calculated. The predictive accuracy of the model is assessed. More training data is added and a more refined model is generated. This process is continued, with more and more terms being included in the expansion, until a model with the desired predictive accuracy is achieved.

5.3 Compressed sensing

Compressed sensing (CS) turns this iterative approach inside out and provides a robust expansion *in one shot*. The iterative approach starts with a simple model (severely truncated) that becomes more and more complex as the training data is expanded. At each stage, the truncation problem becomes more and more difficult. But CS starts with an infinite set of essentially untruncated models, all of which are consistent with the training data, and then discards all of the models except the one which is the most physical.

The number of unique, potentially-relevant clusters is typically very large, and considering all possible clusters suggests solving a highly underdetermined version of equation (6.5) (Many more columns [clusters] than rows [structures] in the matrix $\bar{\Pi}$). This is accomplished using compressed sensing (a.k.a. compressive sampling), a new technique born in the signal processing

community that solves the heavily under-determined problem by constraining the solution search to those solutions with the smallest ℓ_1 norm

$$\min_{\mathbf{J}}\{\|\mathbf{J}\|_1 : \bar{\Pi}\mathbf{J} = \mathbf{E}\}, \tag{5.3}$$

where $\|\mathbf{J}\|_1$ indicates the ℓ_1 norm of vector $\mathbf{J}$, a specific case of the more general ℓ_p norm

$$\|\mathbf{u}\|_p = \left(\sum |u_i|^p\right)^{1/p}. \tag{5.4}$$

The key idea in compressive sensing is the assumption that the solution vector is sparse, or has few non-zero components. The ℓ_1 norm constraint has been used for years as a sparsity measure and is used here to direct the solution search towards the most sparse solution. Since CE models are known to be sparse, CS provides a fast, robust, and efficient way to detect relevant clusters and to compute their corresponding coefficients.[1]

Figure 5.2 illustrates CS for the simple two-dimensional underdetermined problem $10y + 7x = 20$. The straight line in the figure represents all possible solutions corresponding to this system. The circle (diamond) is a constant ℓ_2 (ℓ_1) norm surface. A sparse solution to this system is one where one of the unknowns is non-zero and the other is zero, in other words it is where the straight line intersects one of the axes. The intersection of the solution curve and the constant ℓ_2 norm curve will always occur off-axis yielding a dense solution. The intersection of the solution curve and the constant ℓ_1 norm curve will occur on one of the axes, and therefore yield a sparse solution. Constant ℓ_p surfaces where $0 < p \leq 1$ can enhance the sparsity, but finding the global minimum is an NP-hard, non-convex optimization problem.

The mathematical framework of compressive sensing, put forth by Candes, Romberg, and Tao [47], guarantees the recovery of sparse ECI's from a small number of first-principles total energies given certain properties of the matrix $\bar{\Pi}$ in Eq. (6.5). The mathematical theorems from compressive sensing require that the matrix $\bar{\Pi}$ in Eq. (6.5) take on a certain form, namely that the rows be independent and identically distributed (i.i.d). (For a more complete description of this

[1]The sparsity of a solution, or the number of non-zero components, is commonly referred to as the ℓ_0 norm even though this function is not well-defined mathematically. For this reason, the ℓ_1 norm has been used in place of the ℓ_0 norm as a sparsity-promoting function for decades.

condition see reference 47.) This simple requirement provides a mathematically sound solution to the question of which structures should be used in the training set. To ensure that the rows of the matrix $\bar{\Pi}$ are independent, structures whose correlation vectors are composed of random draws from a uniform distribution should be used as training data. A more detail description of how this is done in practice can be found in reference 1.

The CS-mandated requirement that the rows in the matrix $\bar{\Pi}$ be uncorrelated from one another allows the training set structures to be chosen **once** at the beginning of the model building process instead of using iterative procedures to build up the training set over time. This feature of CS-based CE models provides a very automatic and hands-off framework to the model building process, a sharp contrast to current state-of-the art methods.

The solution to Eq 6.3 was shown to be exact with overwhelming probability if the number of function samples, m, satisfies

$$m \geq C \cdot \mu^2(\Pi, \Psi) \cdot S \cdot \log n. \tag{5.5}$$

where C is some positive constant, n is the number of basis functions being included and S is the sparseness of the solution vector. (An S-sparse solution vector has S non-zero coefficients.) The function $\mu(\Pi, \Psi)$ is a measure of the correlation between training set structures. Eq. (5.5) provides a lower bound on the number of training data points needed to recover the relevant ECIs from a large pool of candidates.

The CS paradigm is uniquely well-suited to the challenges in cluster expansion construction. Not only does compressive sensing solve the cluster selection problem by allowing the inclusion of essentially all clusters, but compressive sensing also gives a well-defined prescription for selecting training data as well as a lower bound on how many are needed (Eq. (5.5)).

Various mathematical techniques exist for solving an underdetermined linear system subject to a constraint. One such method recasts the constrained minimization problem of Eq. 6.3 as the unconstrained minimization problem

$$\min_{\mathbf{J}} \{\mu \|\mathbf{J}\|_1 + \|\Pi\mathbf{J} - \mathbf{E}\|_2^2\}. \tag{5.6}$$

This equation is referred to as the basis pursuit de-noising problem, and one efficient way to a solution is an iterative procedure put forth by Yin *et al.* [58] The sparseness of the solution can be tuned by varying the parameter μ. Smaller(Larger) values of μ mean that the ℓ_1-norm term will be weighted less(greater) than the ℓ_2-norm term and will therefore result in less(more) sparse solutions.

5.4 A Bayesian Implementation

A recently-developed Bayesian implementation of CS [82, 83] provides a *parameterless* framework (automatic), error values on coefficients, and a considerable speed up from current CE construction techniques. When coupled with the re-weighting scheme put forth by Candes *et. al* and outlined in section 5.4.1, BCS enhances sparsity and reduces the total start-to-finish time.

A more complete description of BCS can be found in the Appendix and in reference 82. Here we highlight some key points of the method, assuming that the reader has some prior (no pun intended) knowledge of Bayesian statistics.

The key to merging compressive sensing and Bayesian statistics lies in the choice of prior distribution on the coefficients, J. The Laplace distribution is well known to promote sparsity by placing a large probability mass at the origin thus favoring zero-valued coefficients. However, the Laplace distribution is not conjugate to the normal distribution, which is used as the likelihood. In order to represent the prior information about the coefficients using a Laplace distribution while also preserving conjugacy Babacan *e*t. al. employ a hierarchical approach. In the approach, the prior distribution on the coefficients is chosen to be $\mathcal{N}(0,\gamma)$ (conjugate to the normal likelihood employed), and the hyperprior on γ is chosen to be Laplace. Due to the conjugacy of the prior with the likelihood, the posterior distribution on the coefficients is know to be Gaussian with covariance matrix

$$\Sigma = \left[\beta \Pi^T \Pi + \Gamma\right]^{-1}, \tag{5.7}$$

and mean vector

$$\mu = \Sigma \beta \Pi^T \mathbf{E}, \tag{5.8}$$

where

$$\Gamma = \text{diag}\left(\frac{1}{\gamma_i}\right). \tag{5.9}$$

The parameter β is the inverse variance on the likelihood and gives an estimate to the error in the training data. Once accurate values for these parameters are obtained, the resulting distribution provides the desired estimates on the model coefficients. However, notice that the expressions for Σ and μ depend on other parameters that have been introduced, namely γ and β. Expressions for these parameters are provided through a type II maximum likelihood procedure. In this procedure, analytic expressions for the parameters are obtained by maximizing the joint probability distribution $p(\mathbf{y}, \gamma, \beta)$ with respect to each parameter individually. The full mathematical details for this process can be found in reference [82].

It turns out that the expressions β and γ are dependent on one another. This suggests an iterative procedure for estimating their optimal values. This is done by initially setting all $\gamma_i = 0$. At each iteration a γ_i is selected and it's value is computed. It is informative to notice that the value of the parameters γ_i determines whether basis function i should be included in the model. If $\gamma_i = 0$, $\mu_i = 0$ and the coefficient corresponding to basis function i is identically 0, and thus removed from the model. The remaining parameters are then updated using the newly-computed value of γ_i. This procedure continues with coefficients being added/removed/re-estimated at each iteration until adding new coefficients does not significantly improve the model.

The update of Σ would normally require a costly inverse (especially costly for problems involving large cluster pools). However, updating a single γ_i at each iteration allows for a very efficient update of this matrix. Instead of computing an inverse at each iteration, the relevant entries in the matrix are simply updated. The speed of this implementation hinges critically on this idea. Additionally, since sparse solutions are expected, the matrix Σ can be represented with far fewer dimensions than what would normally be required. The matrix Σ contains information about the spread, or uncertainty, in the value of the coefficients.

5.4.1 Enhancing the sparsity through re-weighted ℓ_1 norm minimization

The ℓ_1 norm is the best, albeit less-than-perfect, measure of sparsity available and has been used for years in this capacity. The more accurate measure of sparsity is given by the ℓ_0 norm, which counts the number of non-zero elements in a vector. However, the ℓ_0 norm is not a norm in a strict mathematical sense and its use in optimization algorithms is not possible.

One drawback with using the ℓ_1 norm as a measure of sparsity is its dependence on the magnitude of the coefficients. The ℓ_1 norm favors solutions with smaller-magnitude coefficients over solutions that are equally sparse (or even slightly more sparse), but whose coefficients have larger magnitudes. To address this imbalance, Candes *et al.* proposed a weighted formulation of the ℓ_1 minimization which penalizes all non-zero coefficients equally. [84] Under this approach the ℓ_1 constrained minimization problem is solved iteratively with the model coefficients being weighted at each iteration according to

$$w_i^{(l+1)} = \frac{1}{|J_i|^{(l)} + \varepsilon}, \tag{5.10}$$

where the index i indicates the basis function being weighted and l is the iteration index. These weights put large and small magnitude coefficients on equal footing by suppressing the contribution of large magnitude coefficients to the ℓ_1 norm. As explained in reference [84], this weighting can be easily enforced by multiplying the sensing matrix by the inverse of the weight matrix

$$\bar{\Pi}(W^{(l)})^{-1}, \tag{5.11}$$

where W is a diagonal matrix with the weights of Eq. (5.10) on the diagonal. Re-weighting was found to increase sparsity and decrease the number of required function measurements.

5.5 Application

Here we demonstrate re-weighted ℓ_1 minimization through Bayesian compressive sensing on cluster expansion models for the binary systems: Cu-Pt, Ag-Pt, and Ag-Pd. Pt group metal alloys have application in catalysis and jewelry, which motivated their study here. Additionally, an alternate implementation of CS was recently used to study Ag-Pt, and a direct comparison to this alloy was desired.

Using the UNCLE software approximately 1000 clusters were enumerated, with approximately the same number from each order up to six-body clusters. For each alloy system, the chemical energies of crystal structures were calculated from the density-functional theory (DFT) using the VASP software. [69, 70] We used projector-augmented-wave (PAW) potentials [71] and the generalized gradient approximation (GGA) to the exchange-correlation functional proposed by

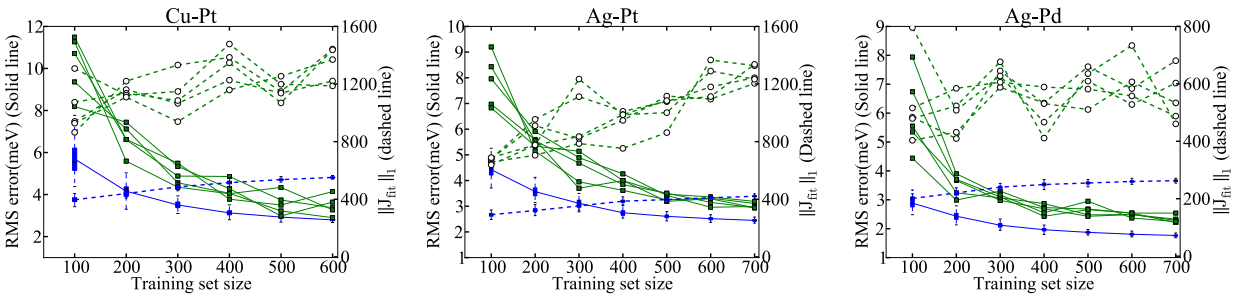


Figure 5.3: Comparison between re-weighted Bayesian compressive sensing and genetic algorithm methods for constructing a cluster expansion model for the binary systems Cu-Pt, Ag-Pt, and Ag-Pd. The solid curves indicate rmse values over a holdout set. The dashed curves represent the ℓ_1 norm of the solution vectors. Approximately 100 BCS fits were performed at each training set size, and the results of these fits are depicted using box-and-whiskers. Due to it's high computational cost, only 5 GA fits were performed, and hence GA results are not depicted using box-and-whiskers.

Perdew, Burke and Ernzerhof. [12] To reduce random numerical errors, equivalent k-point meshes were used for Brillioun zone integration. [19] Optimal choices of the unit cells, using a Minkowski reduction algorithm, were adopted to accelerate the convergence of the calculations. [72] The effect of spin-orbit coupling was not included in our calculations because it's effect was shown to be a simple tilt of the calculated energies, as explained in Ref. 73.

In the absence of the re-weighting procedure, many fits, each using a different training set, must be constructed and the results analysed statistically to identify dominant coefficients. This is needed to increase sparsity and eliminate spurious interactions. However, the re-weighting procedure employed here results in a significant enhancement of sparsity, eliminating the need to average over many solutions. Hence, the results stated below are the exact solutions returned from the re-weighted BCS framework with no post processing whatsoever.

To compare to currently used methods in the cluster expansion community we use the UNCLE code, which uses a genetic algorithm (GA), for the cluster selection/fitting process. GA parameters were set to values that would enable a reasonable computation time and produce typical quality results: 3 populations, 100 generations with 30 children per generation, and a modest mutation rate. While re-weighted BCS is able to consider very large cluster pools, the GA slows considerably as the size of the cluster pool grows. To make a fair comparison, we have used a pool of ~ 1000 clusters for both methods. BCS fits for approximately 100 different choices of the

training set were performed. Due to the high computation cost of a GA fit, fits for only 5 different training set choices were performed with the GA.

The CS paradigm considers all clusters in the pool equally with no explicit restriction on which, or how many, clusters should be used. To make a fair comparison with the genetic algorithm, the maximum number of model coefficients that the GA was allowed to use was set to be 500. In every fit depicted here, the number of model coefficients found was less than 500.

Figure 5.3 give comparisons between GA fits and re-weighted BCS fits for the binary systems Cu-Pt, Ag-Pt, and Ag-Pd respectively. Notice that for every system the root-mean-square error(rmse) over the holdout set is lower for BCS fits for all sizes of the training set. While the rmse of the GA fits is not terrible, the ℓ_1-norm of the solution vector for GA solutions is considerably larger than those from BCS-fits. This is indicative of overfitting and does not foster any confidence that the correct solution has been found. In contrast, the ℓ_1-norm for BCS fits is relatively small and levels off as more training data is added. This is convincing evidence that the solution is converging, and the physical model is being recovered.

Another key feature of BCS is the efficiency of the algorithm. For the three systems discussed here BCS fits were constructed in a fraction of the time needed for the GA. BCS required on the order of minutes to construct 100 fits, whereas the GA needed ~ 24 hours for a single fit.

5.6 Conclusion

It has been shown that the CS paradigm is uniquely well-suited to building CE lattice models. Re-weighted BCS-based provides a fast, efficient, and parameterless framework for constructing CE models. These models are constructed in a fraction of the time required by current state-of-the art techniques and with minimal time and effort required by the user. BCS-constructed CE models converge to a solution which is very inline with widely-held intuition about the nature of physically relevant interactions and predict more accurately than any other modern CE construction method.

From a broader perspective the CS paradigm is poised to have a big impact on computational physics problems of all types. The CS-paradigm is well suited to tackle any highly-underdetermined linear problem: $\mathbb{A}\mathbf{x} = \mathbf{b}$ where $\mathbf{x}$ is known to be sparse. One possible application is the expansion of high-throughput databases to include lattice models. This approach relies

heavily on being able to automatically perform first-principles calculations, and has hitherto not involved using the database information to build materials models. This is mostly due to the high human time cost required to construct such models. However, the hands-off nature of BCS-based CE models will allow materials models to be added to the high-throughput scope of work. In addition to vast amounts of first-principles data, soon high-throughput databases will include accurate lattice models for a diverse array of materials.

Chapter 6

CEFlash: high-throughput CE model construction

6.1 Introduction

The discovery and synthesis of new, high-performing materials has fueled and will continue to fuel technological advances. The role of computation in this discovery process has been significant and promises to expand even further. This is mostly due to increases in computing power over the last half century that make materials simulations and calculations more feasible and affordable.

One recently-emerged technique for uncovering new materials is the so-called high-throughput approach. In this approach, the results of experimental work is combined with first-principles methods to automatically, and intelligently scan over all candidate materials in search of new, advanced materials. In one specific implementation, all experimentally-observed crystal structures are compiled into a database. The energies of these crystal structures are computed, using DFT, for all possible combinations of atomic constituents. The resulting database of first-principles information can then be mined for interesting, new materials. Since hundreds of thousands of DFT calculations are being performed, an automatic framework for performing these calculations is vital to the success of the approach.

The success of the high-throughput approach is evident from the numerous fruitful studies and discoveries already made. These include the discovery of materials candidates for topological insulators [78], thermoelectrics [85], and piezoelectric material [86]. Verification of such predictions by experimental research has not yet been accomplish.

Many materials problems find themselves well beyond the scope of DFT-based methods. This could be because exploring the finite-temperature properties of a material, something beyond the scope of time-independent DFT, is required or simply because the search space of interest is enormous. For example, exploring all derivative superstructures of a parent lattice can eas-

ily involve consideration of millions of crystal structures, something well beyond the scope of computationally-costly DFT-based approaches.

One way to approach such problems is to use a handful of DFT data to build a model, which is much simpler mathematically. The model can typically compute much faster, making it well-suited for performing large, exhaustive searches over many crystal structures. Additionally, thermodynamic simulations, which can require millions of energy calculations, become accessible once an accurate, fast model is constructed. One model commonly used to explore substitutional order in materials is the cluster expansion, which explores all crystal structures whose atoms lie on the sites of a common, parent lattice. The cluster expansion expresses the energy of an atomic configuration as a sum of contributions from localized clusters of atoms, and can compute the energies of millions of atomic configurations in only minutes.

Mathematically, the cluster expansion can be expressed as:

$$E(\sigma) = \sum_{a} \sum_{(s)} J_a^{(s)} \bar{\Pi}_a^{(s)}(\sigma) \tag{6.1}$$

Here σ represents any atomic configuration restricted to live on the parent lattice. The $\bar{\Pi}_a^{(s)}$ are the basis functions and the $J_a^{(s)}$ are the expansion coefficients. From a practical perspective, the problem of constructing a cluster expansion is a linear algebra problem:

$$\bar{\Pi}\mathbf{J} = \mathbf{E} \tag{6.2}$$

with the matrix $\bar{\Pi}$ containing the values of the cluster functions (columns) over a set of training data (rows), and the vector $\mathbf{E}$ containing the energies of the training set structures. The vector $\mathbf{J}$ contains the sought-after model coefficients and obtaining their values is the main goal when constructing the model.

Previously, the inclusion of materials models, like the cluster expansion, in high-throughput databases has not been possible. This is mostly because the model building process has not been automatic, and instead required the user to spend many hours constructing a high-quality model for a single system. This high human-time cost associated with constructing a CE model stems from two long-standing challenges. First, the terms in the cluster expansion must be truncated to a finite number. This is challenging because there is no way to know *a priori* which terms, from a

candidate pool of thousands, will contribute significantly to the model for a given system. Second, a set of crystal structure to be used for training data must be selected. Since the information content for a crystal structure is dependent upon the chosen basis, choosing an optimal set of training data is not independent of the choice of clusters used.

Various techniques exist for addressing these challenges [24–29, 51–54], but the paradigm for all of them is essentially the same: (i) Heavily truncate the expansion using physical intuition and/or an algorithm so that the matrix $\bar{\Pi}$ is either fully determined or over-determined. (ii) Solve the resulting linear algebra problem by inverting the matrix $\bar{\Pi}$ or through singular value decomposition or related techniques. Many methods for truncating the model have been proposed but none provide a robust, efficient way to guarantee the inclusion of all relevant terms. Furthermore, most modern methods are human-time intensive and/or computationally costly.

Since the number of basis functions, $\bar{\Pi}_a^{(s)}$, that *may be relevant* is much larger than the number of first-principles data points that is feasible to calculate for a single system (thousands vs. hundreds), the natural form of equation (6.5) is the under-determined form. In this case, the number of basis functions considered, M, is much greater than then number of training data available, N ($N \ll M$). However, without a well-defined constraint, solving an under-determined system is impossible since there are an infinite number of solutions consistent with the data.

A recently-emerged technique from the signal processing community, compressive sensing (CS), provides a robust, efficient method for solving the heavily under-determined problem presented by the cluster expansion. CS solves the under-determined problem by constraining the solution search to those with the fewest number of non-zero components. A "minimal-component" model is expected to be of high quality because long-standing physical intuition says that the physical properties of a material are governed by relatively few interaction types. Mathematically, CS seeks to minimize the ℓ_1 norm of the solution vector, a quantity that has long been used to enforce sparsity:

$$\min_{\mathbf{J}} \{ \|\mathbf{J}\|_1 : \bar{\Pi}\mathbf{J} = \mathbf{E} \} \tag{6.3}$$

The CS paradigm solves the cluster selection problem by including essentially all possible clusters, and guarantees the recovery of a sparse solution from a small set of training data. Further-

more, CS provides a well-defined recipe for choosing training data by placing a requirement on the form of the sensing matrix. The CS paradigm is backwards from all other modern CE-construction methods. Other methods start with a small set of training data, and fitting to this data produces a very simple, yet poorly-predicting model. The complexity of the model continually increases as more and more data is added to the training set. In contrast, CS begins with all models consistent with the training data, and throws out all but the simplest, most physical model.

CS provides a robust, efficient, and parameterless (automatic) framework for constructing highly-accurate cluster expansion models. Instead of needing days or weeks to construct a model, CS constructs them in seconds. Instead of requiring lengthly iterative processes for building up a training set (with costly first-principles calculations required at each iteration), CS generates one set of training structures at the outset. When combined, these two key qualities of CS enable the inclusion of CE models in high-throughput databases.

6.2 Large-scale construction of cluster expansion models

Constructing CE models for virtually all binary alloys is a huge endeavor, requiring millions of cpu hours. Since the endeavor is so large and will require millions of DFT calculations, it is critical to ensure that all aspects of the process are efficient, accurate, and well thought out before embarking. Some logical questions that need answering include the following: (i) What is the best way to choose training structures? (If we get this one wrong then we will waste millions of cpu hours) (ii) When performing DFT calculations, what k-points scheme should be used to minimize systematic errors and ensure a highly accurate result? (iii) Is the orthogonality of the unit cell vectors important to the quality of the result? The following is a discussion of some of these questions.

6.2.1 Training set selection

The theorems that form the foundation of CS guarantee the recovery of a sparse solution from a small number of training structures. To make this possible, CS requires that the function samples must be incoherent. This is essentially a requirement on the form of the matrix $\bar{\Pi}$, and is key to defining a well-defined recipe for choosing the best set of training data. Whereas other modern cluster expansion construction techniques iteratively add to the training set, each addition

being determined by the current-iteration fit, the CS paradigm allows an optimal set of training data to be assembled *once* without future modification or augmentation. Since this recipe is going to be used to choose training structures for thousands of binary system, ensuring that it is correct and efficient is of high importance.

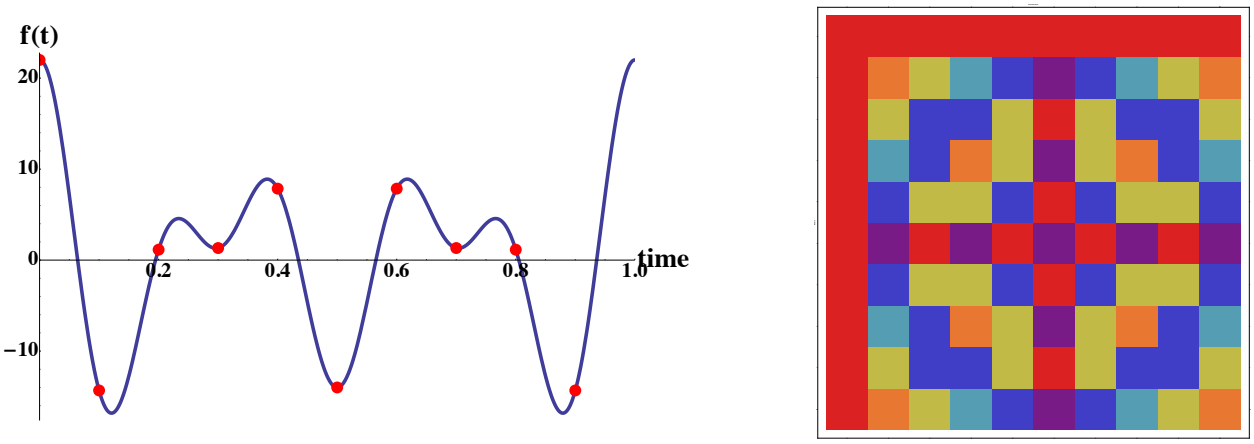


Figure 6.1: Simple one dimensional function representing a signal in time. The red dots are regular samples according the Nyquist's theorem. The figure on the right is a plot of the sensing matrix $\mathbb{A}$ from Eq. (6.5)

To illustrate the concept of coherence, let's first consider the simple one dimensional function given in figure 6.1. In traditional Fourier analysis, this function is expressed as a linear combination of Fourier functions:

$$f(t) = \sum_n a_n \exp(-i2\pi nt) \tag{6.4}$$

and the coefficients are found by solving the linear problem:

$$A\mathbf{x} = \mathbf{b} \tag{6.5}$$

where the matrix $\mathbb{A}$, sometimes called the sensing matrix, contains the chosen Fourier basis functions evaluated at a set of sample points. According to Nyquist's theorem, the function must be sampled regularly and at a rate equal to twice the maximum frequency component present in the

signal. For the function found in figure 6.1 no frequency component greater than 5 Hz is present, and therefore the function was sampled at a rate of 10 Hz, indicated by the red dots.

A plot of the matrix $\mathbb{A}$ for this set of function samples is given in figure 6.1. This is a 10 x 10 matrix where each row corresponds to a sample location, and each column corresponds to a fourier cosine function. The symmetry of the matrix illustrates that the information content in the samples chosen is low. Specifically, note that there is no way to distinguish basis functions $1-5$ from $5-10$ based on this set of function samples. If the function of interest contained frequencies higher than 5 Hz, the solution to equation (6.5) would not result in successful retrieval of the function. The fact that a set of function samples gives redundant information about basis functions is called aliasing and is illustrated in figure 6.2. Inclusion of yet more Fourier basis functions would be futile as the sampling scheme employed would not produce any new information about those basis functions either.

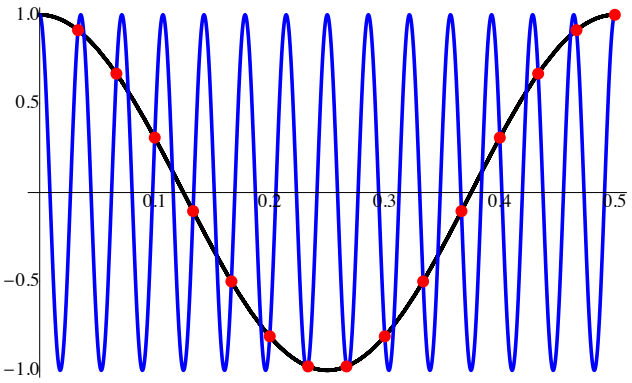


Figure 6.2: The red dots indicate regular samples over the function domain. The blue (black) curve is a cosine function with frequency 2 (28) Hz. Due to the sampling rate employed here, the information contained in the samples is redundant between the two basis functions shown. This is known as aliasing.

The function sampling method put forth by Nyquist was designed to extract only the information content needed and no more. This is effective and efficient for problems where the relevant basis functions are known and all that is lacking is their associated coefficients. However for many problems of practical interest, such as the cluster expansion, the relevant basis functions cannot be easily identified and only a large pool of contenders can be constructed. In such cases it is desirable to maximize the information content in the function samples. This is the objective of the CS requirement that the function samples be "incoherent".

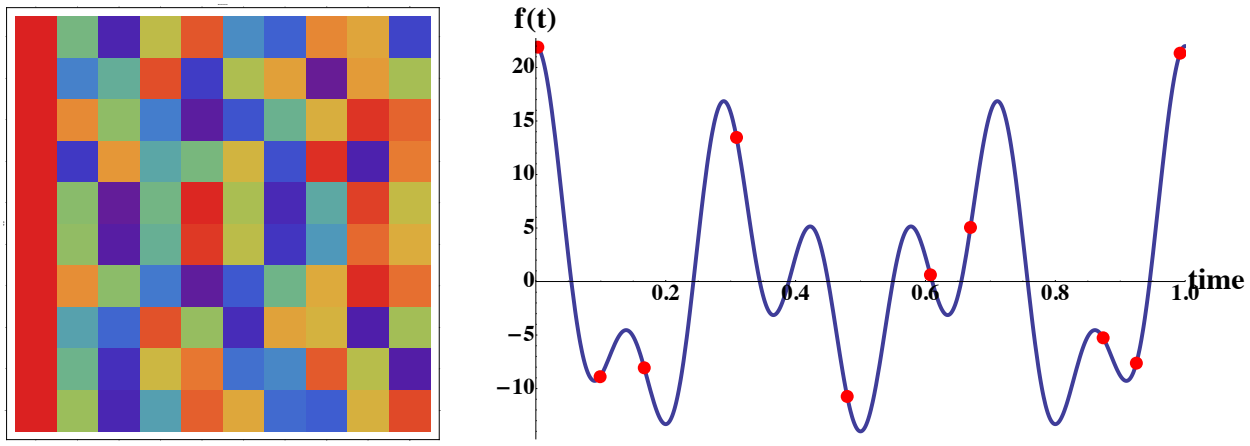


Figure 6.3: Sensing matrix constructed by sampling the function at random locations in its domain. The function on the right contained frequencies: 3, 4, and 7. Recovery of this signal with a 10 x 10 sensing matrix would not be possible with standard Fourier transform tequniques. However, by ensuring that the entries in the sensing matrix are random, the signal can be recovered exactly.

One proposed way to achieve a high level of incoherence is to endeavor to construct the matrix $\mathbb{A}$ such that its entries are randomly drawn numbers on a uniform distribution. [62] This can be a challenge depending on the range of the basis functions. In figure 6.3 is shown a function which contains frequencies greater than 5 and a sensing matrix that was constructed by choosing the sample *locations* randomly from a uniform distribution. To recover this signal using traditional Fourier techniques would require knowing the highest frequency present in the function and adjusting the sample rate in accordance with Nyquist's theorem. However, by ensuring that the sensing matrix is composed of uniformly-distributed random numbers allows the full frequency spectrum of the function to be retrieved.

For the cluster expansion model, constructing the matrix $\bar{\Pi}$ such that its entries are random draws from a uniform distribution is more challenging than the fourier example. This is because the cluster function values form a discrete, non-uniformly distributed set. Figure 6.4 gives the distribution of values for the first, second, and third nearest neighbor cluster functions for all fcc-derived superstructures with 12 atoms/cell or less. Clearly the allowed values are not uniformly distributed, and there are also values which never occur. Furthermore, the values of the cluster functions are correlated to one another, further complicating the task of choosing training data.

One method for choosing training structures which produce an approximately random sensing matrix was given in reference [1]. In that method, vectors of uniformly distributed numbers

were first normalized (i.e. random vectors on a hypersphere) and the structure whose vector of cluster functions was closest to this vector was added to the training set. The metric used for measuring the distance between two vectors was the norm of the difference of the two vectors.

$$\sqrt{\mathbf{v}_1 \cdot \mathbf{v}_2} \tag{6.6}$$

Another method for accomplishing this involves orthonormalizing the random vectors before matching them to real crystal structures. The exact recipe for doing this proceeds as follows:

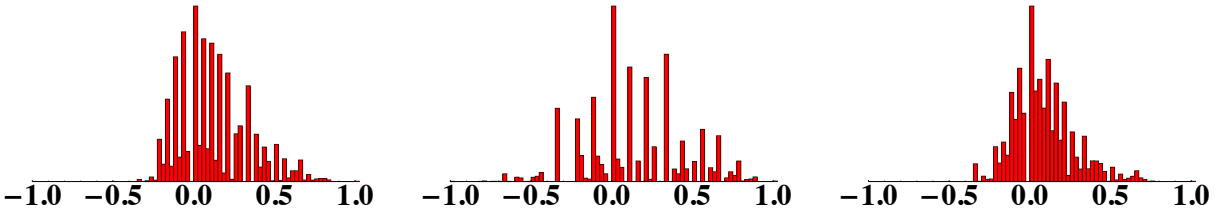


Figure 6.4: Histograms of the value of the 1st, 2nd, and 3rd, nearest neighbor pair cluster functions over all fcc-derived superstructures up to 12 atoms/cell. Most noteworthy is the fact that the cluster function values are not uniformly distributed. Also, note that there are regions of values which never occur over this set of structures. These points make it challenging to construct a sensing matrix composed of random, uniformly distributed entries.

Structure selection procedure

Begin with zero training structures

- Generate a random vector π on the unit hypersphere.
- Orthogonalize π to an orthogonalized version of the current sensing matrix $\bar{\Pi}$.
- Normalize π
- Find the nearest crystal structure to the orthonormalized π.
- Add the structure to the training set.
- Update the matrix $\bar{\Pi}$.

To investigate which method for picking training data results in the most incoherent, or uncorrelated set of data, several approaches were compared:

- The approach defined in this work.
- The approach of reference 1
- Picking structures randomly.

Randomly picked structures were chosen by simply choosing a random integer from 1 to M where M is the number of candidate training structures. The quality of each set of training structures was measured by computing the cross correlation between structures. This is accomplished by taking the dot product of each structure's cluster functions (row in matrix $\bar{\Pi}$) with every other structure in the training set. For vectors that are very close to one another, the dot product will be close to one, while vectors which are nearly orthogonal will be close to zero. For N training structures, this results in an N x N matrix, the off-diagonal entries of which give a measure of how correlated the set of structures are.

The results of this comparison is given in figure 6.5. The matrices depicted on the left hand side of the figure are the cross-correlation matrices, and the histograms on the right give

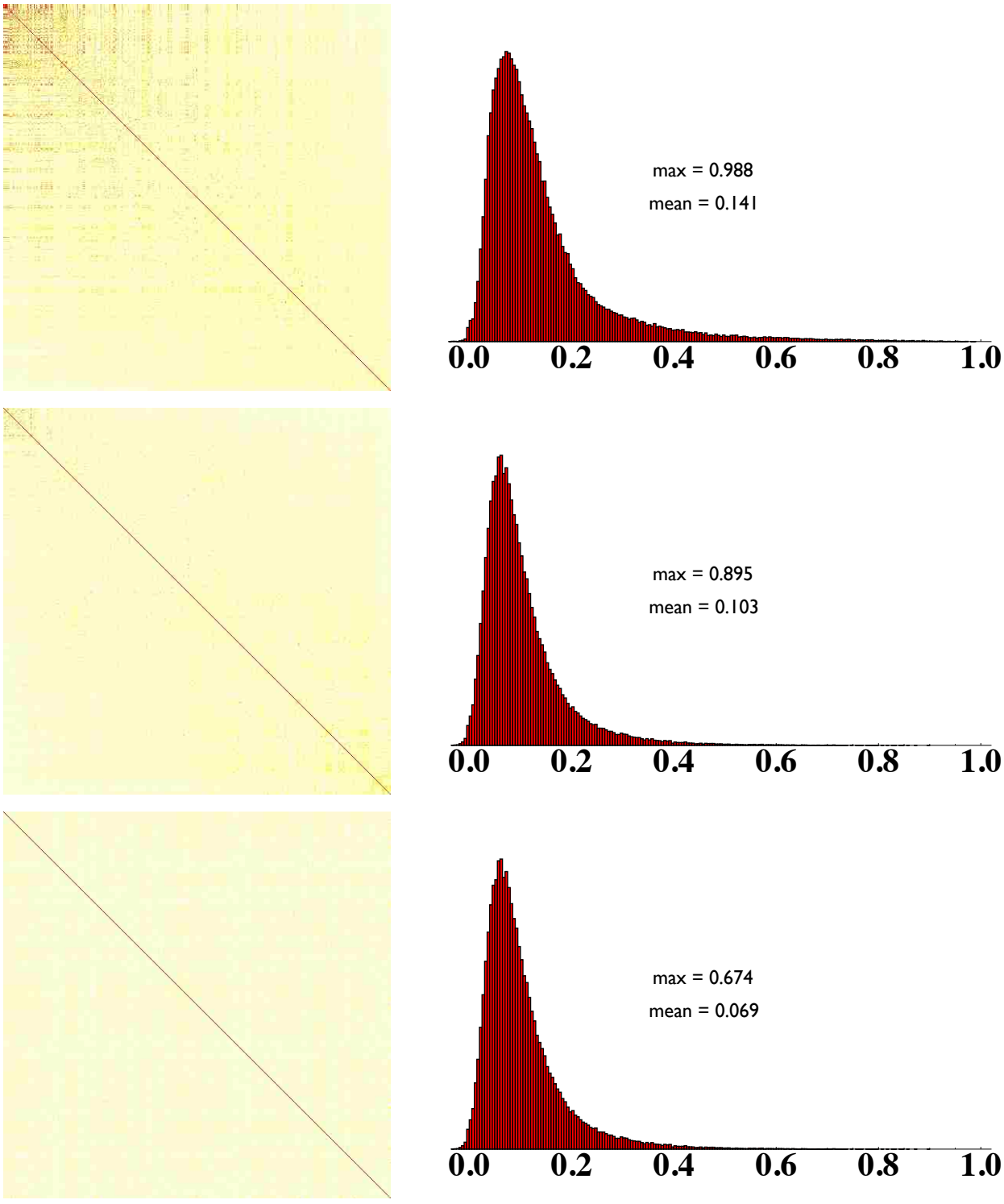


Figure 6.5: Comparison of three different training set selection methods: choosing structures at random (top), the method of reference 1 (middle), and the method discussed in this chapter (bottom). The matrices depicted on the left are cross correlation matrices, and the off-diagonal terms are indicative of how correlated structures are to one another. The histograms on the right show the distribution of off-diagonal cross correlation values. The method described in this work yields lower off-diagonal cross correlations and therefore lower-coherence sets of training structures.

the distribution of the off-diagonal cross correlation values for each structure selection method. Clearly, choosing structure numbers at random leads to the poorest cross-correlation values. A small improvement can be achieved using the method of reference 1, as demonstrated by the fact that the maximum and mean ODCC are slightly lower. Further improvement is achieved with the method put forth in this chapter, and this method will be employed for the current high-throughput work.

For a given lattice type, the set of training structures need not be different between choices of constituent atoms. For example, the training set for an fcc Ag-Pt binary system can be the same set used for Cu-Au on the same lattice. This is analogous to constructing a Fourier expansion for two different functions over the same domain. Although the functions being estimated are different, the locations in the domain where the function is sampled need not be different for the two functions. This means that sets of training structures can be constructed once at the outset for each lattice system that will be considered. First-principles calculations for this set of structures can be performed, using the high-throughput framework, for all binary systems of interest.

6.2.2 Choice of *k*-points

DFT calculations require the evaluation of Brillouin zone integrals, which are typically approximated using numerical techniques. In the case of a periodic configuration of atoms, we can restrict our numerical integration to be over the repeating unit in reciprocal space, or the first Brillouin zone. Numerically evaluating these integrals requires that we first construct a grid of points inside the first Brillouin zone. The single-electron Schrödinger-like equations are then solved at each of these points and the results are used when evaluating the integral. This set of points used in Brillouin zone integration is sometimes referred to as *k*-points.

Two of the most common methods used to construct the *k*-points grids are the Monkhorst-Pack [18] (named after Hendrick J. Monkhorst and James D. Pack, developers of the method) and the equivalent scheme suggested by Froyen. [19] The Monkhorst-Pack scheme subdivides each reciprocal lattice vector into a specified number of divisions. The divisions are chosen such that the resulting mesh is uniform. An example of the Monkhorst Pack *k*-points scheme is given in figure 6.6. The figure on the left shows a rectangular reciprocal unit cell whose reciprocal lattice vectors have been divided into 4 and 3 divisions. The figure on the right shows a hexagonal reciprocal unit

cell whose reciprocal lattice vectors have been divided into 4 divisions. In each case, the specific geometry of the reciprocal unit cell dictated the mesh chosen.

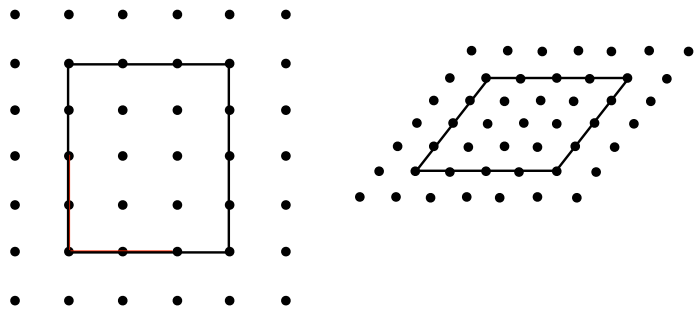


Figure 6.6: Illustration of the Monkhorst-Pack scheme for choosing k-points. Rectangular (left) and triangular (right) reciprocal unit cells are shown. For the rectangular unit cell, the k-points mesh is defined by dividing one lattice vector into 4 divisions and the other into 3, creating a mesh of uniform density. For the triangular unit cell, the mesh is defined by dividing both lattice vectors into 4 divisions.

The equivalent method was suggested by Froyen in cases where the comparison of two energy calculations is to be considered. For example, the formation enthalpy of a binary mixture of two elements is given by

$$H_{\text{formation}} = E_{\text{alloy}} - (E_{\text{A}} x_{\text{A}} + E_{\text{B}}(1 - x_{\text{A}})), \tag{6.7}$$

where E_{AB} is the energy per unit cell of the mixture configuration, N_{AB} is the number of atoms in the unit cell of the mixture, and x_A is the concentration of atom type A in the mixture. The formation enthalpy is a quantity of fundamental importance as it determines the energetic stability of a mixture. This calculation will require three first principles calcuations to be performed, and Froyen suggests that using the same mesh for all three calculations will result in a cancelation of systematic error and therefore a lower overall error in the formation enthalpy of the mixture. Under the "equivalent" scheme for generating k-point meshes a set of vectors in reciprocal space are first defined. The mesh is constructed by adding multiples of these vectors together. The chosen mesh must be commensurate with the reciprocal unit cell. An illustration of the equivalent scheme for constructing k-points meshes is shown in figure 6.7.

Choosing an optimal set of k-points is of high importance for high-throughput database construction. Choosing a very dense mesh can increase the computational burden, and may not

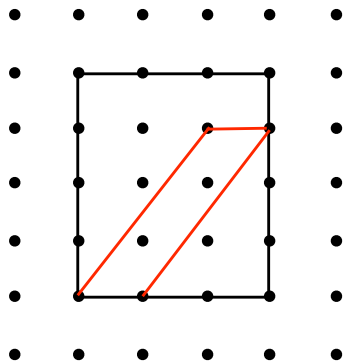
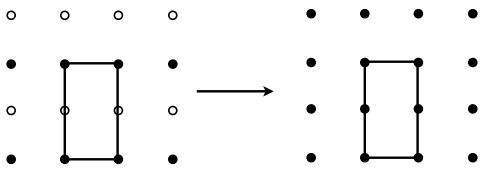


Figure 6.7: Illustration of Froyen's equivalent scheme for choosing *k*-points. The black dots indicate the *k*-points mesh with the black and red polygons being reciprocal unit cells commensurate with the mesh chosen. The mesh of *k*-points shown will be used for both reciprocal unit cells depicted here. Using the same *k*-points mesh is theorized to reduce systematic error.

result in a meaningful increase in accuracy. However, a poor choice here can result in a low-quality result which will be of little or no use. For these reasons, we felt it important to investigate the systematic errors due to *k*-points choices. One way to determine the error associated with the choice of *k*-points for a single unit cell geometry is given as follows:

- Consider a binary mixture of two atomic types, possibly a derivative superstructure of an fcc, bcc, or hcp lattice.
- Replace atoms of type B with atoms of type A. This results in a lattice of all A atoms, but uses a much larger unit cell than what is necessary. This tricks the DFT code into thinking it is computing the energy of a large unit cell crystal structure, when it is actually only computing the energy of a lattice of A atoms.



- Compute the energy of this configuration.
- Compute the energy of a lattice of A atoms using the primitive unit cell.

- Compute the difference in energy from these two calculations. Any differences in these two energies represents systematic numerical errors which can be attributed to the k-points scheme used.

This procedure was carried out for three schemes for choosing k-points: MP with a mesh density of 6000 KPPRA and 1000 KPPRA and the equivalent scheme proposed by Froyen. Unit cells corresponding to fcc, bcc, and hcp-derived superstructures were used. The results of this comparison is given in figure 6.8.

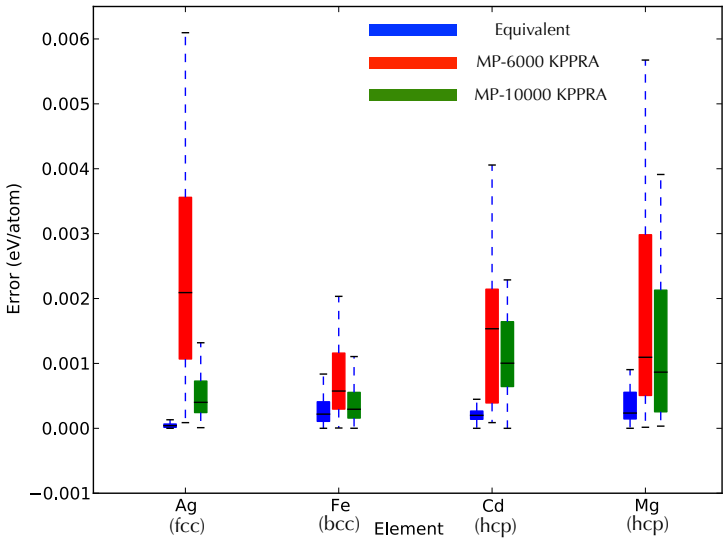


Figure 6.8: Comparison of systematic errors associated with the choice of k-point meshes. Three methods for choosing k-point meshes are depicted: MP with a density of 600 KPPRA, and 10,000 KPPRA and the equivalent scheme of Froyen. Clearly the equivalent scheme results in smaller systematic error. However for high enough densities the MP method appears to be sufficiently accurate.

Clearly, the equivalent scheme results in the lowest level of error for all lattice systems and atomic types depicted here. The low-density MP schemes yield relatively high levels of error. Increasing the density to $10,000$ KPPRA lowers the error considerably. For Ag (fcc) and Fe (bcc) the error level for MP 10000 KPPRA is sufficiently low as to not produce concern. The level of error present in the hcp systems remain slightly higher for the MP 10000 KPPRA scheme. The reason for this is unknown to the authors.

Implementation of the equivalent scheme introduces some unwanted book-keeping complexities and tasks. For each mixture calculation performed, a pure A (B) calculation must be

performed using the same k-points mesh used for the mixture. Since the MP $10,000$ KPPRA errors are not significant, this scheme for generating k-points will be used to avoid such complexities.

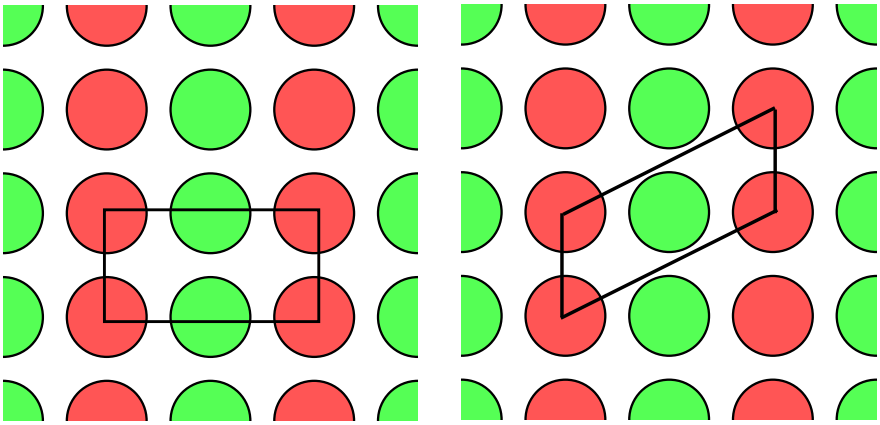


Figure 6.9: Illustration of two different choices of unit cell for the same 2D atomic configuaration. The unit cell on the left has the shortest, most orthogonal lattice vectors. The unit cell on the left provides a perfectly correct description of the crystal, but this choice of unit cell should not be used in DFT calculations.

Another source of error in DFT calculations is the shape of the unit cell. For any given atomic configuaration, there are an infinite number of unit cells that can be used to represent the crystal structure. For example, figure 6.9 illustrates two choices for representing the same 2d atomic configuration: vertical stripes. The choice on the left represents the shortest, and most orthogonal choice, and is the recommended choice for DFT calculations. While the choice on the right is a perfectly correct unit cell choice, it's lattice vectors are longer, and less orthogonal than the choice on the left. Althought the exact reason is not known, it is well known that for DFT calculations the shortest, or most orthogonal, set of lattice vectors should be used.

The exact origin of the error associated with the unit cell shape is not known. A large-scale quantifying investigation of this effect will not be given here. Instead we simply point out that when performing DFT calculations, the unit cell vectors ought to be as short, or as orthogonal, as possible. This can be done using a variety of algorithms in the category of lattice basis reduction. For the current work, the unit cells for all calculations will be reduced using an algorithm put forth by Nguyen and Stehle. [72]

BCS-based CE models for four binary systems

System	No. holdout str.	No. fitting str.	rmse over holdout set (meV/atom)	$\|J\|_1$ (meV/atom)	$\|J\|_0$
Ag-Pt	300	100	4.436	293.2	49.04
		200	3.578	322.4	59.85
		300	3.117	354.2	42.57
		400	2.750	386.9	52.65
		500	2.607	397.9	57.08
		600	2.519	410.9	61.82
		700	2.448	420.7	66.88
Ag-Al	85	100	8.115	352.0	59.89
		200	4.604	433.9	99.48
		300	3.910	480.3	89.55
Cu-Pt	150	100	5.750	397.1	45.74
		200	4.178	439.9	40.11
		300	3.510	486.2	38.63
		400	3.133	515.9	45.73
		500	2.928	536.4	50.88
		600	2.795	551.5	55.77
Ag-Pt	500	100	2.893	201.2	40.7
		200	2.438	220.2	27.6
		300	2.122	240.4	35.09
		400	1.965	249.6	41.39
		500	1.874	254.8	45.48
		600	1.810	260.1	49.82
		700	1.766	262.7	52.75

Table 6.1: Model-quality results for the binary systems Ag-Pt, Ag-Al, Cu-Pt, and Ag-Pt on an fcc lattice. The rms, $\|J\|_1$, and $\|J\|_0$ are averages over 100 different choices of training set.

6.3 Lattice models for all binary alloys: results from a few select systems

To illustrate the feasibility of automatically constructing CE models for many binary systems, we have chosen to exhibit four systems: Ag-Al, Ag-Pd, Ag-Pt, and Cu-Pt. Pt group metal alloys are of high importance for catalysis and jewelry, motivating their study here. Furthermore, CE models for these systems have been constructed previously using other methods, and a comparison to these results was wanted.

For each system, between 800 and 1300 data points were generated. From this large data set, up to 800 were put in a pool to be used for training data, with the rest being held out and used to validate the model. From the pool of training data, subsets of up to 700 data points were

selected to be used to train the model. One hundred different choices of training set were chosen, and results from each of the 100 models were averaged over.

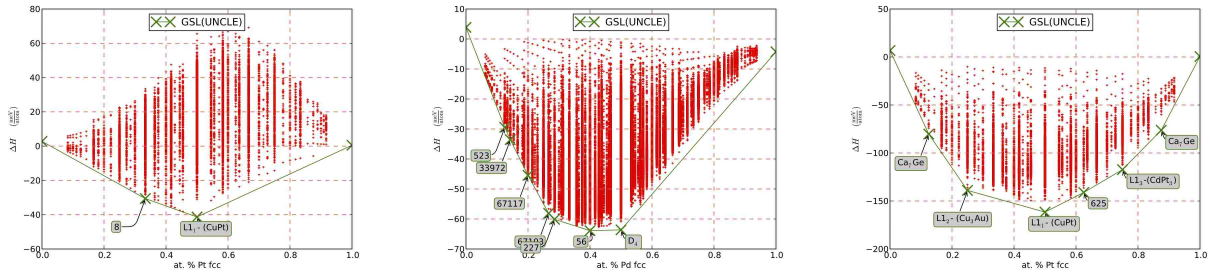


Figure 6.10: Ground state search over all fcc-derived superstructures up to 12 atoms/cell for the binary system Ag-Pt. The green line is the convex hull and indicates the ground states of the system.

The quality of the fit was measured by using the model to predict the energies of all crystal structures in the holdout set. The root mean square (RMS) of the error in these predictions was then calculated:

$$E_{\text{rms}} = \sqrt{\frac{1}{N}\sum_{i}^{N}(E^{(\text{DFT})}(\sigma_i) - E^{(\text{Pred})}(\sigma_i))^2} \tag{6.8}$$

Other fit-quality indicators provided are the ℓ_1 and ℓ_0 norms of the solution vector. The ℓ_0 norm indicates how many basis functions were used in the model, and the ℓ_1 norm indicates whether large-magnitude coefficients are being added to the model as more training data is used. Steady, and large increases in the ℓ_1 norm indicate a lack of model convergence. A summary of the results for these four systems is given in table 6.1

For every system explored here, the rmse over the holdout set reaches very low values (~ 2 meV/atom) as the number of training structures increases. The ℓ_1 and ℓ_0 norm also exhibit a convergence of the solution. The weighted BCS framework for constructing the models is automatic and efficient, requiring only minutes to construct model for 100 different choices of training data.

The results of CE models for all binary alloys and for all relevant lattice types will soon be made available to the scientific community. This will include fit-quality statistics, like those shown

in table 6.1, ground-state-search results, like those shown in figure 6.10, and a data file containing the model details (see Figure 6.11).

Figure 6.11 is a small snippet of the file containg the details of the model. This file provides the coordinates for the vertices of the cluster as well as the Chebychev, or so-called "point" functions, that are to be evaluated on those vertices, and the coefficient asssociated with the cluster function. The center-of-mass distance, or average distance from the center-of-mass to the cluster vertices, is also given and provides a measure of the spatial extent of the cluster. With the contents of these files the model can be reconstructed and used to perform thermodynamic simulations, custom searches over sets of derivative-superstructures, or other useful things.

```
#-------------------------------------------------------
# Cluster number: in this list | original in clusters.out
     7          8
# J (coefficient value in meV)
   4.71184838529987
# Number of vertices
  2
# Average Distance
  0.86602500
# Damping
  0.00000000
# Vertices: (x,y,z) | d-vector label | s-index
 -1.00000000    1.00000000    1.00000000         1     1
  0.00000000    0.00000000    0.00000000         1     1
#-------------------------------------------------------
# Cluster number: in this list | original in clusters.out
     8         15
# J (coefficient value in meV)
   2.80288524645545
# Number of vertices
  2
# Average Distance
  1.22474500
# Damping
  0.00000000
# Vertices: (x,y,z) | d-vector label | s-index
  0.00000000    0.00000000    0.00000000         1     1
  1.00000000    1.00000000   -2.00000000         1     1
#-------------------------------------------------------
```

Figure 6.11: This is a snippet of the file which provides a list of all relevant cluster functions and their coefficients. Information provided in this file includes the coordinates of the cluster vertices, point functions to be evaluated on each cluster vertex, and the associated model coefficient. Soon models for hundreds of alloy systems will become available to the general public.

6.4 Summary and Outlook

Automatic construction of large databases of materials information is becoming a valuable asset to the materials science community. They require litte human time to construct and provide

valuable insight and direction in the search for new, high-performing materials. Until now, these databases have consisted of DFT calculations only, leaving out valuable materials models because it was not automatic to construct them.

Due to the discovery of CS as an automatic, efficient, and robust way to construct CE models, the inclusion of materials models in materials databases is now very feasible. The CS paradigm allows very accurate models to be constructed very efficiently and using a reasonable amount of data. Essentially, once the DFT data is available, the construction of the model is instantaneous and effortless. The work to construct CE models for all relevant (not phase separating) binary systems has begun, and these models will soon be available to the scientfic community. A database of ternary CE models is set to begin following the binary model database.

Mass production of materials models is a novel endeavor and represents a significant stride in materials research. Soon, instead of spending days or weeks to construct a reliable model, materials scientist can visit the database and gain access to a model for the system they are interested in. Using the model, they can perform large, exhaustive searches, perform thermodynamic simulations to extract important, finite-temperature properties of the material and other things.

Chapter 7

Conclusion

7.1 Summary

In the arena of materials science, computational research is becoming a valuable partner to experimental studies. This is due in part to theoretical strides that have made finding the ground-state properties of a collection of atoms possible, and in part to computational advances that have made such calculations computationally viable. One mainstay of computational materials science is density functional theory which provided an alternative to solving the many-body Schrödinger equation. This single theoretical stride enables the accurate calculation of the physical properties of materials of all kinds.

The cluster expansion is a fast, accurate model that has shown itself to be very useful in materials research. It's computational speed allows the exploration of millions of derivative superstructures in only minutes. It also makes thermodynamic simulations, such as Monte-Carlo, metadynamics, and Wang-Landau feasible. However, choosing how to truncate the expansion and which crystal structures to use for training data have been difficult challenges to overcome. Modern techniques for addressing these challenges have fallen short in their ability to address these challenges, and usually require complex algorithms and many user hours. This has prevented the inclusion of CE models in large-scale materials databases. Chapter 3 discussed the cluster expansion in more detail and explained the challenges associated with constructing one in practice.

Compressive sensing, a newly emerged technique from the signal processing community solves, in robust mathematical fashion, the most glaring challenges in the CE community, making the construction of CE models fast and automatic. Chapter 4 of this dissertation discussed the mathematical foundation of CS as proposed by Candes. *et. al* and provided a mathematical recipe for implementing the CS paradigm (put forth by Yin *et. al*). A comparison between this implementation of CS and the direct optimization approach was made for two systems of practical interest:

the binary system: Ag-Pt, and a protein folding problem. The quality of the model was found to be much better than the DO approach and the time needed to construct the model was much shorter (hours vs. days). This chapter was published in the Physical Review B in February 2013.

Chapter 5 discussed a Bayesian implementation of CS, which offers a parameterless framework, error bars on predictions made and vast speed increases over current state-of-the-art methods. Bayesian CS was used to construct models for three binary metallic: Ag-Pt, Ag-Pd, and Cu-Pt. A comparison to the genetic algorithm was used as a way to compare against currently-used methods. The accuracy of the models was measured by predicting the energies of a holdout set of structures. In each case, BCS-based CE models were more physical in nature and predicted more accurately than the GA method. Furthermore, the time required to construct the model was vastly different, with the GA requiring 24 hours for a single fit and BCS needing only minutes to construct a hundred fits. This chapter will soon be submitted to a reputable scientific journal for review and we expect it will be published soon after.

The speed, accuracy, and automatic nature of CS-based CE models makes including them in large-scale materials databases feasible. Hitherto this has not been possible mostly because constructing a materials model, such as the cluster expansion, has not been automatic. Chapter 6 of this dissertation discussed this endeavor and the challenges it faced as well as some example system for illustration.

7.2 Outlook

Previously, the construction of lattice models, such as the cluster expansion required days or weeks of parameter tuning, algorithmic iterations, and first-principles calculations. For this reason, large-scale construction of lattice models for many binary or ternary systems was not feasible. However, the content of this dissertation illustrates that CS is an efficient, robust, and automatic tool for building accurate CE models, making the model-building process effortless and virtually hands-off. Chapter 6 of this dissertation discussed how CS constructed CE models for all intermetallic binary alloys will be automatically constructed using an automatic framework. No doubt this will be a multi-year endeavor, with lattice models becoming available steadily over that time. Soon lattice models for all interesting binary alloy systems will be available to the entire scientific

community. Such a feat has not been accomplished previously and will no doubt be a great benefit to the community at large.

Soon after lattice models for all binary systems are completed, work will commence on lattice models for all interesting ternary systems. The number of ternary systems of interest is much larger and therefore the project will take considerably longer.

From a broader perspective, this dissertation has proven compressive sensing to be a viable theoretical tool for scientists of all disciplines. Compressive sensing is a promising tool for any highly-underdetermined problem of the form

$$\mathbb{A}\mathbf{x} = \mathbf{b} \tag{7.1}$$

if the solution vector, $\mathbf{x}$, is expected to be sparse. Any problem that can be massaged into this form can be solved efficiently and automatically using compressive sensing. While we have demonstrated CS to be effective for constructing cluster expansion models, we expect that in the future many other important physical problems will be tackled using compressive sensing as the main tool.

Bibliography

[1] Nelson, L. J., Hart, G. L., Zhou, F., and Ozoliņš, V., 2013. "Compressive sensing as a paradigm for building physics models." *Physical Review B,* **87**(3), p. 035125. xiii, 33, 66, 68, 81, 83, 84, 85

[2] Curtarolo, S., Hart, G. L., Nardelli, M. B., Mingo, N., Sanvito, S., and Levy, O., 2013. "The high-throughput highway to computational materials design." *Nature materials,* **12**(3), pp. 191–201. 5

[3] Hohenberg, P., and Kohn, W., 1964. "Inhomogeneous electron gas." *Phys. Rev.,* **136**, Nov, pp. B864–B871. 9

[4] Hart, G. L. W., 1999. "Electronic structure studies of materials properties and stability in transition metal-metalloid compounds." PhD thesis, University of California-Davis. 9

[5] Sholl, D., and Steckel, J. A., 2011. *Density functional theory: a practical introduction*. Wiley-Interscience. 9

[6] Kohn, W., and Sham, L. J., 1965. "Self-consistent equations including exchange and correlation effects." *Phys. Rev.,* **140**, Nov, pp. A1133–A1138. 12

[7] Vosko, S. H., Wilk, L., and Nusair, M., 1980. "Accurate spin-dependent electron liquid correlation energies for local spin density calculations: a critical analysis." *Canadian Journal of Physics,* **58**(8), pp. 1200–1211. 15

[8] Perdew, J. P., and Zunger, A., 1981. "Self-interaction correction to density-functional approximations for many-electron systems." *Physical Review B,* **23**(10), p. 5048. 15

[9] Cole, L. A., and Perdew, J., 1982. "Calculated electron affinities of the elements." *Physical Review A,* **25**(3), p. 1265. 15

[10] Perdew, J. P., and Wang, Y., 1992. "Accurate and simple analytic representation of the electron-gas correlation energy." *Physical Review B,* **45**(23), p. 13244.

[11] von Barth, U., and Hedin, L., 1972. "A local exchange-correlation potential for the spin polarized case. i." *Journal of Physics C: Solid State Physics,* **5**(13), p. 1629. 15

[12] Perdew, J. P., Burke, K., and Ernzerhof, M., 1996. "Generalized gradient approximation made simple." *Phys. Rev. Lett.,* **77**, Oct, pp. 3865–3868. 15, 54, 72

[13] Langreth, D. C., and Perdew, J. P., 1980. "Theory of nonuniform electronic systems. i. analysis of the gradient approximation and a generalization that works." *Physical Review B,* **21**(12), p. 5469. 15

[14] Langreth, D. C., and Mehl, M., 1983. "Beyond the local-density approximation in calculations of ground-state electronic properties." *Physical Review B,* **28**(4), p. 1809. 15

[15] Perdew, J. P., and Yue, W., 1986. "Accurate and simple density functional for the electronic exchange energy: Generalized gradient approximation." *Physical Review B,* **33**(12), p. 8800. 15

[16] Perdew, J. P., 1986. "Density-functional approximation for the correlation energy of the inhomogeneous electron gas." *Physical Review B,* **33**(12), p. 8822.

[17] Vanderbilt, D., 1990. "Soft self-consistent pseudopotentials in a generalized eigenvalue formalism." *Physical Review B,* **41**(11), p. 7892. 19

[18] Monkhorst, H., and Pack, J., 1976. "Special points for Brillouin-zone integrations." *Phys. Rev. B,* **13**(12), pp. 5188–5192. 20, 85

[19] Froyen, S., 1989. "Brillouin-zone integration by Fourier quadrature: Special points for superlattice and supercell calculations." *Phys. Rev. B,* **39**, pp. 3168–3172. 20, 55, 72, 85

[20] Sanchez, J., Ducastelle, F., and Gratias, D., 1984. "Generalized cluster description of multicomponent systems." *Physica A: Statistical and Theoretical Physics,* **128**(1-2), pp. 334–350. 24, 35, 39, 64

[21] Fontaine, D., 1994. "Cluster approach to order-disorder transformations in alloys." *Solid State Physics,* **47**, pp. 33–176. 24, 35, 39, 64

[22] Zunger, A., 1994. *First-Principles Statistical Mechanics of Semiconductor Alloys and Intermetallic Compounds*. NATO Advanced Study Institute on Statics and Dynamics of Alloy Phase Transformations, pp. 361–419. 24, 35, 39, 64

[23] Connolly, J., and Williams, A., 1983. "Density-functional theory applied to phase-transformations in transition-metal alloys." *Physical Review B,* **27**(8), pp. 5169–5172. 29, 40, 64

[24] van de Walle, A., Asta, M., and Ceder, G., 2002. "The alloy theoretic automated toolkit: A user guide." *Calphad,* **26**(4), pp. 539–553. 31, 39, 40, 56, 65, 77

[25] Blum, V., Hart, G. L. W., Walorski, M. J., and Zunger, A., 2005. "Using genetic algorithms to map first-principles results to model Hamiltonians: Application to the generalized Ising model for alloys." *Phys. Rev. B,* **72**(16), p. 165113. 31, 41, 65, 77

[26] Lerch, D., Wieckhorst, O., Hart, G. L. W., Forcade, R. W., and Müller, S., 2009. "UNCLE: a code for constructing cluster expansions for arbitrary lattices with minimal user-input." *Modelling and Simulation in Materials Science and Engineering,* **17**, p. 055003. 31, 39, 41, 52, 65, 77

[27] Drautz, R., and Díaz-Ortiz, A., 2006. "Obtaining cluster expansion coefficients in ab initio thermodynamics of multicomponent lattice-gas systems." *Physical Review B,* **73**(22), p. 224207. 32, 41, 65, 77

[28] Díaz-Ortiz, A., Dosch, H., and Drautz, R., 2007. "Cluster expansions in multicomponent systems: precise expansions from noisy databases." *Journal of Physics: Condensed Matter,* **19**(40), p. 406206. 32, 41, 65, 77

[29] Müller, T., and Ceder, G., 2009. "Bayesian approach to cluster expansions." *Phys. Rev. B,* **80**(2), p. 024103. 32, 33, 36, 39, 41, 65, 66, 77

[30] Cockayne, E., and Van De Walle, A., 2010. "Building effective models from sparse but precise data: Application to an alloy cluster expansion model." *Phys. Rev. B,* **81**(1), p. 012104. 32, 33, 36, 39, 66

[31] van de Walle, A., and Ceder, G., 2002. "Automating first-principles phase diagram calculations." *J. Phase Equilibria,* **23**(4), pp. 348–359. 33, 65

[32] Seko, A., Koyama, Y., and Tanaka, I., 2009. "Cluster expansion method for multicomponent systems based on optimal selection of structures for density-functional theory calculations." *Phys. Rev. B,* **80**(16), p. 165122. 33, 48, 65

[33] Zunger, A., 1980. "Systematization of the stable crystal structure of all ab-type binary compounds: A pseudopotential orbital-radii approach." *Phys. Rev. B,* **22**, pp. 5839–5872. 35

[34] Villars, P., 1983. "A three-dimensional structural stability diagram for 998 binary ab intermetallic compounds." *Journal of the Less Common Metals,* **92**(2), pp. 215 – 238. 35

[35] Pettifor, D., 1984. "A chemical scale for crystal-structure maps." *Solid State Communications,* **51**(1), pp. 31 – 34. 35

[36] Pettifor, D., 1986. "The structures of binary compounds. 1. phenomenological structure maps." *Journal of Physics C - Solid State Physics,* **19**(3), pp. 285–313. 35

[37] Miedema, A., Boom, R., and De Boer, F. R., 1975. "On the heat of formation of solid alloys." *Journal of the Less Common Metals,* **41**(2), pp. 283 – 298. 35

[38] Körmann, F., Dick, A., Grabowski, B., Hallstedt, B., Hickel, T., and Neugebauer, J., 2008. "Free energy of bcc iron: Integrated *ab initio* derivation of vibrational, electronic, and magnetic contributions." *Phys. Rev. B,* **78**, Jul, p. 033102. 35

[39] Fischer, C. C., Tibbetts, K. J., Morgan, D., and Ceder, G., 2006. "Predicting crystal structure by merging data mining with quantum mechanics." *Nature Materials,* **5**(8), July, pp. 641–646. 36

[40] Schön, J. C., Doll, K., and Jansen, M., 2010. "Predicting solid compounds via global exploration of the energy landscape of solids on the ab initio level without recourse to experimental information." *physica status solidi (b),* **247**(1), pp. 23–39. 36

[41] Munter, T. R., Landis, D. D., Abild-Pedersen, F., Jones, G., Wang, S., and Bligaard, T., 2009. "Virtual materials design using databases of calculated materials properties." *Computational Science & Discovery,* **2**(1), p. 015006. 36

[42] Setyawan, W., Gaume, R. M., Lam, S., Feigelson, R. S., and Curtarolo, S., 2011. "High-throughput combinatorial database of electronic band structures for inorganic scintillator materials." *ACS Combinatorial Science,* **13**(4), pp. 382–390. 36

[43] Woodley, S. M., and Catlow, R., 2008. "Crystal structure prediction from first principles." *Nature Materials,* **7**(12), Dec., pp. 937–946. 36

[44] Jansen, A. P. J., and Popa, C., 2008. "Bayesian approach to the calculation of lateral interactions: No/rh(111)." *Phys. Rev. B,* **78**, Aug, p. 085404. 36, 39

[45] Candès, E., and Wakin, M., 2008. "An introduction to compressive sampling." *Signal Processing Magazine, IEEE,* **25**(2), pp. 21–30. 36, 38, 47

[46] AlQuraishi, M., and McAdams, H., 2011. "Direct inference of protein–dna interactions using compressed sensing methods." *Proceedings of the National Academy of Sciences,* **108**(36), pp. 14819–14824. 36

[47] Candès, E., Romberg, J., and Tao, T., 2006. "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information." *Information Theory, IEEE Transactions on,* **52**(2), pp. 489–509. 38, 41, 67, 68

[48] van de Walle, A., 2009. "Multicomponent multisublattice alloys, nonconfigurational entropy and other additions to the alloy theoretic automated toolkit." *Calphad,* **33**(2), pp. 266–278. 39, 56

[49] Zarkevich, N. A., and Johnson, D. D., 2004. "Reliable first-principles alloy thermodynamics via truncated cluster expansions." *Phys. Rev. Lett.,* **92**, Jun, p. 255702. 39

[50] Mueller, T., and Ceder, G., 2010. "Exact expressions for structure selection in cluster expansions." *Phys. Rev. B,* **82**, Nov, p. 184107. 39, 65

[51] Laks, D., Ferreira, L., Froyen, S., and Zunger, A., 1992. "Efficient cluster-expansion for substitutional systems." *Physical Review B,* **46**(19), pp. 12587–12605. 41, 59, 65, 77

[52] Kurta, R. P., Bugaev, V. N., and Ortiz, A. D., 2010. "Long-wavelength elastic interactions in complex crystals." *Phys. Rev. Lett.,* **104**, Feb, p. 085502. 41, 65, 77

[53] Thuinet, L., and Besson, R., 2012. "New insights on strain energies in hexagonal systems." *Applied Physics Letters,* **100**(25), jun, pp. 251902 –251902–4. 41, 65, 77

[54] Shchyglo, O., Díaz-Ortiz, A., Udyansky, A., Bugaev, V. N., Reichert, H., Dosch, H., and Drautz, R., 2008. "Theory of size mismatched alloy systems: many-body kanzaki forces." *Journal of Physics: Condensed Matter,* **20**(4), p. 045207. 41, 59, 65, 77

[55] Tibshirani, R., 1996. "Regression shrinkage and selection via the lasso." *J. Roy. Stat. Soc. Ser. B,* **58**(1), pp. 267–288. 42

[56] Chen, S., Donoho, D., and Saunders, M., 1998. "Atomic decomposition by basis pursuit." *SIAM J. Sci. Comput.,* **20**(1), pp. 33–61. 42

[57] Hale, E., Yin, W., and Zhang, Y., 2007. "A fixed-point continuation method for l1-regularized minimization with applications to compressed sensing." *CAAM TR07-07, Rice University*. 43, 44

[58] Yin, W., Osher, S., Goldfarb, D., and Darbon, J., 2008. "Bregman iterative algorithms for l1-minimization with applications to compressed sensing." *SIAM Journal on Imaging Sciences,* **1**(1), pp. 143–168. 44, 52, 69

[59] Boyd, S., and Vandenberghe, L., 2004. *Convex Optimization*. Cambridge University Press. 45

[60] Goldstein, T., and Osher, S., 2009. "The split bregman method for l1 regularized problems." *SIAM Journal on Imaging Sciences,* **2**(2), pp. 323–343. 45, 52, 61

[61] Donoho, D., and Huo, X., 2001. "Uncertainty principles and ideal atomic decomposition." *IEEE Transactions on Information Theory,* **47**(7), pp. 2845–2862. 46

[62] Candes, E., and Romberg, J., 2007. "Sparsity and incoherence in compressive sampling." *Inverse problems,* **23**, p. 969. 47, 81

[63] Candes, E., Romberg, J., and Tao, T., 2006. "Stable signal recovery from incomplete and inaccurate measurements." *Communications on pure and applied mathematics,* **59**(8), pp. 1207–1223. 47

[64] Hart, G. L. W., and Forcade, R. W., 2008. "Algorithm for generating derivative structures." *Phys. Rev. B,* **77**(22), p. 224115. 48, 49

[65] Hart, G. L. W., and Forcade, R. W., 2009. "Generating derivative structures from multilattices: Algorithm and application to hcp alloys." *Phys. Rev. B,* **80**(1), p. 014120. 48

[66] Van de Walle, A., Asta, M., and Ceder, G., 2002. "The alloy theoretic automated toolkit: A user guide." *Calphad,* **26**(4), pp. 539–553. 48

[67] Johnstone, I., 2001. "On the distribution of the largest eigenvalue in principal components analysis." *The Annals of statistics,* **29**(2), pp. 295–327. 49

[68] Durussel, P., and Feschotte, P., 1996. "A revision of the binary system ag–pt." *J. Alloys Compound.,* **239**, pp. 226–230. 54

[69] Kresse, G., and Joubert, D., 1999. "From ultrasoft pseudopotentials to the projector augmented-wave method." *Phys. Rev. B,* **59**(3), p. 1758. 54, 71

[70] Kresse, G., and Furthmüller, J., 1996. "Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set." *Comp. Mat. Sci.,* **6**(1), pp. 15–50. 54, 71

[71] Blöchl, P. E., 1994. "Projector augmented-wave method." *Phys. Rev. B,* **50**(24), p. 17953. 54, 71

[72] Nguyen, P. Q., and Stehlé, D., 2009. "Low-dimensional lattice basis reduction revisited." *ACM Trans. Algorithms,* **5**(4), Nov., pp. 1–48. 55, 72, 89

[73] Nelson, L., Hart, G., and Curtarolo, S., 2012. "Ground-state characterizations of systems predicted to exhibit $L1_1$ or $L1_3$ crystal structures." *Phys. Rev. B,* **85**, p. 054203. 55, 72

[74] Holliger, L., and Besson, R., 2011. "Reciprocal-space cluster expansions for complex alloys with long-range interactions." *Phys. Rev. B,* **83**(17), p. 174202. 59

[75] Kurta, R. P., Bugaev, V. N., and Ortiz, A. D., 2010. "Long-wavelength elastic interactions in complex crystals." *PHYSICAL REVIEW LETTERS,* **104**(8), FEB 26. 59

[76] Zhou, F., Grigoryan, G., Lustig, S., Keating, A., Ceder, G., and Morgan, D., 2005. "Coarse-graining protein energetics in sequence variables." *Physical review letters,* **95**(14), p. 148103. 60, 61

[77] Grigoryan, G., Reinke, A., and Keating, A., 2009. "Design of protein-interaction specificity gives selective bzip-binding peptides." *Nature,* **458**(7240), pp. 859–864. 60

[78] Yang, K., Setyawan, W., Wang, S., Buongiorno Nardelli, M., and Curtarolo, S., 2012. "A search model for topological insulators with high-throughput robustness descriptors." *Nat. Mater.,* **11**(7), pp. 614–619. 63, 75

[79] Bloch, J., Levy, O., Pejova, B., Jacob, J., Curtarolo, S., and Hjörvarsson, 2012. "Prediction and hydrogen-acceleration of ordering in iron-vanadium alloys." *Phys. Rev. Lett.,* **108**, p. 215503. 63

[80] Wang, S., Wang, Z., Setyawan, W., Mingo, N., and Curtarolo, S., 2011. "Assessing the thermoelectric properties of sintered compounds via high-throughput ab-initio calculations." *Phys. Rev. X,* **1**, p. 021012. 63

[81] Garbulsky, G. D., and Ceder, G., 1995. "Linear-programming method for obtaining effective cluster interactions in alloys from total-energy calculations: Application to the fcc pd-v system." *Phys. Rev. B,* **51**, Jan, pp. 67–72. 65

[82] Babacan, S., Molina, R., and Katsaggelos, A., 2010. "Bayesian compressive sensing using laplace priors." *Image Processing, IEEE Transactions on,* **19**(1), pp. 53–63. 69, 70, 105

[83] Ji, S., Xue, Y., and Carin, L., 2008. "Bayesian compressive sensing." *Signal Processing, IEEE Transactions on,* **56**(6), pp. 2346–2356. 69, 105

[84] Candes, E., Wakin, M., and Boyd, S., 2008. "Enhancing sparsity by reweighted ℓ_1 minimization." *Journal of Fourier Analysis and Applications,* **14**(5), pp. 877–905. 71

[85] Castelli, I. E., Olsen, T., Datta, S., Landis, D. D., Dahl, S., Thygesen, K. S., and Jacobsen, K. W., 2012. "Computational screening of perovskite metal oxides for optimal solar light capture." *Energy & Environmental Science,* **5**(2), pp. 5814–5819. 75

[86] Roy, A., Bennett, J. W., Rabe, K. M., and Vanderbilt, D., 2012. "Half-heusler semiconductors as piezoelectrics." *Physical Review Letters,* **109**(3), p. 037602. 75

Appendix A

Bayesian Statistics

The whole discipline of Bayesian statistics is founded on one theorem, Bayes' theorem, which is a simple statement of conditional probability:

$$P(a|b) = \frac{P(b|a)P(a)}{P(b)} \tag{A.1}$$

In words this theorem states that the probability of a given that b is true is proportional to the probability of b given that a is true. This rule can be easily applied to answer questions involving simple yes/no events. For example, if a corresponds to actually having breast cancer and b corresponds to receiving a positive test result for breast cancer, then the result of Bayes' rule would give the probability that a person who receives a positive test result actually has cancer. In this case, each term in Bayes' rule is a single number, the probability of the corresponding event.

When the problem of interest is not a simple yes/no question, the terms in Bayes' rule become distributions. In this case, the argument to the distribution $P(a|b)$ is a and the parameter(s) for the distribution is b (in general there may be multiple parameters). The first term in the numerator on the left hand side, $P(b|a)$, is called the likelihood and the other term(s) in the numerator, $P(a)$ in the case of (A.1), is called a prior distribution. The denominator is simply a number that ensures that the posterior distribution, the result of Bayes' rule, is normalized to 1.

As an example, consider a simple study on heights of university students. It seems reasonable that these heights be random draws from a normal distribution

$$y \sim \mathcal{N}(\mu, \sigma^2). \tag{A.2}$$

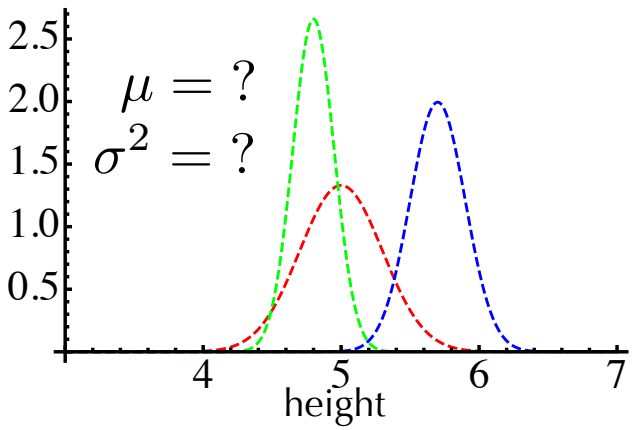


Figure A.1: Illustration of Bayes' rule. It is reasonable to assume that the distribution of University students' heights be a Normal (Gaussian) distribution. However, the location and width of this distribution are unknown. Shown are three possible Normal distributions that could represent these heights. Bayes' rule provides distributions on these values, the mean and width of the likelihood, indicating what values are likely for these parameters. Bayes' rule weighs both the data provided and the prior information about the parameters of interest.

The width and location of this distribution are unknowns and obtaining estimates to their values are provided by Bayes' theorem. This normal distribution is the likelihood in Bayes' rule, which takes the following form for this problem

$$p(\mu, \sigma^2|\mathbf{y}) \propto p(\mathbf{y}|\mu, \sigma^2)p(\mu)p(\sigma^2), \tag{A.3}$$

with

$$p(\mathbf{y}|\mu, \sigma^2) = \mathcal{N}(\mu, \sigma^2). \tag{A.4}$$

Since there are two parameters in the likelihood, there must also be two prior distributions, one for each parameter. These distributions, $p(\mu)$ and $p(\sigma^2)$, are a priori knowledge about the values of μ and σ and are chosen using physical intuition about the situation. The posterior distribution,$p(\mu, \sigma^2|\mathbf{y})$, appropriately weights information from the priors and the data to provide inference on the value and uncertainty of the parameters μ and σ^2.

The choice of prior distribution can greatly affect the computational complexity of the problem. Choosing a conjugate prior distribution will result in a known posterior distribution. Conjugate in this context means that the prior and the posterior belong to the same family of distributions. If the prior is conjugate then when the product of the prior distribution and the likelihood is formed, the resulting distribution is recognizable. For example, if I multiply a normal

likelihood,

$$\mathcal{N}(\mathbf{y}|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(y-\mu)^2}, \tag{A.5}$$

with an inverse gamma distribution,

$$\gamma(\mu|\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \mu^{-\alpha-1} e^{-\frac{\beta}{\mu}}, \tag{A.6}$$

the result is an inverse gamma distribution with the parameters

$$\alpha_n = \alpha + \frac{n}{2} \quad \beta_n = \beta + \frac{\sum_{i=1}^n (y_i - \mu)^2}{2}, \tag{A.7}$$

which means that the mean, variance, and form of the posterior distribution are known. If a non-conjugate prior is chosen, retrieving the posterior distribution requires costly sampling algorithms such as Metropolis-Hasting.

A.1 Bayesian compressive sensing

Here we highlight some of the key points of a Bayesian implementation of compressive sensing put forth by Babacan *et al.*. [82, 83] Framed in the context of cluster expansion, Bayes' rule becomes

$$p(\mathbf{J},\beta|\mathbf{E}) \propto p(\mathbf{E}|\mathbf{J},\beta)p(\mathbf{J})p(\beta), \tag{A.8}$$

where $p(\mathbf{E}|\mathbf{J},\beta)$ is the likelihood and $p(\mathbf{J})$ and $p(\beta)$ are prior distributions. The result of Bayes' rule, or posterior distribution $p(\mathbf{J},\beta|\mathbf{E})$, provides an *a posteriori* estimate on the parameters of interest. In this case those parameters are the model coefficients $\mathbf{J}$ and the variance on the training data β.

The key to merging compressive sensing and Bayesian statistics lies in choosing appropriate distributions for the r.h.s. of Eq. (A.8). The training data are independent and Gaussian

distributed with mean $\bar{\Pi}\mathbf{J}$ and variance β^{-1}

$$p(\mathbf{E}|\mathbf{J},\beta) = \mathcal{N}(\mathbf{E}|\bar{\Pi}\mathbf{J},\beta^{-1}). \tag{A.9}$$

To ensure conjugacy in the analysis of β, the prior $p(\beta)$ was chosen to be a gamma distribution. The prior distribution $p(\mathbf{J})$ contains the *a priori* knowledge about the value of the model coefficients and this choice is key to implementing the CS paradigm. It is well known that the Laplace distribution is a sparsity-promoting prior and is formally equivalent to the convex optimization problem of equation (6.3).

$$p(\mathbf{J}|\lambda) = \frac{\lambda}{2}\exp\left(-\frac{\lambda}{2}\|\mathbf{J}\|_1\right) \tag{A.10}$$

The Laplace distribution enforces the ℓ_1 norm constraint by placing a large probability mass at zero so that signal coefficients close to zero are preferred. Furthermore, the Laplace prior is log-concave, which produces a unimodal posterior distribution and therefore eliminates local minima.

However, the Laplace distribution is not conjugate to the normal likelihood employed here and it's use as the prior would add considerable computational complexity. To maintain conjugacy while still modeling the coefficients with a Laplace distribution, Babacan et. al employ a hierarchical approach. The prior distribution is chosen to be Normal

$$p(\mathbf{J}|\gamma) = \prod_{i=1}^{N}\mathcal{N}(J_i|0,\gamma_i), \tag{A.11}$$

which is conjugate to the Normal likelihood. A prior distribution on the parameter γ, or hyperprior, is then chosen to be a Laplace distribution

$$p(\gamma_i|\lambda) = \frac{\lambda}{2}\exp\left(-\frac{\lambda\gamma_i}{2}\right), \tag{A.12}$$

so that when the prior and the hyperprior are multiplied together and γ is integrated out the resulting distribution is Laplace

$$p(\mathbf{J}|\lambda) = \int p(\mathbf{J}|\gamma)p(\gamma|\lambda)d\gamma = \prod_i \int p(J_i|\gamma_i)p(\gamma_i|\lambda)d\gamma_i \tag{A.13}$$

$$= \frac{\lambda^{N/2}}{2^N}\exp\left(-\sqrt{\lambda}\|\mathbf{u}\|_1\right). \tag{A.14}$$

Eq. (A.8) has now become

$$p(\mathbf{J},\gamma,\beta,\lambda|\mathbf{E}) \propto p(\mathbf{E}|\mathbf{J},\beta)p(\mathbf{J}|\gamma)p(\gamma|\lambda)p(\beta)p(\lambda), \tag{A.15}$$

with the Bayesian framework now providing *a posteriori* estimates on the parameters γ and λ in addition to $\mathbf{J}$ and β.

By preserving conjugacy on the model coefficients $\mathbf{J}$, the exact form of the conditional posterior distribution $p(\mathbf{J}|\gamma,\beta,\lambda,\mathbf{E})$ is known to be Gaussian with mean vector

$$\mu = \Sigma\beta\Pi^T\mathbf{E}, \tag{A.16}$$

and covariance matrix

$$\Sigma = \left[\beta\Pi^T\Pi + \Gamma\right]^{-1}, \tag{A.17}$$

where

$$\Gamma = \text{diag}\left(\frac{1}{\gamma_i}\right). \tag{A.18}$$

Once accurate values for these parameters are known, the resulting distribution provides the sought-after estimate of the model coefficients. However, notice that these parameters are dependent on the parameters γ and β. To estimate the values of γ and β Babacan *et al.* employ a type II maximum likelihood procedure where the conditional posterior distribution $p(\beta,\gamma,\lambda|\mathbf{E})$, is first assembled algebraically. The maximum of the distribution is then found by taking partial derivatives of the distribution w.r.t each parameter in turn. This process yields algebraic expressions for the values

of the parameters which maximize the distribution. These expressions all depend on the other parameters in the model, which suggests an iterative process where the most current version of the set of parameters is used to update the remaining, out-of-date, parameters.

One obvious problem with the iterative process described above is that at each iteration it requires the solution of a system of N equations, where N is the number of basis functions to be considered (a large number for cluster expansion models). To avoid the computationally expensive inverse found in Eq. (A.17) Babacan *et al.* update a single γ_i per iteration. This leads to a very efficient update of the matrix Σ and the mean vector μ. It is insightful to note that if $\gamma_i = 0$ then $\mu_i = 0$ and the corresponding model coefficient is 0. Since we expect sparse solutions, many of the γ_i's are expected to be zero, and the covariance matrix and mean vector can be represented with far fewer dimensions than N.

The algorithm proceeds by beginning with the zero model, all γ_i's are set to zero (all model coefficients are zero), and proceeds as follows.

- set all $\gamma_i = 0$
- While not converged do:
  1. Choose a basis function to consider, γ_i.
  2. Compute the value of γ_i which maximizes the posterior distribution, $\gamma_i^{(m)}$.
     - If $\gamma_i^{(m)} < 0$: prune γ_i out of the model(set $\gamma_1 = 0$).
     - If $\gamma_i^{(m)} > 0$ and $\gamma_i = 0$: Add γ_i to the model.
     - If $\gamma_i^{(m)} > 0$ and $\gamma_i > 0$: Re-stimate the value of γ_i
  3. Update all other parameters.(Σ, μ, λ)
- end While
