2017-12-01

Machine Learning to Discover and Optimize Materials

Conrad Waldhar Rosenbrock
Brigham Young University

Machine Learning to Discover and Optimize Materials

Conrad Waldhar Rosenbrock

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
Branton Campbell
Bret Hess
Eric Homer
Mark Transtrum

Department of Physics and Astronomy

Brigham Young University

ABSTRACT

Machine Learning to Discover and Optimize Materials

Conrad Waldhar Rosenbrock
Department of Physics and Astronomy, BYU
Doctor of Philosophy

For centuries, scientists have dreamed of creating materials by design. Rather than discovery by accident, bespoke materials could be tailored to fulfill specific technological needs. Quantum theory and computational methods are essentially equal to the task, and computational power is the new bottleneck. Machine learning has the potential to solve that problem by approximating material behavior at multiple length scales. A full end-to-end solution must approximate the quantum mechanics, microstructure and engineering tasks well enough to be predictive in the real world.

In the realm of enumeration, systems with many degrees of freedom such as high-entropy alloys may contain prohibitively many unique possibilities so that enumerating all of them would exhaust available compute memory. One possible way to address this problem is to know in advance how many possibilities there are so that the user can reduce their search space. Although tools to calculate this number were available, none performed well for very large systems and none could easily be integrated into low-level languages for use in existing scientific codes. I present an algorithm to solve these problems.

Testing the robustness of machine-learned models is an essential component in any materials discovery or optimization application. Typically, a small number of system-specific tests are used to validate an approach, this may be insufficient in many cases. In particular, for Cluster Expansion models, the expansion may not converge quickly enough to be useful and reliable. Although the method has been used for decades, a rigorous investigation across many systems to determine when CE "breaks" was still lacking. This dissertation includes this investigation along with heuristics that use only a small training database to predict whether a model is worth pursuing in detail.

Computational materials discovery must lead to experimental validation. However, experiments are difficult due to sample purity, environmental effects and many other considerations. In many cases, it is difficult to connect theory to experiment because computation is deterministic. By combining advanced group theory with machine learning, we created a new tool that bridges the experiment-theory gap so that experimental and computed phase diagrams can be harmonized.

Grain boundaries in real materials control many important material properties such as corrosion, thermal conductivity, and creep. Because of their high dimensionality, learning the underlying physics to optimizing grain boundaries is extremely complex. By leveraging a mathematically rigorous representation for local atomic environments, machine learning becomes a powerful tool to approximate properties for grain boundaries. But it also goes beyond predicting properties by highlighting those atomic environments that are most important for influencing the boundary properties. This provides an immense dimensionality reduction that empowers grain boundary scientists to know where to look for deeper physical insights.

Keywords: materials discovery, machine learning, grain boundaries, derivative structure enumeration, Pólya enumeration theorem, monte carlo structure identification, order parameters

ACKNOWLEDGMENTS

In many respects a document like this one is evidence of a remarkable journey. Science is a full-contact sport and requires support both on and off the field. If there was a "best advisor in the world" award, I am certain that Gus would be a serious competitor. In the past five years we have been to Italy, Germany, the UK (4 times) and various other state-side locations for a total of 18 trips to date. Gus has introduced me to people whose combined number of citations exceeds a million. He has mentored me expertly through the rigors and politics of science so that I feel confident and ready to embark on journeys into the unknown. When I first started, I felt a little timid at the prospect of being "an independent scientific researcher", but the journey with him has done that and much more. I have grown exponentially in every area of my life and Gus' influence is a large factor in that growth.

The support at home has been equally stellar! Helen, Emma, Lana and Baby Aster have all borne their share of the load happily and gracefully. They provided just the right amount of pressure to help me strike a balance between everything at play while assuring me throughout that their love would continue anyway. And abroad, the continued support and encouragement of my parents from week to week continues their legacy of love.

Finally, I must acknowledge the support, patience and help of all our collaborators (Gábor Csányi and Eric Homer in particular), and the faculty and staff at BYU for providing the facilities and infrastructure that make such journeys possible.

MACHINE LEARNING

APPLICATIONS IN MULTI-SCALE OPTIMIZATION AND

MATERIALS DISCOVERY

Conrad W. Rosenbrock

Department of Physics and Astronomy
Brigham Young University
December 2017

Contents

CHAPTER 1

Machine Learning and Materials Discovery

Throughout recorded history, materials have been the driving force behind both advances and limitations in technological ability. Although the stone, bronze and iron ages serve as good examples, we are privileged to personally witness the silicon age and the explosion of technologies that now impact every individual in the world. It is natural to ask what comes next.

1.1 QUANTUM MECHANICS AND MATERIALS DISCOVERY

For several decades, researchers have worked tirelessly to find approximate models for material behavior in the real world. As our understanding of quantum mechanics has grown, there have been spectacular successes in engineering materials that directly exploit the quantum nature of matter. Perhaps most famous, semiconductors such as silicon have been engineered and re-engineered to meet diverse requirements in multiple applications [1–4]. However, exclusively experimental approaches to material optimization are slow and expensive. Furthermore, the combinatoric complexity of the space of all possible materials suggests that we have only uncovered the tip of the iceberg. This defines our quest: to discover the next materials age, and to do it within our lifetime.

Inasmuch as experimental optimization and discovery are time-intensive, computational methods to approximate quantum mechanics for materials have become mainstream. Of these methods, Density Functional Theory (DFT) [5, 6] has emerged as the main tool of use because of its sensible trade-off between chemical accuracy and speed. DFT has also had some spectacular successes. For example, in Figure 1.1, several STEM micrographs for a molybdenum disulphide surface with defects are contrasted with simulated surfaces and calculations made with DFT [7]. The computations clearly discover the correct behavior as verified in experiment. While DFT does have several problems [8], it is remarkably versatile for many materials science problems and continues to be used.

With the steady ascent of commodity computing resources, it is now standard practice to perform thousands of DFT calculations every day and compile large materials databases[1]. For example, as of November 2017, aflowlib.org has more than 1.7 million material compounds. Unfortunately, even while this collection is large in an absolute sense, it is only a small fraction of all possibilities. For example, if common structure prototypes are taken (i.e., those commonly found in nature) and a substitutional enumeration is performed for alloys including up to four elements, more than 30 trillion materials are possible. Considering that `aflowlib` has been working for almost a decade, the additional three orders of magnitude are prohibitive.

And this is not the only problem. Even if we do examine the common prototypes, in order to be predictive and confident, we also need to examine other likely candidates from an enumerated list. Depending on the number of elements in the material and the number of basic arrangements to investigate, this can also require a large number of calculations, each of which is in on the order of hours or days when using DFT. In the case of high-entropy alloys [9], the list of possible structures is often large

[1] aflowlib.org, materialsproject.org and the NOMAD Laboratory are three of the largest and most-used.

enough that the enumeration itself becomes a computational challenge. Chapter 2 presents a numerical algorithm that we developed to make such enumeration problems tractable by calculating memory requirements for the final enumerated list before structure enumeration has even started.

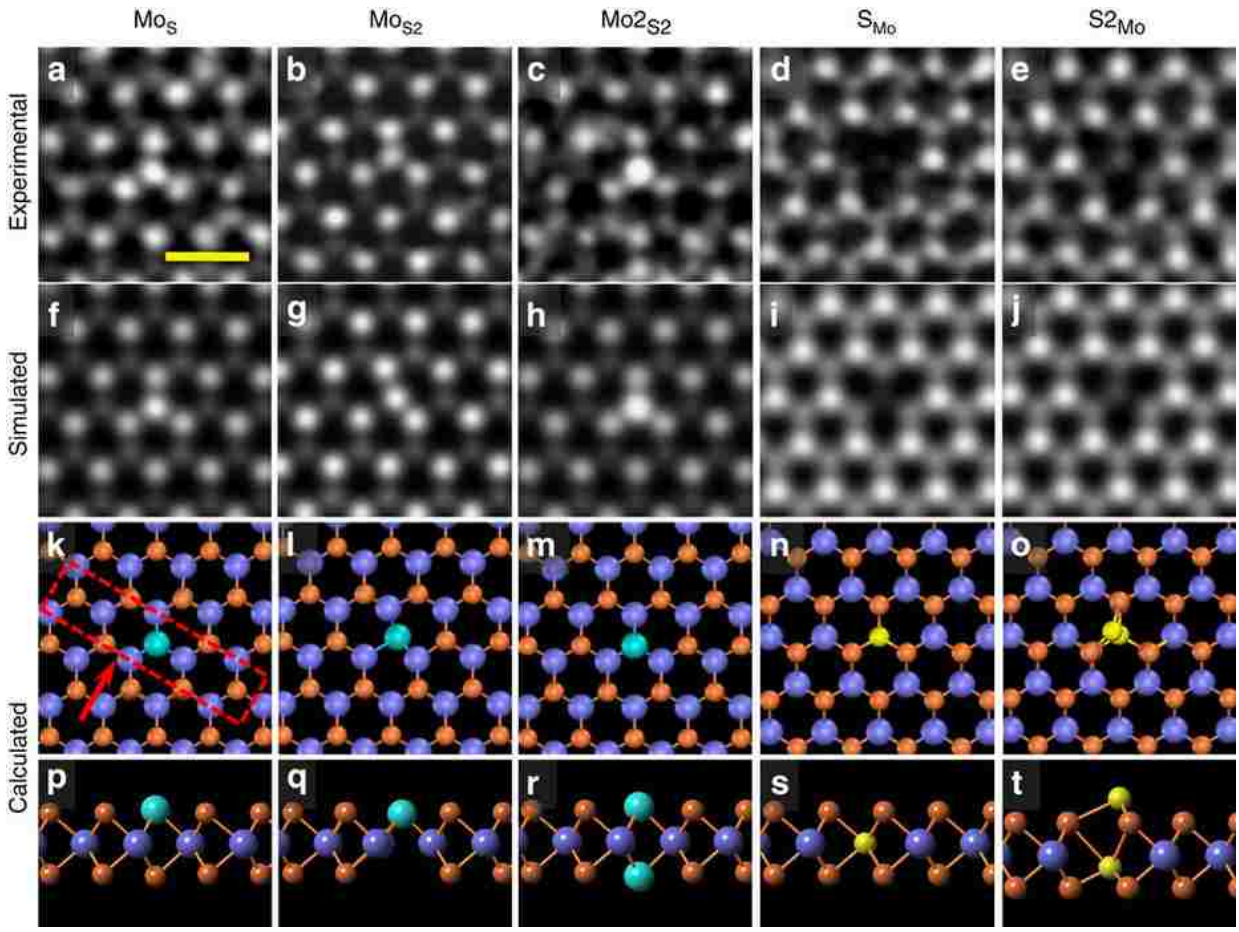


Figure 1.1 Taken from [7]. (**a-c**) High-resolution STEM–ADF images of antisite Mo_S, Mo_{S2} and $Mo2_{S2}$, respectively; (**d-e**) Atomic structures of antisite defects S_{Mo} and $S2_{Mo}$, respectively.; (**f-j**) Simulated STEM images based on the theoretically relaxed structures of the corresponding point defects using QSTEM [10] ; (**k-t**) Relaxed atomic model of all antisite defects in a-e through DFT calculation, with top and side views, respectively.

1.2 MACHINE LEARNING MATERIALS

Statistical models, recently popularized as "machine learning," can approximate high-dimensional spaces with a reasonable precision tradeoff for many problems [11–13]. Some of the hype surrounding machine learning is due to the remarkable discovery that, as datasets grow extremely large, the accuracy of certain models continues to improve rather than saturating asymptotically [14, 15]. For example, better-than-human performance on image recognition [16, 17], tumor detection [18–20], and voice recognition [21] is now possible.

Given the success of machine learning models in other fields, it is natural to ask whether they might be leveraged in materials design. Certainly, examining all possible materials is a problem of high dimensionality. But does the problem lend itself to standard machine learning approaches? There are two important differences between the standard machine learning problems of image recognition, voice recognition, etc., and materials prediction. In the first instance, we cannot afford the typical accuracy tradeoff—materials predictions are not useful without meeting a high accuracy target; the energy difference of competing phases is often very small, requiring high fidelity in the models. The second difference is the amount of training data—we don't have "big data". Although materials science may move into the petabyte regime of distributed data in the future, for now, high accuracy, DFT-computed materials databases contain about only about one million unique materials.

Another potential drawback of traditional machine learning tools in the context of materials problems is that these tools often make excellent predictions in specific cases, but the "knowledge" that they "learn" remains undisclosed. It would be better if the knowledge could be generalized and converted to general engineering principles. In Chapter 5, I show that well-designed representations lead to the discovery of new physics, not merely to predictions alone. And we can do this even though our database is small compared to typical machine learning applications. Importantly, the machine learning was based on mathematically rigorous representations for the material systems built specifically for that task. The often-exaggerated promotion surrounding machine learning comes from a certain class of data sets that lend themselves well to automatic representation. However, automatic representation is generally not possible for arbitrary data sets and is particularly challenging for materials.

Even with readily available tools for finding correlations in datasets, model performance is largely determined by the way in which data is represented to the algorithms (the so-called feature matrix). Images lend themselves well to discrete convolutional representations [22]; audio does well with traditional harmonic analysis [23], wavelet representations [24, 25], and some acoustical analysis [26]. Finding the right representation is the dark art of machine learning [27]. An exciting race to find effective representations for machine learning in materials discovery has produced great innovation in modeling molecular properties [28–35] and behaviors [36], and some innovation in producing force fields for solids [37–42]. However, while these results are encouraging in a fundamental sense, they are only a single part of a multi-scale materials discovery process.

1.3 ASSESSING THE ROBUSTNESS OF MODELS

While several proof-of-concept, generalized machine learning approaches for materials exist, there are important lessons to be learned from history. Cluster Expansion (CE) is a machine learning approach that can accurately approximate quantum mechanical energies for many alloy systems [43–46]. It relies on a representation that is purely configurational, meaning that it learns from materials' composition and relative atomic positions only and ignores the small differences in atomic positions that are present in non-ideal (i.e., real life) materials. Despite this approximation, it is still quite successful and has been a tool of choice in alloy stability studies for many years [47–57]. Interestingly, although CE was often used, its stability and robustness was not rigorously investigated until recently [58]. Whenever machine learning is applied in a materials setting, it is essential to rigorously investigate whether its predictions can be trusted.

In Chapter 3, I present a rigorous study on how errors and noise in DFT calculations affect the ability of CE to approximate material energies. This chapter also includes a discussion on the dangers of applying methods outside of the subspace for which they were designed and when it may be possible to detect such deviations.

1.4 MULTI-SCALE MATERIALS DISCOVERY

The discussion above about DFT addresses the atomistic view of materials. But macroscopic applications of materials encounter additional problems. Grain boundaries are interfaces between neighboring crystalline regions and thus act across length scales several orders of magnitude bigger than DFT can handle. Furthermore, grain boundaries exert a significant influence on material properties such as strength, ductility, corrosion resistance, crack resistance, and conductivity.

Modern day technologies need materials with optimized properties but are hampered by a limited understanding of the physics that drives grain boundary behavior and, in turn, dictates material properties. Chapter 5 shows how machine learning helps us to approximate grain boundary properties *and* to learn the physics that governs them.

Once we have entered the realm of macroscopic materials, difficulties between computational theory and experiment begin to surface as well. Even if DFT or a machine learning model correctly predicts a stable structure at a certain composition, it may not be easy to create that structure in the lab: although the arrangement may indeed be stable, kinetic pathways to reach that arrangement may not be accessible. Chapter 4 presents one possible way to address this problem by combining a machine learning model with Monte Carlo simulations and group theoretical analysis to predict transition structures between stable or meta-stable states. This allows a bridge to be created between theory and experiment.

1.5 OUTLOOK AND FUTURE WORK

The chapters presented in this dissertation are a sampling of some of the difficulties associated with an end-to-end materials discovery process. The ultimate goal is to design and optimize materials from scratch using only computational methods, and then have the final product in the lab match the predictions at every level. This task is sufficiently vast that it requires collaboration with scientists worldwide.

Currently, we are working with collabo-

rators in the UK and Russia to verify the first-ever approach to building alloy potentials that are quantum accurate for static properties such as energies and forces, but which also reproduce dynamic features such as the phonon dispersion curves. At the same time, we are verifying these alloy potentials as candidates for generating quantum-accurate phase diagrams, a first in the history of the world. One of these potentials uses the same representation discussed in Chapter 5. One of the most exciting aspects of these alloy potentials is that they are amenable to automated fitting. Although free parameters do exist in any of the models, we can automate selection of these using heuristic algorithms.

We are concurrently working on isolating the "fingerprint" of material systems by equating the linear bases of Cluster Expansion (CE) and the Smooth Overlap of Atomic Potentials (SOAP). Since both methods provide mathematically complete bases, this provides a formally rigorous approach to identifying the space-dependent interaction parameters between chemical species. Since CE is effective across all compositions for on-lattice structures, it learns the species interactions extremely well. The tradeoff is that it cannot handle structures that have relaxed from ideal lattice positions. In contrast, the SOAP basis is flexible enough to span the entire materials space, but there is no well-defined process for extracting the species interaction terms needed by the basis. By equating the two approaches on a physical property such as energy, we can use CE's approximation on-lattice to approximate species interactions throughout materials space. The approach is discussed in more detail in the appendix.

Finally, we plan to release the method and code to convert existing Cluster Expansion models into alloy potentials using existing data and a minimal number of additional calculations. This enables expertise from multiple decades of work to be transferred to state-of-the-art models with very little human time.

Each of the next chapters is centered around a peer-reviewed article, typeset in the style of the journal in which it was published. Each includes a short revision of the context surrounding the work and then a description of how it is currently being used, with plans for any future development. The concluding chapter has a short list of software packages produced in conjunction with the research presented in this dissertation and a description of their typical use cases.

CHAPTER 2

Optimizing Enumeration for Complex Systems

A seminal question in alloy discovery is how to enumerate all possible alloys for a given set of elements. In theory, we should investigate all of these possibilities to ensure that we have found the lowest energy (i.e., most stable) structure. The problem is complicated by the presence of symmetry: we would prefer to enumerate only symmetrically unique possibilities so that we don't compute the same configuration twice. The enumeration problem was essentially solved by Hart et al. [59]. However, certain systems such as high entropy alloys [9] may contain prohibitively many unique possibilities so that enumerating all of them would exhaust available compute memory.

One possible way to address this problem is to know in advance how many possibilities there are so that the user can reduce their search space by restricting the occupation of certain lattice sites. The Pólya enumeration theorem discussed below [60] answers precisely this question. However, no low-level algorithm was available to make it usable in practice. We produced this algorithm for two reasons:

1. To allow the alloy enumeration code to know in advance the number of unique structures that it would find. This helps with memory partitioning (which optimizes the enumeration) and provides a way to notify the user of an intractable problem.

2. To verify that the alloy enumeration code did in fact find all possible unique decorations of the lattice.

Our algorithm is currently used in the latest version of the enumeration code and is also available open source for both `fortran` and `python` at https://github.com/rosenbrockc/polya. Because of the central role that enumeration plays in the multi-scale process of materials discovery, this contribution will continue to be used for many years.

My contribution to this project includes creation of the algorithm and drafting the majority of the paper. Wiley Morgan helped me debug the algorithm, implement it in `fortran` and provide unit test cases. He and the other authors helped refine the text, figures and presentation of the algorithm to make it easier to understand.

The following article is reproduced with permission. A license is on file with the Department of Physics and Astronomy.

Numerical Algorithm for Pólya Enumeration Theorem

CONRAD W. ROSENBROCK, WILEY S. MORGAN, and GUS L. W. HART,
Brigham Young University
STEFANO CURTAROLO, Duke University
RODNEY W. FORCADE, Brigham Young University

Although the Pólya enumeration theorem has been used extensively for decades, an optimized, purely numerical algorithm for calculating its coefficients is not readily available. We present such an algorithm for finding the number of unique colorings of a finite set under the action of a finite group.

Categories and Subject Descriptors: G.2.1 [**Discrete Mathematics**]: Combinatorics

General Terms: Combinatorial Algorithms, Counting Problems

Additional Key Words and Phrases: Pólya enumeration theorem, expansion coefficient, product of polynomials

ACM Reference Format:
Conrad W. Rosenbrock, Wiley S. Morgan, Gus L. W. Hart, Stefano Curtarolo, and Rodney W. Forcade. 2016. Numerical algorithm for pólya enumeration theorem. J. Exp. Algorithmics 21, 1, Article 1.11 (August 2016), 17 pages.
DOI: http://dx.doi.org/10.1145/2955094

1. INTRODUCTION

A circle partitioned into 4 equal sectors can be colored 16 different ways using two colors, $2^4 = 16$, as shown in Figure 1. But only 6 of these colorings are symmetrically distinct, several others being equivalent (under rotations and reflections) as shown by the arrows in the figure. The Pólya enumeration theorem provides a way to determine how many symmetrically distinct colorings there are with, for example, all sectors red (only one, as shown in the figure), one red sector and three green (again, only one), or the number with two red sectors and two green sectors (two, as shown in the figure). Borrowing a word from physics and chemistry, we refer to the partition of red and green sectors as the *stoichiometry*. For example, a coloring with 1 red sector and 3 green sectors has a stoichiometry of 1:3.

The Pólya theorem [Pólya 1937; Pólya and Read 1987] produces a polynomial (generating function), shown in the figure, whose coefficients answer the question of how many distinct colorings there are for each stoichiometry (each partition of the colors). For example, the $2r^2g^2$ term in the polynomial indicates that there are two distinct ways to color the circle with 2:2 stoichiometry (). For all other stoichiometries (4:0,

This work was supported under Grant No. ONR (MURI N00014-13-1-0635).
Authors' addresses: C. W. Rosenbrock, W. S. Morgan, and G. L. W. Hart, Department of Physics and Astronomy, 84602, Brigham Young University; S. Curtarolo, Materials Science, Electrical Engineering, Physics and Chemistry, 27708, Duke University; R. W. Forcade, Department of Mathematics, 84602, Brigham Young University.

DOI: http://dx.doi.org/10.1145/2955094

$$P(r,g) = r^4 + r^3g + 2r^2g^2 + rg^3 + g^4$$

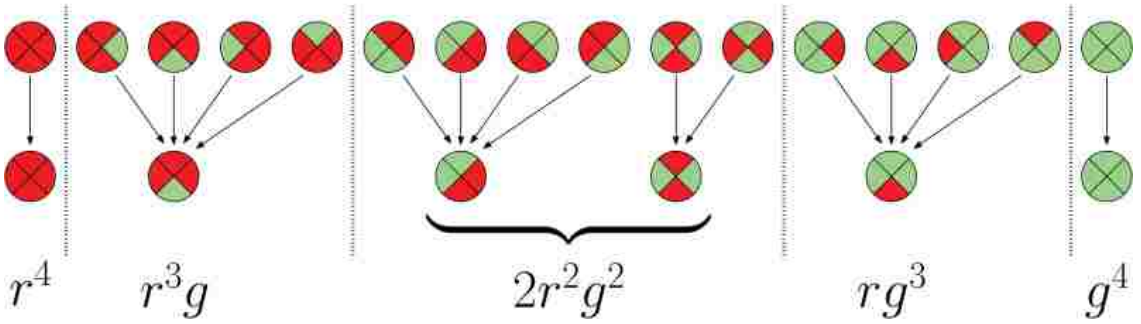


Fig. 1. Top row: All possible two-color colorings of a circle divided into four equal sectors (left side of figure). Bottom row: All symmetrically distinct binary colorings of the circle. Arrows indicate combinatorically distinct colorings that are equivalent by symmetry.

0:4, 1:3, and 3:1), the polynomial coefficients are all 1, indicating that for each of these cases there is only one distinct coloring, as is obvious from the figure.

A common problem in many fields involves enumerating[1] the *symmetrically distinct* colorings of a finite set, similar to the toy problem of Figure 1. The Pólya theorem has shown its wide range of applications in a variety of contexts. Classically, it was applied to counting chemical isomers [Robinson et al. 1976; Kennedy et al. 1964; Pólya 1937] and graphs [Harary 1955]. Recent examples include confirming enumerations of molecules in bioinformatics and chemoinformatics [Deng and Qian 2014; Ghorbani and Songhori 2014]; unlabeled, uniform hypergraphs in discrete mathematics [Qian 2014]; analysis of tone rows in musical composition [Lackner et al. 2015]; commutative binary models of Boolean functions in computer science [Genitrini et al. 2015]; generating functions for single-trace-operators in high-energy physics [McGrane et al. 2015]; investigating the role of nonlocality in quantum many-body systems [Tura et al. 2015]; and photosensitizers in photosynthesis research [Taniguchi et al. 2014].

In computational materials science, chemistry, and related subfields such as computational drug discovery, combinatorial searches are becoming increasingly important, especially in high-throughput studies [Curtarolo et al. 2013]. As computational methods gain a larger market share in materials and drug discovery, algorithms such as the one presented in this article are important as they provide validation support to complex enumeration codes. Pólya's theorem is the only way to independently confirm that an enumeration algorithm has performed correctly. The present algorithm has been useful in checking a new algorithm extending the work in Hart and Forcade [2008, 2009] and Hart et al. [2012], and Pólya's theorem was recently used in a similar crystal enumeration algorithm [Mustapha et al. 2013] that has been incorporated into the CRYSTAL14 software package [Dovesi et al. 2014].

Despite the widespread use of Pólya's theorem in different science and mathematics contexts, a low-level, numerical implementation is not available. Typical approaches use Computer Algebra Systems (CASs) to symbolically generate the Pólya polynomial. This strategy is ineffective for two reasons. First, CASs are too slow for large problems that arise in a research setting, and, second, generating the entire Pólya polynomial (which can have billions or trillions of terms) is unnecessary when one is interested in only a single stoichiometry.

[1]The Pólya theorem does not generate the list of unique colorings (which is generally a much harder problem), but it does determine the *number* of unique colorings.

Here we demonstrate a low-level algorithm for finding the polynomial coefficient corresponding to a single stoichiometry. It exploits the properties of polynomials and *a priori* knowledge of the relevant term. We briefly describe the Pólya enumeration theorem in Section 2, followed by the algorithm for calculating the polynomial coefficients in Section 3. In the final section, we investigate the scaling and performance of the algorithm.

2. PÓLYA ENUMERATION THEOREM

2.1. Introduction the Pólya Enumeration Theorem

Pólya's theorem provides a simple way to construct a generating polynomial whose coefficients count the numbers of symmetrically distinct colorings for each possible stoichiometry. The polynomial in Figure 1 above was easy to verify because we were able to hand count the symmetrically distinct colorings. But suppose there were dozens of colors and dozens of sites to be colored and hundreds of symmetries to apply. In that case, it is easier to use Pólya's theorem to construct the polynomial directly from the symmetry group.

To describe this very useful theorem, we refer once more to Figure 1. There are four symmetries—the identity, two 90° rotations (clockwise and counterclockwise), and a 180° rotation. If we label the colorable sectors 1, 2, 3, and 4, and write the permutations in *disjoint-cycle* notation, we have (1)(2)(3)(4) for the identity, the two 90° rotations are represented by (1234) and (1432), while the 180° rotation is (13)(24) in cycle notation.

Now Pólya's theorem simply tells us to replace each cycle of length λ with a sum of λ-th powers of variables corresponding to the colors available. For example, letting r and g stand for red and green, the identity is represented by $(r+g)(r+g)(r+g)(r+g)$, the two 90° rotations are each replaced by (r^4+g^4), and the 180° rotation is replaced by $(r^2+g^2)(r^2+g^2)$. When we *average* these four polynomials, we get the Pólya polynomial predicted above:

$$\begin{aligned} P(r,g) &= \frac{1}{4}\big((r+g)(r+g)(r+g)(r+g) + (r^4+g^4) + (r^4+g^4) + (r^2+g^2)(r^2+g^2)\big) \\ &= r^4 + r^3g + 2r^2g^2 + rg^3 + g^4. \end{aligned} \tag{1}$$

In other words, Pólya's theorem relies on a structural representation of the symmetries *as permutations written in disjoint-cycle notation* to construct the generating polynomial we need.

The problem with Pólya, however, is that it requires us to compute the *entire* polynomial when we may need only one of its coefficients. For example, if we have 50 sites to color, and 20 colors available, the number of *terms* in our polynomial (regardless of symmetries) would be about 4.6×10^{16}. That is a lot of work (and memory) to compute the entire polynomial (and all of those very large terms) if we needed *only* to know the number of symmetrically distinct colorings for a single stoichiometry. That information is contained in just 1 term of the 46 quadrillion terms of the Pólya polynomial. Can we spare ourselves the work of computing all the others?

Suppose we have a target stoichiometry $[c_1 : c_2 : \cdots : c_\xi]$, where ξ is the number of colors and $\sum_{j=1}^{\xi} c_j = n$ is the number of sites to be colored. To find the number of symmetrically distinct colorings with those frequencies, we must determine the coefficient of the single term in the Pólya polynomial containing the product $x_1^{c_1}x_2^{c_2}\ldots x_\xi^{c_\xi}$. The Pólya polynomial is the average,

$$P(x_1, x_2, \ldots, x_\xi) = \frac{1}{|G|}\left(\sum_{\pi\in G} P_\pi(x_1, x_2, \ldots, x_\xi)\right), \tag{2}$$

of the polynomials $P_\pi(x_1, x_2, \ldots, x_\xi)$ computed for each permutation π in the symmetry group G, each P_π being formed by multiplying the representations of each disjoint cycle in π (as illustrated in Equation (1)).

Clearly, if we are only interested in the coefficient of $x_1^{c_1}x_2^{c_2}\ldots x_\xi^{c_\xi}$ in P, we may simply find the coefficient of that product in each P_π and add those partial coefficients together. Thus, given a permutation π with k_1 cycles of length r_1, k_2 cycles of length r_2, and so on, up to k_t cycles of length r_t, with $\sum_{i=1}^{t} r_i k_i = n$ (the number of sites, t is the number of cycle types), we must compute the coefficient of $x_1^{c_1}x_2^{c_2}\ldots x_\xi^{c_\xi}$ in P_π.

It is well known that a product of sums is equal to the sum of all products one can obtain by taking one summand from each factor (generalizing the familiar First Outer Inner Last (FOIL) rule used by undergrads to multiply two binomials). Thus the polynomial P_π is the sum of all products of the form $\prod_s x_{i_s}^{\lambda(s)}$ (where the product runs over all cycles s, $\lambda(s)$ is the length of the cycle s, and x_{i_s} is one of the colors chosen from the sum for that cycle). Thus the product we want, $x_1^{c_1}x_2^{c_2}\ldots x_\xi^{c_\xi}$, has a coefficient that simply counts the number of products of the form $\prod_s x_{i_s}^{\lambda(s)}$ where the sum of the exponents for each x_i is equal to the target c_i.

Each cycle, of length r_i ($i = 1\ldots t$), gets assigned to one of the colors. Let s_{ij} be the number of cycles of length r_i assigned to color j ($j = 1\ldots\xi$). This defines a $t \times \xi$ matrix $S = (s_{ij})$ of non-negative integers, where (1) the sum of row i equals the number of cycles of length r_i:

$$\sum_{j=1}^{\xi} s_{ij} = k_i \quad \text{(row sum condition)}, \tag{3}$$

and (2) weighted sum of column j must equal the target frequency of the j-th color:

$$\sum_{i=1}^{t} r_i s_{ij} = c_j \quad \text{(column sum condition)}, \tag{4}$$

in order to achieve our target stoichiometry.

For each such matrix, there are a number of possible ways to assign colors to the cycles, with multiplicities prescribed by S. The number is

$$F(S) = \prod_{i=1}^{t} \binom{k_i}{s_{i1}, s_{i2}, \ldots, s_{i\xi}}, \tag{5}$$

the product of the number of ways to do it for each cycle. Thus we are obliged to sum the function $F(S)$, so computed, over all matrices S meeting the given row and column sum conditions (3) and (4).

If we do this computation for each permutation π, and average them (add them and divide by $|G|$), we then get the coefficient of the Pólya polynomial $P(x_1, x_2, \ldots, x_i)$ corresponding to our target stoichiometry $[c_1 : c_2 : \cdots : c_\xi]$. This calculation depends only on the *cycle type* of the permutation, the number of disjoint cycles of different lengths comprising the disjoint-cycle representation. Thus we only need to make an inventory of the cycle types for our permutations and do the calculation once for each distinct cycle type. There will not be more such cycle types than the number of conjugacy classes in the symmetry group. Also, note, the utility of multinomial coefficients in this context stems from the likelihood that our permutations will have many cycles of the same length.

Algorithmically, the process is straight forward. First, we must find all matrices S which meet the row and sum conditions (3) and (4) above. For each successful matrix,

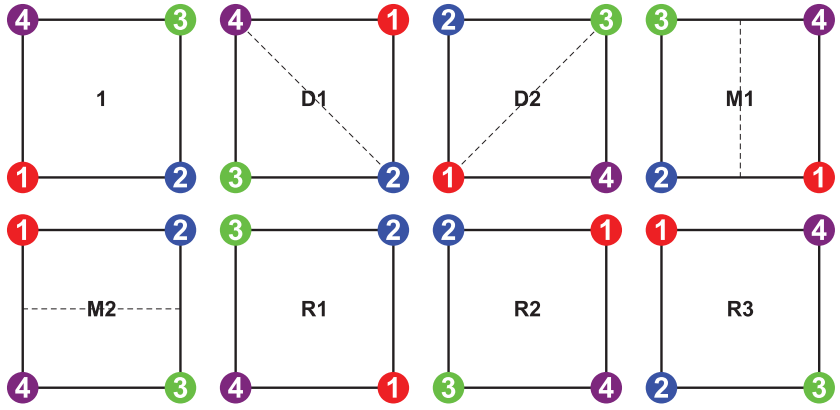


Fig. 2. The symmetry group operations of the square. This group is known as the dihedral group of degree 4 or D_4. The dashed lines are guides to the eye for the horizontal, vertical, and diagonal reflections (**M1**,**M2** and **D1**, **D2**).

we then compute the product of row-multinomial-coefficients. We add those up and multiply by the number of permutations in the conjugacy class, sum those results for the conjugacy classes, and divide by the group order. That gives us the Pólya coefficient for the given stoichiometry.

For example, suppose our permutation is made up of two 1-cycles, three 2-cycles, and one 4-cycle (so the number of sites is 12), and we have three colors with frequencies (red:green:blue $\rightarrow$ 4:6:2) respectively. Then we are looking for 3×3 matrices S whose rows sum to $\begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix}$ and whose columns (when dotted with the cycle lengths $\begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix}$) sum to 4, 6, and 2 respectively. There are exactly five such matrices (see Figure 3 and discussion in Section 3):

$$\begin{pmatrix} 0 & 0 & 2 \\ 0 & 3 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 2 \\ 2 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 2 & 0 \\ 0 & 2 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 2 & 0 \\ 2 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 2 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix}. \tag{6}$$

The multinomial coefficient for the top and bottom row in each case is $\binom{2}{2,0,0} = \binom{2}{2} = 1 = \binom{1}{1,0,0}$, so the $F(S)$ in each case is equal to the multinomial coefficient of the middle row; thus $\binom{3}{3} = 1$ in the first case, $\binom{3}{2,1} = 3$ for the middle three matrices, and $\binom{3}{1,1,1} = 6$ for the right-hand matrix. So our count for this problem is $1 + 3 + 3 + 3 + 6 = 16$. We may check this by computing $(r + g + b)^2(r^2 + g^2 + b^2)^3(r^4 + g^4 + b^4)$ (a la Pólya) and noting that the coefficient of $r^4g^6b^2$ is indeed 16.

Clearly, we can do that for each permutation in the group and sum the results. That is equivalent to determining in how many ways we may assign a single color to each cycle in the permutation—in such a way that the total number of occurrences of each color achieves its target frequency.

2.2. Example: Applying Pólyas Theorem

Here we present a simple example showing how Pólya's theorem is applied to a small, finite group. The square has the set of symmetries displayed in Figure 2. These symmetries include four rotations (by 0°, 90°, 180°, and 270°; labeled **1**, **R1**, **R2**, and **R3**) and four reflections (one horizontal, one vertical, and two for the diagonals; labeled

Table I. Disjoint-Cyclic Form for Each Group Operation in D_4 and the Corresponding Polynomials, Expanded Polynomials and the Coefficient of the x^2y^2 Term for Each

Op.	Disjoint-Cyclic	Polynomial	Expanded	Coeff.
$\mathbb{1}$	(1)(2)(3)(4)	$(x+y)^4$	$x^4+4x^3y+\boxed{6x^2y^2}+4xy^3+y^4$	6
D1	(1, 3)(2)(4)	$(x^2+y^2)(x+y)^2$	$x^4+2x^3y+\boxed{2x^2y^2}+2xy^3+y^4$	2
D2	(1)(2, 4)(3)	$(x^2+y^2)(x+y)^2$	$x^4+2x^3y+\boxed{2x^2y^2}+2xy^3+y^4$	2
M1	(1, 2)(3, 4)	$(x^2+y^2)^2$	$x^4+\boxed{2x^2y^2}+y^4$	2
M2	(1, 4)(2, 3)	$(x^2+y^2)^2$	$x^4+\boxed{2x^2y^2}+y^4$	2
R1	(1, 4, 3, 2)	(x^4+y^4)	$x^4+\quad+y^4$	0
R2	(1, 3)(2, 4)	$(x^2+y^2)^2$	$x^4+\boxed{2x^2y^2}+y^4$	2
R3	(1, 2, 3, 4)	(x^4+y^4)	$x^4+\quad+y^4$	0

M1, **M2** and **D1**, **D2**). This group is commonly known as the dihedral group of degree four, or D_4 for short.[2]

The group operations of the D_4 group can be written in disjoint-cyclic form as in Table I. For each r-cycle in the group, we can write a polynomial in variables x_i^r for $i = 1 \ldots \xi$, where ξ is the number of colors used. For this example, we will consider the situation where we want to color the four corners of the square with only two colors. In that case we end up with just two variables x_1, x_2, which are represented as x, y in the table.

The Pólya representation for a single group operation in disjoint-cyclic form results in a product of polynomials that we can expand. For example, the group operation **D1** has disjoint-cyclic form $(1, 3)(2)(4)$ that can be represented by the polynomial $(x^2 + y^2)(x + y)(x + y)$, where the exponent on each variable corresponds to the length of the r-cycle of which it is a part. For a general r-cycle, the polynomial takes the form

$$\left(x_1^r + x_2^r + \cdots + x_\xi^r\right), \tag{7}$$

for an enumeration with ξ colors. As described in Section 2.1, we exchange the group operations acting on the set for polynomial representations that obey the familiar rules for polynomials.

We will now pursue our example of the possible colorings on the four corners of the square involving two of each color. Excluding the symmetry operations, we could come up with $\binom{4}{2} = 6$ possibilities, but some of these are equivalent by symmetry. The Pólya theorem counts how many *unique* colorings we should recover. To find that number, we look at the coefficient of the term corresponding to the overall color selection (in this example, two of each color); thus we look for coefficients of the x^2y^2 term for each group operation. These coefficient values are listed in Table I. The sum of these coefficients, divided by the number of operations in the group, gives the total number of unique colorings under the entire group action, in this case $(6 + 2 + 2 + 2 + 2 + 0 + 2 + 0)/8 = 16/8 = 2$.

Next, we apply the procedure discussed in connection with Equation (6) to construct the matrix S for one of the permutations of the square. It illustrates the idea behind the general algorithm presented in the next section.

In the symmetries of the square, there is a cycle type consisting of two 1-cycles and one 2-cycle. The two permutations with that type are (1)(3)(24) and (2)(4)(13). The cycle lengths are 1 (with multiplicity 2) and 2 (with multiplicity 1). So each of those

[2]The dihedral groups have multiple, equivalent names. D_4 is also called Dih_4 or the dihedral group of *order* 8 (D_8).

permutations requires a matrix $S = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix}$ satisfying $s_{11} + s_{12} = 2$ and $s_{21} + s_{22} = 1$ (row sum condition (3)) and $s_{11} + 2s_{21} = 2$ and $s_{12} + 2s_{22} = 2$ (column sum condition (4)). There are only two matrices of non-negative integers satisfying those conditions simultaneously:

$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \text{ and } \begin{pmatrix} 0 & 2 \\ 1 & 0 \end{pmatrix}. \tag{8}$$

For each of these matrices, the row-multinomial coefficients are $\binom{2}{0,2} = 1$ and $\binom{1}{0,1} = 1$ so each matrix yields a product 1. Thus each permutation of this cycle type contributes 2 to the sum. This corresponds to the fact that the coefficient of x^2y^2 in $(x+y)^2(x^2+y^2)$ is 2.

Since there are two permutations of this cycle type, the total contribution of the cycle type to the overall Pólya polynomial is 4 (which must then be divided by the number of symmetries in the group).

Thus, in general, the only problem is to find an efficient way of generating these matrix solutions. Since the problem is equivalent to enumerating all lattice points within a high-dimensional polytope, we presume that a tree search (implemented recursively or via a backtracking algorithm) may be the most efficient way to achieve this.

3. COEFFICIENT-FINDING ALGORITHM

Our implementation of the tree search is fundamentally identical to the method of the last section; however, the details may not be immediately recognizable as such.[3] In this section we rephrase the row and column sum conditions (3) and (4) to highlight the logical connections between our specific implementation and the general ideas from Section 2. We adopt this approach because (1) for pedagogical value, the matrix approach is much easier to visualize and (2) the algorithms presented here mirror the accompanying code closely, which we consider valuable.

First, for a generic polynomial

$$\left(x_1^r + x_2^r + \cdots + x_\xi^r\right)^d, \tag{9}$$

the exponents of each x_i in the *expanded* polynomial are constrained to the set

$$V = \{0, r, 2r, 3r, \ldots, dr\}. \tag{10}$$

Next, we consider the terms in the expansion of the polynomial:

$$\left(x_1^r + x_2^r + \cdots + x_\xi^r\right)^d = \sum_{k_1,k_2,\ldots,k_\xi} \mu_k \prod_{i=1}^{\xi} x_i^{rk_i}, \tag{11}$$

where the sum is over all possibles sequences $k_1, k_2, \ldots, k_\xi$ such that the sum of the exponents (represented by the sequence in k_i) is equal to d,

$$k_1 + k_2 + \cdots + k_\xi = d. \tag{12}$$

[3]If all you are looking for is a working code, you now know enough to use it. Download it at https://github.com/rosenbrockc/polya.

As described in the introduction, the coefficients μ_k in the polynomial expansion Equation (11) are found using the multinomial coefficients

$$\begin{aligned}\mu_k &= \binom{n}{k_1, k_2, \ldots, k_\xi} = \frac{n!}{k_1! k_2! \cdots k_\xi!} \\ &= \binom{k_1}{k_1}\binom{k_1+k_2}{k_2}\cdots\binom{k_1+k_2+\cdots+k_\xi}{k_\xi} \\ &= \prod_{i=1}^{\xi}\binom{\sum_{j=1}^{i} k_j}{k_i}. \end{aligned} \tag{13}$$

Finally, we define the polynomial (7) for an arbitrary group operation $\pi \in \mathbf{G}$ as[4]

$$P_\pi(x_1, x_2, \ldots, x_\xi) = \prod_{\alpha=1}^{m} M_\alpha^{r_\alpha}(x_1, x_2, \ldots, x_\xi), \tag{14}$$

where each $M_\alpha^{r_\alpha}$ is a polynomial of the form (9) for the α^{th} distinct r-cycle and d_α is the multiplicity of that r-cycle; m is the number of cycle types in P_π. Linking back to the matrix formulation, each $M_\alpha^{r_\alpha}$ is equivalent to a *row* S_i in matrix S.

Since we know the fixed stoichiometry term $T = \prod_{i=1}^{\xi} T_i = \prod_{i=1}^{\xi} x_i^{c_i}$ *in advance, we can limit the possible sequences of* k_i *for which multinomial coefficients are calculated*. This is the key idea of the algorithm and the reason for its high performance.

For each group operation π, we have a product of polynomials $M_\alpha^{r_\alpha}$. We begin filtering the sequences by choosing only those combinations of values $v_{i\alpha} \in V_\alpha = \{v_{i\alpha}\}_{i=1}^{d_\alpha+1}$ for which the sum

$$\sum_{\alpha=1}^{m} v_{i\alpha} = T_i, \tag{15}$$

where V_α is the set from Eq. (10) for multinomial $M_\alpha^{r_\alpha}$. At this point it is useful to refer to Figure 3 to make the connection to the recursive tree search for possible matrices. The V_α are equivalent to all the possible values that any of the elements in a row of the matrix may take. If we take $M_1^{r_1}$ as an example, then V_1 is the collection of all values that show up in row 1 of any matrix in the figure, multiplied by the number of cycles with length r_1. Constraint (15) is equivalent to the column sum requirement (4).

We first apply constraint (15) to the x_1 term across the product of polynomials to find a set of values $\{k_{1\alpha}\}_{\alpha=1}^{m}$ that could give exponent T_1 once all the polynomials' terms have been expanded. This is equivalent to finding the set of first columns in each matrix that match the target frequency for the first color. Once a value $k_{1\alpha}$ has been fixed for each $M_\alpha^{r_\alpha}$, the remaining exponents in the sequence $\{k_{1\alpha}\} \cup \{k_{i\alpha}\}_{i=2}^{\xi}$ are constrained via (12). We can recursively examine each variable x_i in turn using these constraints to build a set of sequences

$$S_l = \{S_{l\alpha}\}_{\alpha=1}^{m} = \{(k_{1\alpha}, k_{2\alpha}, \ldots, k_{\xi\alpha})\}_{\alpha=1}^{m}, \tag{16}$$

where each $S_{l\alpha}$ defines the exponent sequence for its polynomial $M_\alpha^{r_\alpha}$ that will produce the target term T after the product is expanded. Each $S_{l\alpha} \in S_l$ represents the transposed matrix S that survives both the row and column sum conditions (highlighted in green in the figure). Thus, S_l is the set of these matrices for the group operation π. The

[4]We will use Greek subscripts to label the polynomials in the product and Latin subscripts to label the variables within any of the polynomials.

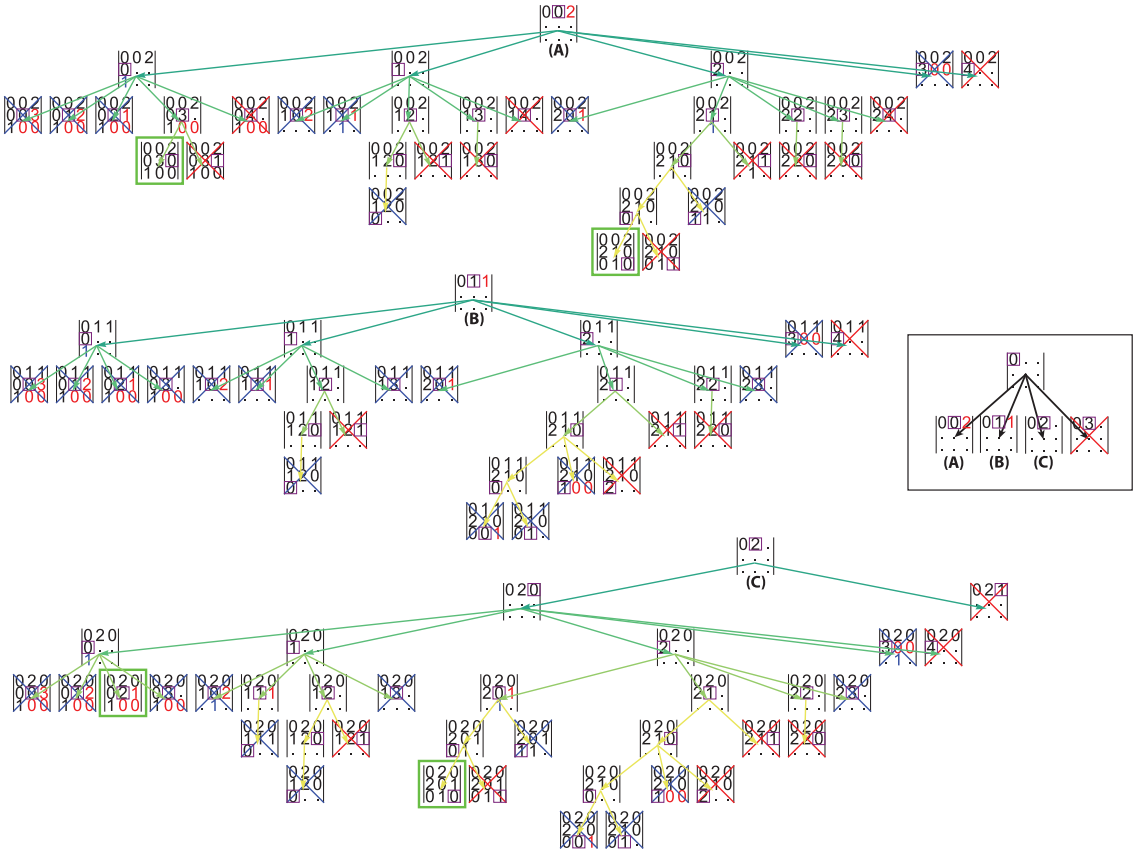


Fig. 3. A recursive tree search for some of the possible matrices S for the problem of Section 2: two 1-cycles, three 2-cycles, and one 4-cycle. We have restricted the figure to include only the zero pendants of the tree, which produce four of the five relevant matrices in Equation (6). Matrix elements in red (blue) represent the only possible values that would satisfy the row (column) sum conditions. A red (blue) cross over a matrix shows that it fails the row (column) sum condition, and its descendants need not be examined. Matrices with green borders are solutions to the tree search problem. The purple squares show the current row and column on which the recursive search is operating.

maximum value of l depends on the target term T and how many possible $v_{i\alpha}$ values are filtered out using constraints (15) and (12) at each step in the recursion.

Once the set $\mathbf{S} = \{S_l\}$ has been constructed, we use Equation (13) on each polynomial's $\{k_{i\alpha}\}_{i=1}^{\xi}$ in $S_{l\alpha}$ to find the contributing coefficients. The final coefficient value for term T resulting from group operation π is

$$t_\pi = \sum_l \tau_l = \sum_l \prod_{\alpha=1}^{m} \binom{d_\alpha}{S_{l\alpha}}. \tag{17}$$

To find the total number of unique colorings under the group action, this process is applied to each element $\pi \in \mathbf{G}$ and the results are summed and then divided by $|\mathbf{G}|$.

We can further optimize the search for contributing terms by ordering the exponents in the target term T in descending order. All the $\{k_{1\alpha}\}_{\alpha=1}^{m}$ need to sum to T_1 (15); larger values for T_1 are more likely to result in smaller sets of $\{k_{i\alpha}\}_{\alpha=1}^{m}$ across the polynomials. This happens because if T_1 has smaller values (like 1 or 2), then we end up with lots of possible ways to arrange them to sum to T_1 (which is not the the case for the larger values). Since the final set of sequences S_l is formed using a Cartesian product, including a few extra sequences from any T_i prunings multiplies the total number of

sequences significantly. In the figure, this optimization is equivalent to completing a row with red entries because all the remaining, unfilled entries are constrained by the row sum condition.

Additionally, constraint (12) applied within each polynomial will also reduce the total number of sequences to consider if the first variables x_1, x_2, and so on, are larger integers compared to the target values T_1, T_2, and so on. This speed-up comes from the recursive implementation: If x_1 is already too large (compared to T_1), then possible values for $x_2, x_3, \ldots$ are never considered. This optimization is equivalent to completing matrix columns with blue entries because of the column sum constraint.

3.1. Pseudocode Implementation

Note: Implementations in PYTHON and FORTRAN are available in the supplementary material.

For both algorithms presented below, the operator ($\Leftarrow$) pushes the value to its right onto the list to its left.

For algorithm (1) in the EXPAND procedure, the $\cup$ operator horizontally concatenates the integer *root* to an existing sequence of integers.

For BUILD_S$_l$, we use the exponent $k_{1\alpha}$ on the first variable in each polynomial to construct a full set of possible sequences for that polynomial. Those sets of sequences are then combined in SUM_SEQUENCES (alg. 2) using a Cartesian product over the sets in each multinomial.

When calculating multinomial coefficients, we use the form in Eq. (13) in terms of binomial coefficients with a fast, stable algorithm from Manolopoulos [2002].

In practice, many of the group operations π produce identical products $M_1^{r_1} M_2^{r_2} \ldots M_m^{r_m}$. Thus before computing any of the coefficients from the polynomials, we first form the polynomial products for each group operation and then add identical products together.

4. COMPUTATIONAL ORDER AND PERFORMANCE

The algorithm is structured around the *a priori* knowledge of the target stoichiometry. At the earliest possibility, we prune terms from individual polynomials that would not contribute to the final Pólya coefficient in the expanded product of polynomials (see Figure 3). Because the Pólya polynomial for each group operation is based on its disjoint-cyclic form, the complexity of the search can vary drastically from one group operation to the next. That said, it is common for groups to have several classes whose group operations (within each class) will have similar disjoint-cyclic forms and thus also scale similarly. However, from group to group, the set of classes and disjoint-cyclic forms may differ considerably; this makes it difficult to make a statement about the scaling of the algorithm in general. As such, we instead provide a formal, worst-case analysis for the algorithm's performance and supplement it with experimental examples. For these experiments, we crafted special groups with specific properties to demonstrate the various scaling behaviors as group properties change.

4.1. Worst-Case Scaling

Heuristically, the behavior of our algorithm should depend roughly on the size of the group: the number of permutations we have to analyze. That seems consistent with our experiments. But that can also be mitigated by noting that some groups of the same size have many more distinct cycle types than others. For example, if our group is generated by a single cycle of prime integer length p, then there are only two cycle types, despite the group having order p.

The majority of computation time should be spent in enumerating those matrices S and be proportional to the number of same (see Figure 4). Numerical experiments

ALGORITHM 1: Recursive Sequence Constructor

Procedure `initialize`(i, $k_{i\alpha}$, $M_\alpha^{r_\alpha}$, V_α,**T**)

Constructs a Sequence Object tree recursively for a single $M_\alpha^{r_\alpha}$ by filtering possible exponents on each x_i in the polynomial. The object has the following properties:

root: $k_{i\alpha}$, proposed exponent of x_i in $M_\alpha^{r_\alpha}$.
parent: proposed Sequence for $k_{i-1,\alpha}$ of x_{i-1}.
used: the sum of the proposed exponents to left of and including this variable $\sum_{j=1}^{i} k_{i\alpha}$.
i: index of variable in $M_\alpha^{r_\alpha}$ (column index).
$k_{i\alpha}$: proposed exponent of x_i in $M_\alpha^{r_\alpha}$ (matrix entry at $i\alpha$).
$M_\alpha^{r_\alpha}$: Pólya polynomial in P_π (14).
V_α: possible exponents for $M_\alpha^{r_\alpha}$ (10).
T: $\{T_i\}_{i=1}^{\xi}$ target stoichiometry.

```
if i = 1 then
    self.used ← self.root + self.parent.used
else
    self.used ← self.root
end
self.kids ← empty
if i ≤ ξ then
    for p ∈ V_α do
        rem ← p − self.root
        if 0 ≤ rem ≤ T_i and |rem| ≤ d_α r_α − self.used and |p − self.used| mod r_α = 0 then
            self.kids ⇐ initialize(i + 1, rem, M_α^{r_α}, V_α, T)
        end
    end
end
```

Function `expand`(*sequence*)

Generates a set of $S_{l\alpha}$ from a single Sequence object.

sequence: the object created using `initialize()`.

```
sequences ← empty
for kid ∈ sequence.kids do
    for seq ∈ expand(kid) do
        sequences ⇐ kid.root ∪ seq
    end
end
if len(sequence.kids) = 0 then
    sequences ← {kid.root}
end
return sequences
```

Function `build_`S_l(**k**, **V**, P_π, **T**)

Constructs S_l from $\{k_{1\alpha}\}_{\alpha=1}^{m}$ for a P_π (14).

k: $\{k_{1\alpha}\}_{\alpha=1}^{m}$ set of possible exponent values on the *first* variable in each $M_\alpha^{r_\alpha} \in P_\pi$.
V: $\{V_\alpha\}_{\alpha=1}^{m}$ possible exponents for each $M_\alpha^{r_\alpha}$ (10).
P_π: Pólya polynomial representation for a single operation π in the group **G** (14).
T: $\{T_i\}_{i=1}^{\xi}$ target stoichiometry.

```
sequences ← empty
for α ∈ {1 ... m} do
    seq ← initialize(1, k_{1α}, M_α^{r_α}, V_α, T)
    sequences ⇐ expand(seq)
end
return sequences
```

ALGORITHM 2: Coefficient Calculator

Function `sum_sequences`(S_l)
  Finds τ_l (17) *for* $S_l = \{S_{l\alpha}\}_{\alpha=1}^{m}$ (16)
  S_l: a set of lists (of exponent sequences $\{k_{i\alpha}\}_{i=1}^{\xi}$) for each polynomial $M_\alpha^{r_\alpha}$ in P_π (14).
  $K_l \leftarrow S_{l1} \times S_{l2} \times \cdots \times S_{lm} = \langle \{(k_{i\alpha})_{i=1}^{\xi}\}_{\alpha=1}^{m} \rangle_l$
  coeff $\leftarrow 0$
  for each $\{(k_{i\alpha})_{i=1}^{\xi}\}_{\alpha=1}^{m} \in K_l$ **do**
    if $\sum_{\alpha=1}^{m} k_{i\alpha} = T_i \ \forall\, i \in \{1 \ldots \xi\}$ **then**
      coeff $\leftarrow$ *coeff* $+ \prod_{\alpha=1}^{m} \binom{d_\alpha}{\{k_{i\alpha}\}_{i=1}^{\xi}}$
    end
  end
  return *coeff*
Function `coefficient`(**T**, P_π, **V**)
  Constructs $\mathbf{S} = \{S_l\}$ *and calculates* t_π (17)
  T: $\{T_i\}_{i=1}^{\xi}$ target stoichiometry.
  P_π: Pólya polynomial representation for a single operation π in the group **G** (14).
  V: $\{V_\alpha\}_{\alpha=1}^{m}$ possible exponents for each $M_\alpha^{r_\alpha}$ (10).
  if $m = 1$ **then**
    if $r_1 > T_i \ \forall\, i = 1..\xi$ **then**
      return 0
    else
      return $\binom{d_1}{T_1 T_2 \ldots T_\xi}$
    end
  else
    T $\leftarrow$ sorted(**T**)
    possible $\leftarrow V_1 \times V_2 \times \cdots \times V_m$
    coeffs $\leftarrow 0$
    for $\{k_{1\alpha}\}_{\alpha=1}^{m} \in$ *possible* **do**
      if $\sum_{\alpha=1}^{m} k_{1\alpha} = T_1$ **then**
        $S_l \leftarrow$ `build_`S_l`(`$\{k_{1\alpha}\}_{\alpha=1}^{m}$, **V**, P_π, **T**`)`
        coeffs $\leftarrow$ *coeffs*$+$ `sum_sequences(`S_l`)`
      end
    end
    return *coeffs*
  end

confirm[5] that the number of matrices scales exponentially with the number of colors (fixed group and number of elements in the set), linearly with the number of elements in the set (fixed number of colors and group), and is linear with the group size (fixed number of colors and elements in the set). The number of entries in the matrix S is $t\xi$ (see the discussion above Equation (3)) and the height of the entries is (roughly) bounded by the number of cycles and (very roughly) by the color frequencies divided by cycle lengths. This makes computing a time estimate based on these factors very difficult, but in the worst case, it could grow like the $t\xi$-th power of the average size of the entries, which will depend on the size of the target frequencies, and so on. This would be a very complex function to estimate, but we may expect it to grow exponentially for

[5]Figures are included in the code repository. See supplementary material.

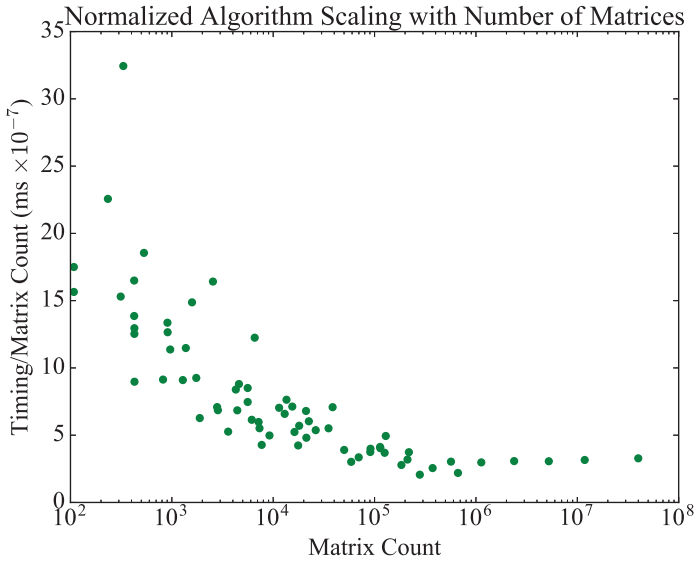


Fig. 4. Normalized algorithm scaling with the number of relevant matrices to enumerate. For large matrix counts, the behavior appears linear, supporting the hypothesis that the algorithm scales roughly with the number of matrices. The scatter is appreciable only for small matrix counts (less than 10^6).

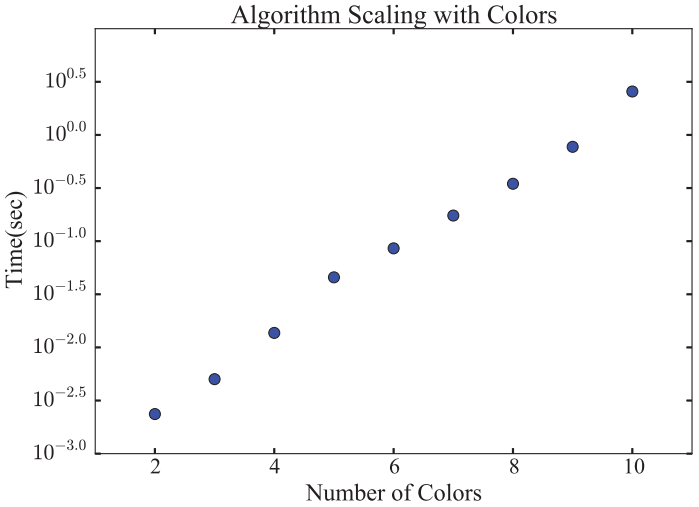


Fig. 5. Log plot of the algorithm scaling as the number of colors increases. Since the number of variables x_i in each polynomial increases with the number of colors, the combinatoric complexity of the expanded polynomial increases drastically with each additional color; this leads to an exponential scaling. The linear fit to the logarithmic data has a slope of 0.403.

very large input. We did not find that to be an impediment for the sizes of problems we needed to solve.

4.2. Experiments Demonstrating Algorithm Scaling

In Figure 5, we plot the algorithm's scaling as the number of colors in the enumeration increases (for a fixed group and number of elements). For each r-cycle in the disjoint-cyclic form of a group operation, we construct a polynomial with ξ variables, where ξ is the number of colors used in the enumeration. Because the group operation results in a product of these polynomials, increasing the number of colors by 1 increases the combinatoric complexity of the polynomial *expansion* exponentially. For

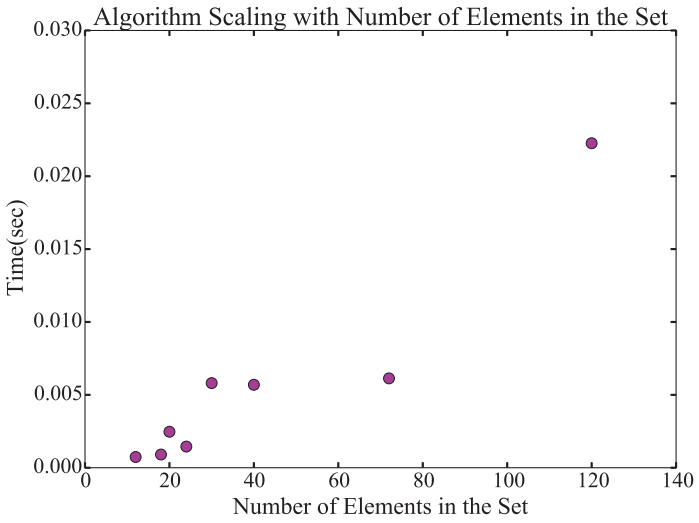


Fig. 6. Algorithm scaling as the number of elements in the finite set increases (for two colors). The Pólya polynomial arises from the group operations' disjoint-cyclic form, so more elements in the set results in a richer spectrum of possible polynomials multiplied together. Because of the algorithms aggressive pruning of terms, the exact disjoint-cyclic form of individual group operations has a large bearing on the algorithm's scaling. As such, it is not surprising that there is some scatter in the timings as the number of elements in the set increases.

this scaling experiment, we used the same transitive group acting on a finite set with 20 elements for each data point but increased the number of colors in the fixed color term T. We chose T by dividing the number of elements in the group as equally as possible; thus for two colors, we used $[10, 10]$; for three colors we used $[8, 6, 6]$, then $[5, 5, 5, 5]$, $[4, 4, 4, 4, 4]$, and so on. Figure 5 plots the $\log_{10}$ of the execution time (in ms) as the number of colors increases. As expected, the scaling is linear (on the log plot).

As the number of elements in the finite set increases, the possible Pólya polynomial representations for each group operation's disjoint-cyclic form increases exponentially. In the worst case, a group acting on a set with k elements may have an operation with k 1-cycles; on the other hand, that same group may have an operation with a single k-cycle, with lots of possibilities in between. Because of the richness of possibilities, it is almost impossible to make general statements about the algorithm's scaling without knowing the structure of the group and its classes. In Figure 6, we plot the scaling for a set of related groups (all are isomorphic to the direct product of $S_3 \times S_4$) applied to finite sets of varying sizes. Every data point was generated using a transitive group with 144 elements. Thus, this plot shows the algorithm's scaling when the group is the same and the number of elements in the finite set changes. Although the scaling appears almost linear, there is a lot of scatter in the data. Given the rich spectrum of possible Pólya polynomials that we can form as the set size increases, the scatter is not surprising.

Finally, we consider the scaling as the group size increases (Figure 7). For this test, we selected the set of unique groups arising from the enumeration of all derivative super structures of a simple cubic lattice for a given number of sites in the unit cell [Hart and Forcade 2008]. Since the groups are formed from the symmetries of real crystals, they arise from the semidirect product of operations related to physical rotations and translations of the crystal. In this respect, they have similar structure for comparison. In most cases, the scaling is obviously linear; however, the slope of each trend varies from group to group. This once again highlights the scaling's heavy dependence on

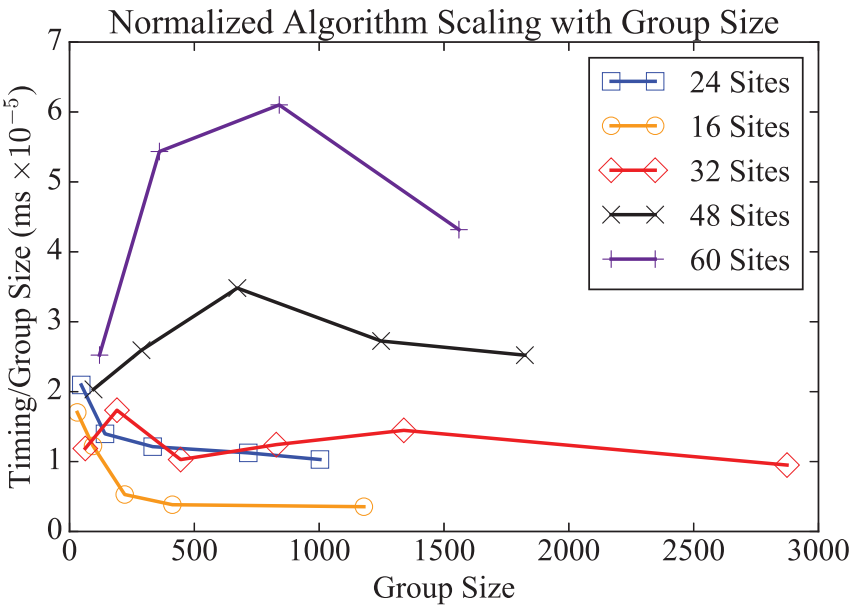


Fig. 7. Normalized algorithm scaling with group size for an enumeration problem from solid state physics [Hart and Forcade 2008]. We used the unique permutation groups arising from all derivative super structures of a simple cubic lattice for a given number of sites in the unit cell. The behavior is generally linear with increasing group size.

the specific disjoint-cyclic forms of the group operations. Even for groups with obvious similarity, the scaling may differ.

4.3. Comparison with Computer Algebra Systems

In addition to the explicit timing analysis and experiments presented above, we also ran a group of representative problems with our algorithm and MATHEMATICA (a common CAS). We also attempted the tests with MAPLE but were unable to obtain consistent results between multiple runs of the same problems.[6] So, we have opted to exclude the MAPLE timing results. For the comparison with MATHEMATICA, we used MATHEMATICA's `Expand` and `Coefficient` functions to return the relevant coefficient from the Pólya polynomial (see Figure 8).

5. SUMMARY

Until now, no low-level, numerical implementation of Pólya's enumeration theorem has been readily available; instead, a CAS was used to symbolically solve the polynomial expansion problem posed by Pólya. While CAS's are effective for smaller, simpler calculations, as the difficulty of the problem increases, they become impractical solutions. Additionally, codes that perform the actual enumeration of the colorings are often implemented in low-level codes, and interoperability with a CAS is not necessarily easy to automate.

We presented a low-level, purely numerical algorithm and code that exploits the properties of polynomials to restrict the combinatoric complexity of the expansion. By considering only those coefficients in the unexpanded polynomials that might contribute to the final answer, the algorithm reduces the number of terms that must be included to find the significant term in the expansion.

[6]The inconsistency manifests in MAPLE sometimes returning 0 instead of the correct result and sometimes running the same problem unpredictably in hours or seconds.

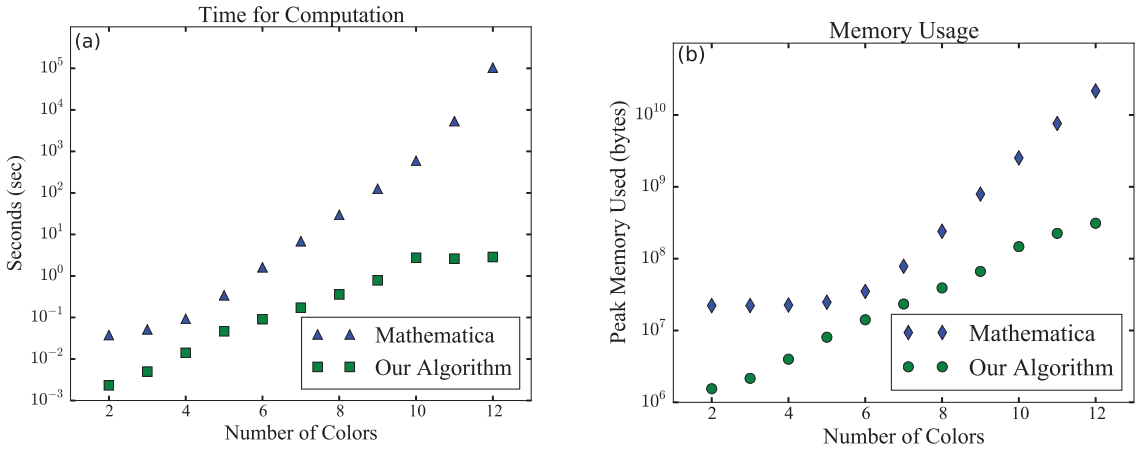


Fig. 8. Comparison of the CPU time (a) and memory usage (b) between the FORTRAN implementation of our algorithm and MATHEMATICA as the number of colors increases. These are the times needed to generate the data in Figure 5.

Because of the algorithm scaling's reliance on the exact structure of the group and the disjoint-cyclic form of its operations, a rigorous analysis of the scaling is not possible without knowledge of the group. Instead, we presented some numerical timing results from representative, real-life problems that show the general scaling behavior.

In contrast to the CAS solutions whose execution times range from milliseconds to hours, our algorithm consistently performs in the millisecond to second regime, even for complex problems. Additionally, it is already implemented in both high- and low-level languages, making it useful for confirming enumeration results. This makes it an effective substitute for alternative CAS implementations.

REFERENCES

Stefano Curtarolo, Gus L. W. Hart, Marco Buongiorno Nardelli, Natalio Mingo, Stefano Sanvito, and Ohad Levy. 2013. The high-throughput highway to computational materials design. *Nat. Mater.* 12, 3 (MAR 2013), 191–201. DOI:http://dx.doi.org/10.1038/NMAT3568

Kecai Deng and Jianguo Qian. 2014. Enumerating stereo-isomers of tree-like polyinositols. *J. Math. Chem.* 52, 6 (2014), 1581–1598.

Roberto Dovesi, Roberto Orlando, Alessandro Erba, Claudio M. Zicovich-Wilson, Bartolomeo Civalleri, Silvia Casassa, Lorenzo Maschio, Matteo Ferrabone, Marco De La Pierre, Philippe D'Arco, Yves Nol, Mauro Caus, Michel Rrat, and Bernard Kirtman. 2014. CRYSTAL14: A program for the ab initio investigation of crystalline solids. *Int. J. Quant. Chem.* 114, 19 (2014), 1287–1317. DOI:http://dx.doi.org/10.1002/qua.24658

Antoine Genitrini, Bernhard Gittenberger, Veronika Kraus, and Cécile Mailler. 2015. Associative and commutative tree representations for Boolean functions. *Theor. Comput. Sci.* 570 (2015), 70–101.

Modjtaba Ghorbani and Mahin Songhori. 2014. The enumeration of Chiral isomers of tetraammine platinum (II). *Match-Communications in Mathematical and in Computer Chemistry* 71, 2 (2014), 333–340.

Frank Harary. 1955. The number of linear, directed, rooted, and connected graphs. *Trans. Am. Math. Soc.* 78, 2 (1955), 445–463.

Gus L. W. Hart and Rodney W. Forcade. 2008. Algorithm for generating derivative structures. *Phys. Rev. B* 77 (Jun 2008), 224115. Issue 22. DOI:http://dx.doi.org/10.1103/PhysRevB.77.224115

Gus L. W. Hart and Rodney W. Forcade. 2009. Generating derivative structures from multilattices: Application to HCP alloys. *Phys. Rev. B* 80 (July 2009), 014120.

Gus L. W. Hart, Lance J. Nelson, and Rodney W. Forcade. 2012. Generating derivative structures for a fixed concentration. *Comp. Mat. Sci.* 59 (2012), 101–107. DOI:http://dx.doi.org/10.1016/j.commatsci.2012.02.015

B. A. Kennedy, D. A. McQuarrie, and C. H. Brubaker Jr. 1964. Group theory and isomerism. *Inorg. Chem.* 3, 2 (1964), 265–268.

Peter Lackner, Harald Fripertinger, and Gerhard Nierhaus. 2015. Peter Lackner/tropical investigations. In *Patterns of Intuition*. Springer, Berlin, 279–313.

Yannis Manolopoulos. 2002. Binomial coefficient computation: Recursion or iteration? *ACM SIGCSE Bulletin InRoads* 34 (Dec 2002). Issue 4. DOI: http://dx.doi.org/10.1145/820127.820168

James McGrane, Sanjaye Ramgoolam, and Brian Wecht. 2015. Chiral ring generating functions & branches of moduli space. *arXiv preprint arXiv:1507.08488* (2015).

Sami Mustapha, Philippe DArco, Marco De La Pierre, Yves Nol, Matteo Ferrabone, and Roberto Dovesi. 2013. On the use of symmetry in configurational analysis for the simulation of disordered solids. *J. Phys.: Condens. Matter* 25, 10 (2013), 105401. http://stacks.iop.org/0953-8984/25/i=10/a=105401.

George Pólya. 1937. Kombinatorische anzahlbestimmungen fr gruppen, graphen und chemische verbindungen. *Acta Math.* 68, 1 (1937), 145–254.

George Pólya and Ronald C. Read. 1987. *Combinatorial Enumeration of Groups, Graphs, and Chemical Compounds* (1987).

Jianguo Qian. 2014. Enumeration of unlabeled uniform hypergraphs. *Discr. Math.* 326, 1 (2014), 66–74.

R. W. Robinson, F. Harry, and A. T. Balaban. 1976. The numbers of chiral and achiral alkanes and monosubstituted alkanes. *Tetrahedron* 32, 3 (1976), 355–361.

Masahiko Taniguchi, Sarah Henry, Richard J. Cogdell, and Jonathan S. Lindsey. 2014. Statistical considerations on the formation of circular photosynthetic light-harvesting complexes from rhodopseudomonas palustris. *Photosynth. Res.* 121, 1 (2014), 49–60.

J. Tura, R. Augusiak, A. B. Sainz, B. Lücke, C. Klempt, M. Lewenstein, and A. Acín. 2015. Nonlocality in many-body quantum systems detected with two-body correlators. *arXiv preprint arXiv:1505.06740* (2015).

Received December 2015; revised May 2016; accepted June 2016

CHAPTER 3

Assessing Robustness in Machine Learning Models

As discussed above, Cluster Expansion (CE) is a machine learning model whose representation relies on compositional degrees of freedom only. This means that it does an excellent job of approximating properties on-lattice. However, in practice we are most interested in "relaxed" structures, those for which atoms are allowed to move off of ideal lattice sites to lower their energy. Once the atoms move off-lattice, CE may no longer be a good basis for expanding the property of interest.

In practice, if the structures within an alloy system relax "too much", the CE fails and has no predictive power. Before this study was published [58], there was no reliable way to know *a priori* whether the CE would fail for a given system. This meant that several hundred expensive DFT calculations may be produced to train a CE that will not approximate the physics well. We set out to answer the following questions:

- How do we quantify "too much" relaxation?
- Is it possible to know in advance (i.e., with only a few DFT calculations) whether the CE will converge rapidly enough to justify the additional calculations?

By combining a thorough analysis over hundreds of systems with our group's expertise in Compressive Sensing [61, 62], we were able to show that it is possible to detect slow convergence with relatively few calculations. In the context of machine learning for materials discovery, this study serves as a gentle reminder not to forget rigor in applying models. The CE community applied the methodology for many years before someone asked enough questions to get at the fundamental problems behind CE convergence.

Inasmuch as CE will soon be replaced by alloy potentials, the analysis serves mostly as a guide to understanding historical results published using CE.

For this article, I performed the numerical error analyses for experimental and analytic distributions presented in Section III. This includes interpreting the results in light of Bayesian Compressive Sensing and using that framework to generate predictive heuristics for failure of CE. Using these heuristics, it is possible to predict whether a CE is worth pursuing by generating comparatively few training configurations and analyzing the behavior of the learning rate with respect to sparsity. Andrew Nguyen performed the analyses in Section II and we both drafted the introduction and conclusion.

The following article is reproduced with permission. A license is on file with the Department of Physics and Astronomy.

Robustness of the cluster expansion: Assessing the roles of relaxation and numerical error

Andrew H. Nguyen,[1] Conrad W. Rosenbrock,[1] C. Shane Reese,[2] and Gus L. W. Hart[1,*]
[1]*Department of Physics and Astronomy, Brigham Young University, Provo, Utah 84602, USA*
[2]*Department of Statistics, Brigham Young University, Provo, Utah 84602, USA*
(Received 11 January 2017; revised manuscript received 13 June 2017; published 12 July 2017)

Cluster expansion (CE) is effective in modeling the stability of metallic alloys, but sometimes cluster expansions fail. Failures are often attributed to atomic relaxation in the DFT-calculated data, but there is no metric for quantifying the degree of relaxation. Additionally, numerical errors can also be responsible for slow CE convergence. We studied over one hundred different Hamiltonians and identified a heuristic, based on a normalized mean-squared displacement of atomic positions in a crystal, to determine if the effects of relaxation in CE data are too severe to build a reliable CE model. Using this heuristic, CE practitioners can determine a priori whether or not an alloy system can be reliably expanded in the cluster basis. We also examined the error distributions of the fitting data. We find no clear relationship between the type of error distribution and CE prediction ability, but there are clear correlations between CE formalism reliability, model complexity, and the number of significant terms in the model. Our results show that the *size* of the errors is much more important than their distribution.

DOI: 10.1103/PhysRevB.96.014107

I. INTRODUCTION

Increases in computational power and algorithmic advancements are making many computational materials problems more tractable. For example, density functional theory (DFT) is used to assess the stability of potential metal alloys with high accuracy. However, the computational costs of DFT prevents exhaustive exploration of all possible configurations of a system. In certain cases, one can map first-principles results on to a faster Hamiltonian, the cluster expansion (CE) [1–3]. Over the past 30 years, CE has been used in combination with first-principles calculations to predict the stability of metal alloys [4–16], to study the stability of oxides [17–21], and to model interaction and ordering phenomena at metal surfaces [22–26]. Numerical error and *relaxation* effects decrease the predictive power of CE models. The aim of this paper is to demonstrate the effects of both and to provide a heuristic so one can know when a reliable CE model can be expected for a particular material system.

CE treats alloys as a purely configurational problem, i.e., a problem of decorating a fixed lattice with the alloying elements [1,2]. However, CE models are usually trained with data taken from "relaxed" first-principles calculations where the individual atoms assume positions that minimize the total energy, displaced from ideal lattice positions. Unfortunately, cluster expansions of systems with larger lattice relaxation converge more slowly than cluster expansions for unrelaxed systems [27]. In fact, CEs with increased relaxation may fail to converge altogether. No rigorous description of conditions for when the CE breakdown occurs exists in the CE literature.

A persistent question in the CE community regards the impact of relaxation on the accuracy of the cluster expansion. Some proponents of CE argue that the CE formalism holds even when the training structures are relaxed because there is a one-to-one correspondence in configurational space between relaxed and unrelaxed structures. This is an assumption. Independent of whether or not it is true, the relevant issue is not the correspondence but the sparsity of the expansion. In this paper, we demonstrate a relationship between relaxation and sparsity in the CE model. As relaxation increases, CE sparsity and the accuracy of CE predictions decreases.

In addition to the effects of relaxation, we also examine the impact of numerical error on the reliability of the CE fits. There are several sources of numerical error: approximations to the physics of the model, the number of k points, the smearing method, basis set sizes and types, etc. Most previous studies [28–30] only examine the effect of Gaussian errors on the CE model, but Arnold *et al.* [28] also investigated systematic error (round-off and saturation error). They showed that, above a certain threshold, the CE model fails to recover the correct answer, that is, the CE model started to incorporate spurious terms (i.e., sparsity was reduced). A primary question that we seek to answer is whether the shape of the error distribution impacts predictive performance of a CE model.

In this study, we quantify the effects of: (1) relaxation, by comparing CE fits for relaxed and unrelaxed data sets and (2) numerical error, by adding different error distributions (i.e., Gaussian, skewed, etc.) to ideal CE models. We study more than one hundred Hamiltonians ranging from very simple pair potentials to first-principles DFT Hamiltonians. We present a heuristic for judging the quality of the CE fits. We find that a small mean-squared displacement is indicative of a good CE model. In agreement with past studies, we show that the predictive power of CE is lowered when the level of error is increased. We find that there is no clear correlation between the shape of the error profile and the CE predictive power. It is possible to decide whether the computational cost of generating CE fitting data is worthwhile by examining the degree of relaxation in a smaller set of 50–150 structures.

II. RELAXATION

Relaxation is distinct from numerical error—it is not an error—but it has a similar negative effect. When relaxations are significant, it is less likely that a reliable CE model

*gus.hart@gmail.com

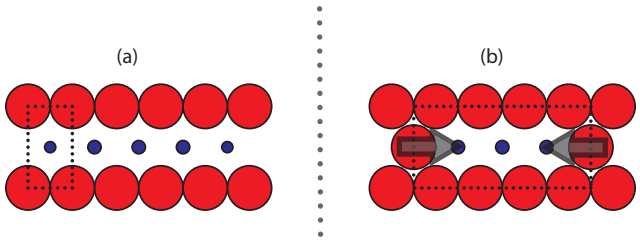


FIG. 1. Symmetry-allowed distortions for two different unit cells. The atomic positions of the cell on the left do not have any symmetry-allowed degrees of freedom, but the aspect ratio of the unit cell is allowed to change. For the unit cell on the right, the horizontal positions of the atoms in the middle layer may change without destroying the symmetry. (The unit cell aspect ratio may also change.)

exists. Relaxation is a systematic form of distortion, the local adjustment of atomic positions to accommodate atoms of different sizes. Atoms "relax" away from ideal lattice sites to reduce the energy, with larger atoms taking up more room, and smaller atoms giving up volume. The type of relaxations (i.e., the distortions that are possible) for a particular unit cell are limited by the symmetry of the initially undistorted case, as shown in Fig. 1. In the rectangular case (left), the unit cell aspect ratio may change without changing the initial rectangular symmetry. At the same time, the position of the blue atom is *not allowed* to change because doing so would destroy rectangular symmetry. In contrast, the two blue atoms in the similar structure shown in the right panel of the figure can move horizontally without reducing the symmetry.

Conceptually, the cluster expansion is a technique that describes the local environment around an atom and then sums up all the "atomic energies" (environments in a unit cell) to determine a total energy for the unit cell. For the cluster expansion model to be sparse—to be a predictive model with few parameters—it relies on the premise that any specific local neighborhood contributes the same atomic energy to the total energy regardless of the crystal in which it is embedded. For example, the top row of Fig. 2 shows the same local environment (denoted by the hexagon around the central blue atom) embedded in two distinct crystals. If the contribution of this local environment to the total energy is the same in both cases, then the cluster expansion of the energy will be sparse.

The effect of relaxation on the sparsity becomes clear in the bottom row of Fig. 2. In the left-hand case [panel (a)], the crystal relaxes dramatically and the central blue atom is now *fourfold coordinated* entirely by red atoms. By contrast, in the right-hand case [panel (b)], a collapse of the layers is not possible and the blue atoms are allowed by symmetry to move closer to each other. From the point of view of the cluster expansion, the local environments of the central blue atom are the same for both cases. This fact, that two different relaxed local environments have identical descriptions in the cluster expansion basis, leads to a slow convergence of cluster expansion models. The problem is severe when the atomic mismatch is large and relaxations are significant (i.e., when atoms move far from the ideal lattice positions.)

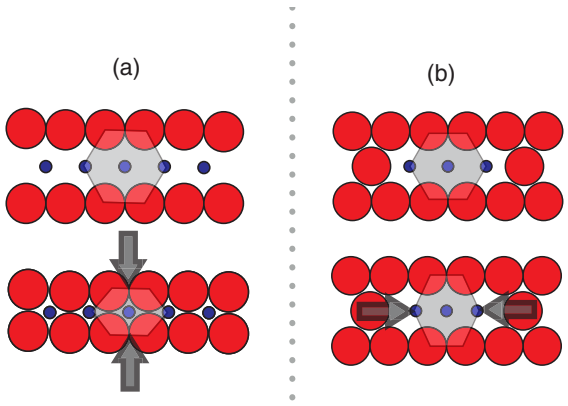


FIG. 2. Relaxation scheme. The top images show the original unrelaxed configurations, while the bottom figures show the relaxed configuration. The left images (a) shows the relaxation where the hexagon is contracted as shown by the black arrows in the bottom left figure. The relaxation in the right images (b) is restricted to displacement of the blue atoms as shown by the black arrows in the bottom right figure.

A. Methodology

We investigated the predictive power of cluster expansions using data from more than one hundred Hamiltonians generated from density functional theory (DFT), the embedded atom method, Lennard-Jones potential, and Stillinger-Weber potential. To investigate the effects of relaxation, we examined different metrics to measure the degree of atomic relaxation in a crystal configuration.

1. Hamiltonians

First-principles DFT calculations have been used to simulate metal alloys and for building cluster expansion models [7,9–14]. However, DFT calculations are too expensive to extensively examine the relaxation in many different systems (lattice mismatch). Thus, we examine other methods such as the embedded atom method (EAM) which is a multibody potential. The EAM potential is a semiempirical potential derived from first-principles calculations. EAM potentials of metal alloys such as Ni-Cu, Ni-Al, and Cu-Al have been parameterized from DFT calculations and validated to reproduce their experimental properties such as bulk modulus, elastic constants, lattice constants, etc. [31]. EAM potentials are computationally cheaper, allowing us to explore the effects of relaxation for large training sets; however, we are limited by the number of EAM potentials available.

Therefore, we also selected two classical potentials, Lennard-Jones (LJ) and Stillinger-Weber (SW), to adequately examine various degrees of relaxation, which can be varied using free parameters in each model. The Lennard-Jones potential is a pairwise potential. Using the LJ potential, we can model a binary (A_xB_{1-x}) alloy with different lattice mismatch and interaction strength between the A and B atoms by adjusting the σ parameter in the model. Additionally, we also examined the Stillinger-Weber potential which has a pair term and an angular (three-body) term. In attempting to determine the conditions under which the CE formalism

breaks down, we implemented a set of parameters in the SW potential where the angular dependent term could be turned on/off using the λ coefficient [32]. For example, depending on the strength of λ, the local atomic environment in a relaxed, two-dimensional structure switches between three-, four- and six-fold coordination. When the system relaxes to a different coordination, the CE fits would no longer be valid or at least not sparse.

All first-principles calculations were performed using the Vienna *ab initio* simulation package (VASP) [33–36]. We used the projector-augmented-wave (PAW) [37] potential and the exchange-correlation functional proposed by Perdew, Burke, and Ernzerhof (PBE) [38]. In all calculations, we used the default settings implied by the high-precision option of the code. Equivalent k-point meshes were used for Brillioun zone integration to reduce numerical errors [39]. We used 1728 (12^3) k points for the pure element structures and an equivalent mesh for the binary alloy configurations. Each structure was allowed to fully relax (atomic, cell shape, and cell volume).

Relaxation was carried out using molecular dynamics simulations for EAM, LJ, and SW potentials. Two molecular dynamics packages were used to study the relaxation: GULP [40,41] and LAMMPS [42]. Details for the LJ, SW, and EAM potentials and the DFT calculations can be found in the Supplemental Material [43].

2. Cluster expansion setup

The universal cluster expansion (UNCLE) software [44–46] was used to generate 1000 derivative superstructures each of face-centered cubic (FCC), body-centered cubic (BCC), and hexagonal closed-packed (HCP) lattice. For the DFT calculations, we used only 500 structures instead of 1000 due to the computational cost. We generated a set of 1100 clusters, ranging from two-body up to six-body interactions. 100 independent CE fits were performed for each system (Hamiltonian and lattice).

We briefly discuss some important details about cluster expansion here, but for a more complete description, see the Supplemental Material [43] and past works [1,4,10,13,47–50]. Cluster expansion is a generalized Ising model with many-body interactions. The cluster expansion formalism allows one to map a physical property, such as E, to a configuration ($\vec{\sigma}$):

$$E_i^{\mathrm{CE}} = \Sigma_i J_i \Pi_i(\vec{\sigma}), \tag{1}$$

where E is energy, Π is the correlation matrix (basis), and J is coefficient or effective cluster interaction (ECI).

When constructing a CE model, we are solving for the effective cluster interactions or Js. We used the compressive sensing (CS) framework to solve for these coefficients [13,50]. The key assumption in compressive sensing is that the solution vector has few nonzero components, i.e., the solution is sparse [51,52]. The CS framework guarantees that the sparse solution can be recovered from a limited number of DFT energies. Using the Js, we can build a CE model to interpolate the configuration space.

Each CE fit used a random selection of 25% of the data for training and 75% for validation. Results were averaged over the 100 CE fits with error bars computed from the standard deviation. We defined the percent error as a ratio of the prediction root mean squared error (RMS) over the standard deviation of the input energies, percent error = $\mathrm{RMS/STD}(\mathrm{E_{input}}) \times 100\%$. This definition of percent error allowed us to consistently compare different systems.

3. Relaxation metrics

Currently, there is no standard measure to indicate the degree of relaxation. We evaluated different metrics as a measure of the relaxation: normalized mean-squared displacement, Ackland's order parameter [53], difference in Steinhardt order parameter (D_6) [54], SOAP [55], and the centrosymmetry parameter [56]. We compared the metrics across various Hamiltonians to find a criterion that is independent of the potentials and systems [43]. We found that none of these metrics are descriptive/general enough except for the normalized mean-squared displacement.

4. Normalized mean-squared displacement (NMSD)

To measure the relaxation of each structure/configuration, we used the mean-squared displacement (MSD) to measure the displacement of an atom from its reference position, i.e., the unrelaxed atomic position. The MSD metric is implemented in the LAMMPS software [42], which also incorporates the periodic boundary conditions to properly account for displacement across a unit cell boundary. The MSD is the total squared displacement averaged over all atoms in the crystal:

$$\mathrm{MSD} = \frac{1}{N_{\mathrm{atom}}} \sum_{\mathrm{atom}} \sum_{X=x,y,z} (X[t] - X[0])^2, \tag{2}$$

where X represents the Cartesian components of each atom position, t is the final relaxed configuration, and 0 is the initial unrelaxed configuration. Additionally, we defined a normalized mean-square displacement (NMSD) percent:

$$\mathrm{NMSD} = \frac{\mathrm{MSD}}{V^{2/3}} \times 100\% \tag{3}$$

which is the ratio of MSD to volume of the system. This allows for a relaxation comparison parameter that is independent of the overall scale.

B. Results and discussions

To explore the effects of relaxation on CE predictability, we examine relaxation in various systems from very high accuracy (DFT) to very simple, tunable systems (LJ and SW potentials). We examine more than one hundred different Hamiltonians and we find several common trends among the different systems.

In most cases, we find that the relaxed CE fits are worse (higher prediction error and higher number of coefficients) than the unrelaxed ones. For example, Fig. 3 shows the cluster expansion fitting for unrelaxed and relaxed data sets of Ni-Cu alloy system using DFT and EAM with two different primitive lattices, FCC and BCC. Because Ni-Cu alloys are naturally FCC-like and the lattice mismatch is small, the training structures for the FCC-based training structures have small relaxations, whereas BCC-based training structures have large relaxations. The contrast between the two cases demonstrates the effect of atomic relaxations. As Fig. 3 shows, Ni-Cu alloy

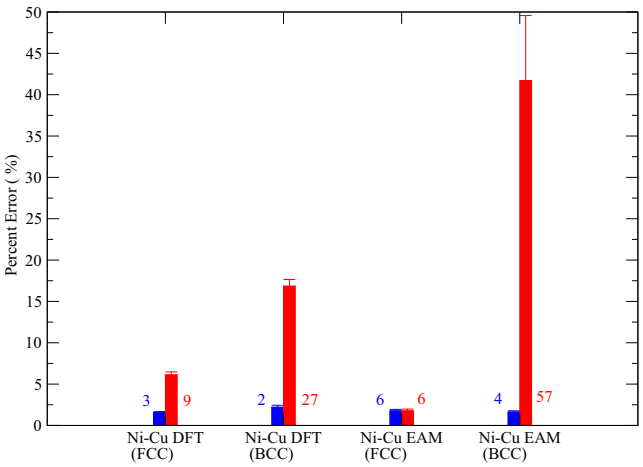


FIG. 3. Cluster expansion fits for Ni-Cu alloy using DFT or EAM potential. Each bar represents the average percent error and error bar (standard deviations) for 100 independent CE fits. The blue bars represent the unrelaxed CE fits, while the red bars represent the relaxed CE fits. The colored number represents the average number of coefficients used in the CE models. When the configurations are relaxed, we find that the CE fits are often worse (higher prediction error and higher number of Js) than the unrelaxed system. However, we show that in one case (Ni-Cu EAM) the unrelaxed and relaxed CE fits are identical (same error and same number of coefficients) and this is due to a small relaxation.

fitting for a FCC lattice is below 10% error, while BCC fitting result in more Js and higher percent error (above 10%) [57]. We find similar results in the relaxation of Ni-Cu alloy using first-principles DFT and EAM potential. The difference between relaxed and unrelaxed CE fits are negligible when relaxations are small. This is shown in Fig. 3 for the relaxation of FCC superstructures using a Ni-Cu EAM potential.

Figure 3 shows that relaxation is often associated with reduced sparsity (increased cardinality of Js) [58]. One possible implication is that a number of coefficients (J) could be used to evaluate the predictive performance of the CE fits. The number of coefficients used in the fits (such as in Fig. 3) is a simple way to determine whether or not a CE fit can be trusted. Figures 4(b) and 4(f) show similar clusters across the 100 independent CE fittings; thus, vertical lines indicate the presence of the same cluster across all CE fits. When the fit is good, only a small subset of clusters is needed [Fig. 4(b)]. On the other hand, Fig. 4(f) shows some common clusters in all of the CE fits with several additional clusters. Figure 5(a) shows the correlation of the percent error with the number of terms in the expansion. We find that as the number of coefficients increases the percent error increases. However, this is not a sufficient metric as shown in Fig. 5(a) where the number of coefficient varies a lot. Nonetheless, the number of coefficients may be used as a general, quick test.

The degree of relaxation is crucial to define whether or not the CE model is accurate or not. However, there is no standard for *when* cluster expansion fails due to relaxation. Thus far, we have made some remarks about relaxation and CE fits. But the question of how much relaxation is allowed has not been addressed. By examining a few metrics: NMSD, SOAP [55], D6 [54], Ackland [53], and centrosymmetry [56], we find that there is a relationship between degree of relaxation and the quality of CE fits. As shown in the Supplemental Material, we have used these metrics to investigate over 100+ systems (different potentials, lattice mismatches, and interaction strengths). Here, we present a heuristic to measure the degree of relaxation based on the NMSD.

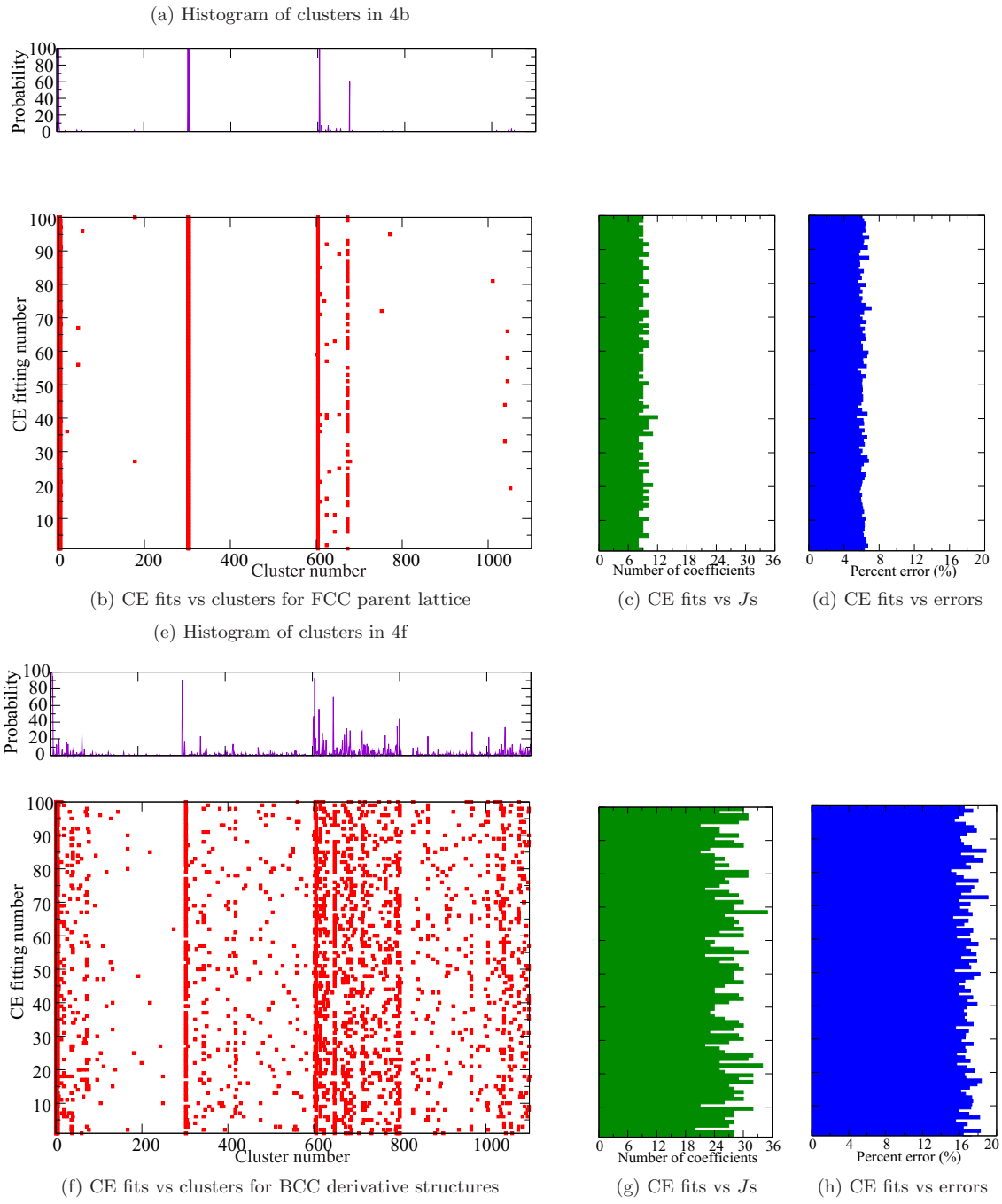
In general, cluster expansion will fail when the relaxation is large. Figure 5(b) shows that a small NMSD weakly correlates with a small number of coefficients. However, Fig. 6 highlights the correlation between degree of relaxation and prediction error. There is a roughly linear relationship between the degree of relaxation and the CE prediction. We partition the quality of the CE models into three regions: good (NMSD $<$ 0.1%), maybe (0.1% $\leqslant$ NMSD $\leqslant$ 1%), and bad (NMSD $>$ 1%). The "maybe" region is the gray area where the CE fit can be good or bad. This metric provide a heuristic to evaluate the reliability of the CE models, i.e., any systems that exhibit high relaxation will fail to provide an accurate CE model.

III. NUMERICAL ERROR

As we have shown in the previous section, greater relaxation results in worse CE fitting. In addition to the effects of relaxation, we now investigate the effects of numerical error on reliability of CE models. The distinction between relaxation "error" and numerical error is that the former is inherent in the data used to train the CE model. Numerical error can be completely eliminated, in principle. Numerical error arises from various sources such as the number of k points, the smearing method, minimum force tolerance, basis set sizes and types, etc. These errors are not stochastic errors or measurement errors; they arise from tuning the numerical methods. We assume that the relaxation-induced change in energy for each structure is an *error term* that the CE fitting algorithm must handle. The collection of these "errors" from all structures in the alloy system then form an error profile (or distribution). Using the simulated relaxation error profiles from the previous section together with common analytic distributions, we built "toy" CE models with known coefficients. We then examined whether or not the shape of the error distribution affects the CE predictive ability.

A. Methodology

The numerical errors in DFT calculations are largely understood, but it is difficult to disentangle the effects of different, individual error sources. Instead of studying the effects of errors separately, we added different distributions of error to a "toy" model in order to imitate the aggregate effects of the numerical error on CE models. Hence, we opt to simplify the problem by creating a "toy" problem for which the exact answer is known. To restrict the number of independent variables, we formulated a "toy" cluster expansion model by selecting five nonzero values for a subset of the total clusters. Using this toy CE, we predicted a set of energies y for 2000 known derivative superstructures of an FCC lattice. These y values are used as the true energies for all subsequent analysis. We added error to y, chosen from either: (1) "simulated"



(a) Histogram of clusters in 4b

(b) CE fits vs clusters for FCC parent lattice

(c) CE fits vs Js

(d) CE fits vs errors

(e) Histogram of clusters in 4f

(f) CE fits vs clusters for BCC derivative structures

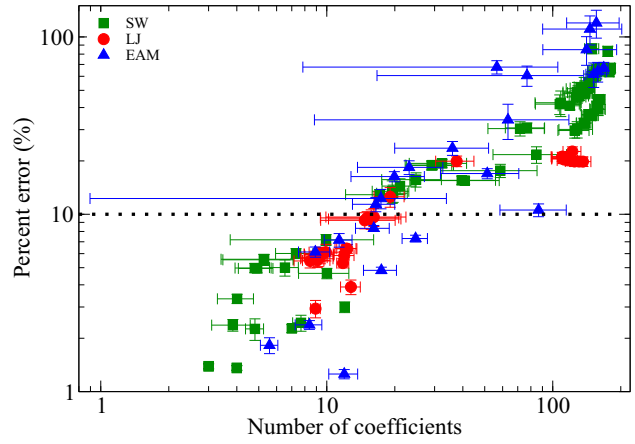
(g) CE fits vs Js

(h) CE fits vs errors

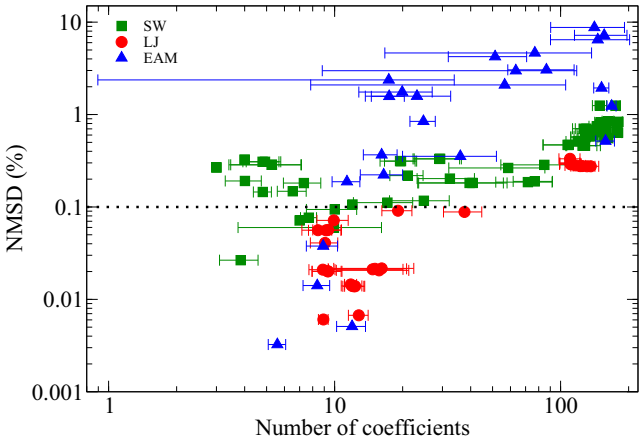
FIG. 4. CE fitting and relaxation of Ni-Cu alloys (DFT calculations) using FCC derivative superstructures and BCC derivative superstructures. Shown in Figs. 4(b) and 4(a) are the 100 CE fits and the histogram of the clusters used for the FCC lattices, while plots 4(f) and 4(e) are for the BCC lattice. The errors and coefficients are shown in 4(c) and 4(d) for the FCC structures and in 4(g) and 4(h) for the BCC lattice. The plot shows that the number of clusters used in fitting is small when cluster expansion fitting is good (error is on average 6.03% for FCC derivative structures). However, the CE fitting of the BCC parent lattice is worse at 16.70% compared to FCC at 6.03%. More coefficients are used when CE fails. The increased number of Js and error indicate a bad CE fitting model as shown by plots 4(g) and 4(h). Figure 4(e) shows only a few significant terms with many other clusters used sparingly in the fits.

distributions obtained by computing the difference between relaxed and unrelaxed energies predicted by either DFT, EAM, LJ, or SW models (Fig. 7) or (2) common analytic distributions (Fig. 8).

To generate the simulated distributions, we chose a set of identical structures and fitted them using a variety of classical and semiclassical potentials, and quantum mechanical calculations using VASP. For each of the potentials we selected, we calculated an unrelaxed total energy y for each structure and then performed relaxation to determine the lowest energy state $\tilde{y}$. The difference between these two energies ($\Delta y = \tilde{y} - y$) was considered to be the "relaxation" error.



(a) Error vs Js



(b) Relaxation vs Js

FIG. 5. Plot 5(a) displays the CE fitting error vs the number of coefficients, while plot 5(b) highlights the relationship between number of coefficients and relaxation. The dashed line approximates what we consider as the maximum acceptable error for a CE model (10%). The dashed line in Fig. 5(b) marks the estimated threshold for acceptable relaxation level. Each symbol represents 100 independent CE fittings for each Hamiltonian. Higher error correlates with a higher number of coefficients.

Certain assumptions are usually made about the error in the signal, namely that it is Gaussian. The original CS paradigm proves that the ℓ_2 error for signal recovery obeys [52]:

$$||x^* - x||_{\ell_2} \leqslant C_0 \cdot ||x - x_S||/\sqrt{S} + C_1 \cdot \epsilon, \tag{4}$$

where ϵ bounds the amount of error in the data, x^* is the CS solution, x is the true solution, and x_S is the vector x with all but the largest S components set to zero. This shows that, *at worst, the error in the recovery is bounded by a term proportional to the error*. For our plots of this error, we first normalized Δy so that $\epsilon \equiv \text{normalized}(\Delta y) \in [0,1]$ using

$$\epsilon = \frac{y - \min(y)}{\max(y) - \min(y)}. \tag{5}$$

Not surprisingly, the various potentials produced different error profiles.

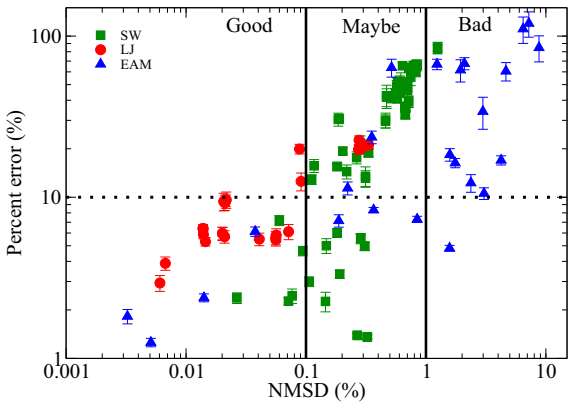


FIG. 6. Relationships between relaxation and CE fitting reveal a heuristic for determining the quality of a CE model. This graph shows the CE fitting error vs normalized mean square displacement (NMSD). Each mark represents 100 individual CE fittings for each system (potentials and parameters). As the NMSD (relaxation) increases, the CE fitting error increases for various systems and potentials. Using the relaxation metric, the quality/ reliability of the CE fits can be divided into three regions: good, maybe, and bad CE model. The solid black lines indicates these three areas.

The expectation value of the distributions was set to be a percentage of the average, unrelaxed energy across all structures. Thus, "15% error" means that each unrelaxed energy was changed by adding a randomly drawn value from a distribution with an expectation value of 15% of the mean energy. We performed CE fits as a function of the %-error added (2, 5, 10, and 15%) for each distribution. Although we only present the 15% error results in the next section, all results at different error levels can be found in the supporting information [43]. For each data point, we performed 100 independent CE fits and used the mean and standard deviation to produce the values and error bars for the plots.

B. Results and discussions

As shown in Fig. 9, the error is weakly uniform across all (analytic and simulated) distributions, implying that there is no correlation between specific distribution and error. None of the normal quantifying descriptions of distribution shape (e.g., width, skewness, kurtosis, standard deviation, etc.) show a correlation with the CE prediction error. The error increased proportionally with the level of error in each system (2, 5, 10, and 15% error). We therefore turn to the compressive sensing (CS) formalism for insight.

The theorems of Tao and Candés [51] guarantee that the solution for an underdetermined CS problem can be recovered *exactly* with overwhelming probability provided:

(1) The solution is sparse within the chosen representation basis.

(2) Sufficient data points, sampled independent and identically distributed (i.i.d).

(3) The sensing and representation bases are maximally incoherent.

If all of these conditions are met, we know that CS will provide a solution that is very close to the true answer. Conversely, if CS cannot converge to a good solution, it means that one of

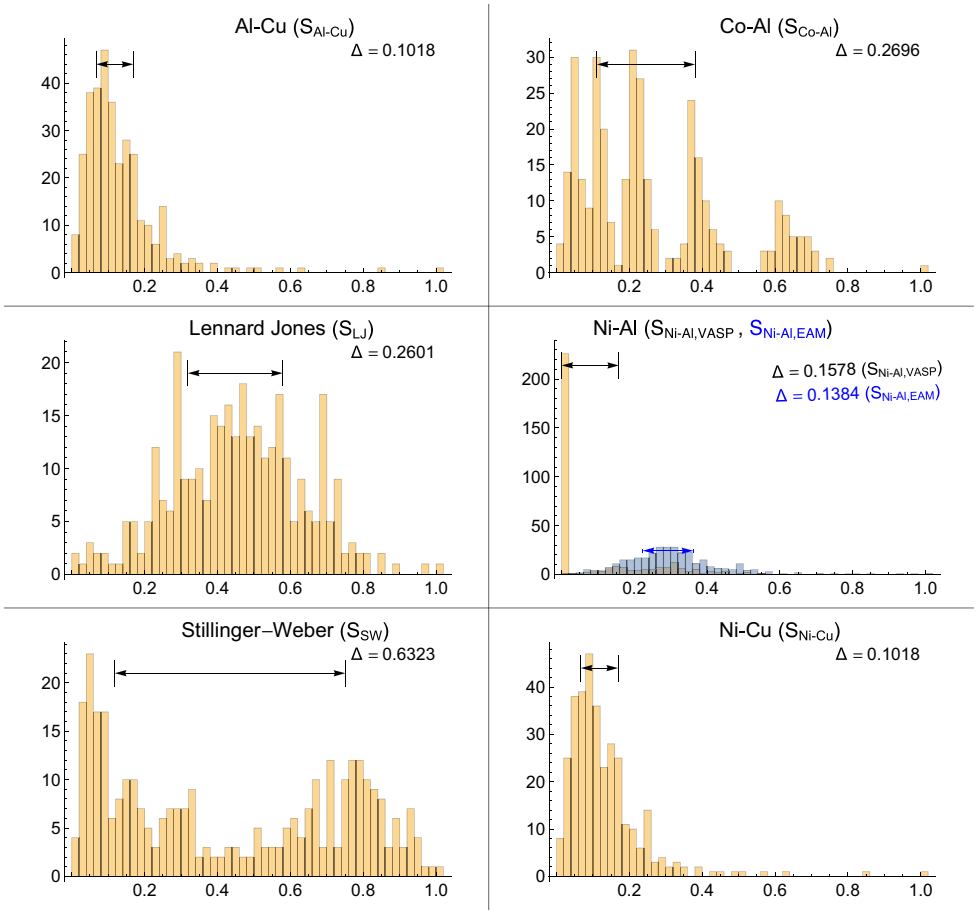


FIG. 7. Distributions from real relaxations using classical and semiclassical potentials, as well as DFT calculations. The distributions are all normalized to fall within 0 and 1. The widths Δ were calculated by taking the difference between the 25th and 75th percentiles.

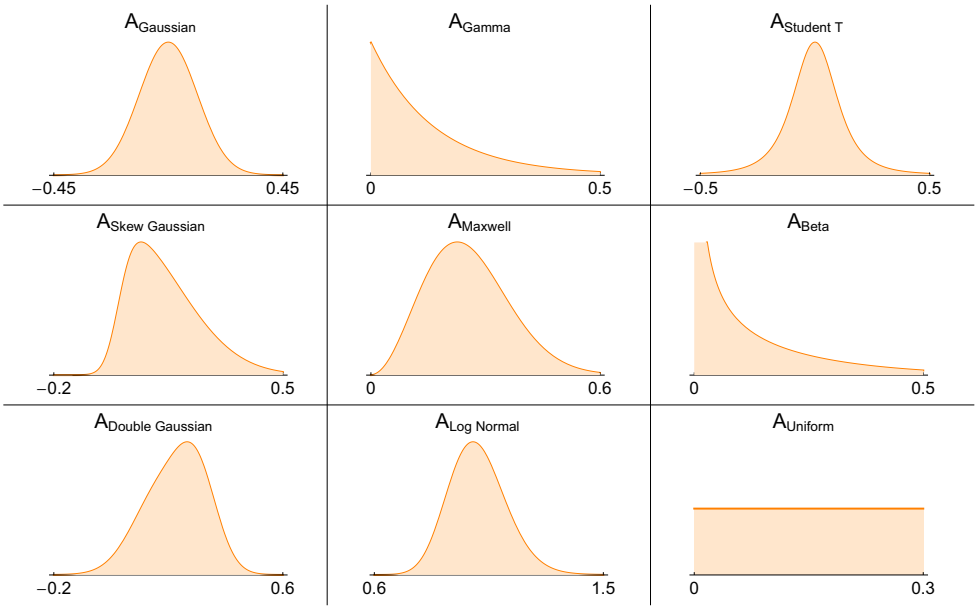


FIG. 8. The analytic, equal width distributions used for adding error to the toy model CE fit.

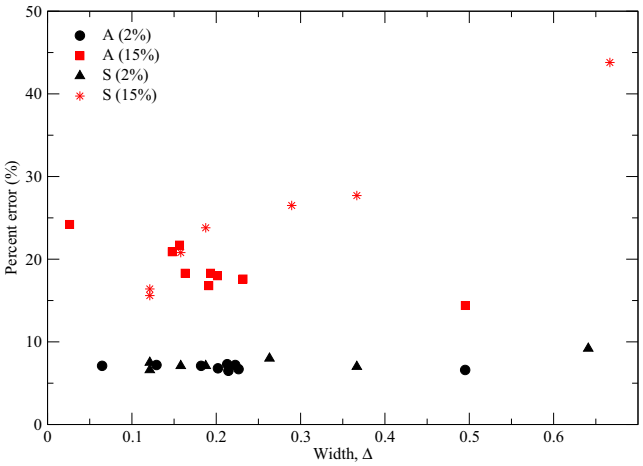


FIG. 9. Comparison of the predictive error in CE fits as the shape of relaxation error changes. (A) refers to the analytic distribution while (S) refers to simulated distribution. The fits are ordered from lowest to highest distribution width. Fits were averaged over 100 randomly selected subsets with 500/2000 data points used for training; the remaining 1500 were used to verify the model's predictions. The black and red colored symbols represent 2% and 15% error levels, respectively. The circles and triangles represent the analytical and simulated distributions, respectively. Higher error produces higher prediction errors.

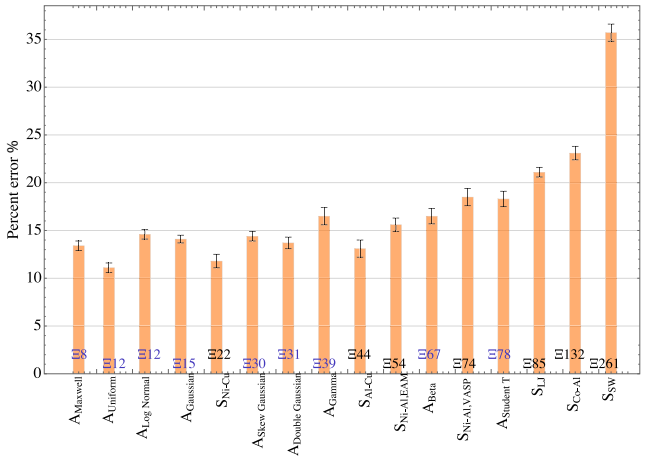


FIG. 10. Prediction error over 65% of the structures for the "toy" cluster expansion (at 15% error added). The systems are ordered by Ξ, which is the total number of unique clusters used by any of the 100 CE fits for the system. This ordering shows a definite trend with increasing Ξ.

these conditions has been violated. We have control over the number of training points, and the incoherence of the sensing-representation bases. However, we *cannot* control whether the true physical solution is sparsely represented for relaxed systems. This suggests a useful connection between the CS framework and the robustness of CE: if CS cannot reproduce a good CE fit (quantified below), then sparsity has been lost.

In the CS framework, the foundational assumption is that of sparsity, meaning that the compressed signal (or cluster expansion) requires only a few terms to accurately represent the true signal (physics). Thus, the number of terms recovered by CS to produce the CE is a good measure of the quality of the CS fit. This begs the question: Can we use the number of terms within the CS framework to heuristically predict *in advance* whether the CE fit will converge well?

In answering the question of predictability for a good CE fit, we define three new quantities:

(1) Ξ: total number of unique clusters used over 100 CE fits of the same dataset. We also call this the model complexity.

(2) ∉: number of "exceptional" clusters. These are clusters that show up fewer than 25 times across 100 fits, implying that they are not responsible for representing any real physics in the signal, but are rather included because the CE basis is no longer a sparse representation for the relaxed alloy system. They are sensitive to the training/fitting structures.

(3) Λ: number of *significant* clusters in the fit; essentially just the total number of unique clusters minus the number of "exceptional" clusters, $\Lambda = \Xi - \notin$.

In the relaxation section, we showed that the average number of coefficient is not sufficient to determine the quality of the CE model. Here, we decompose the number of Js into three new quantities to provide additional insights into the reliability of the CE fits. In Fig. 10, we plot the CE error, ordered by model complexity and show that it reproduces the trend identified by the number of coefficients (indeed they are intimately related, Ξ being the statistically averaged number of coefficients across many fits). An ordering by the number of exceptional clusters ∉ produces an identical trend, showing that it may also serve to quantify a good fit [43].

As indicated earlier, all these experiments were performed for a *known* CE model that had five nonzero terms. Additional insight is gained by plotting the errors, ordered by Λ, the number of significant clusters (Fig. 11). Figure 11 shows that in almost all cases, once we remove the exceptional clusters ∉, the remaining model is almost *exactly* the known CE model that we started with. The CS framework provides a rigorous mathematical framework for this statement because it guarantees to

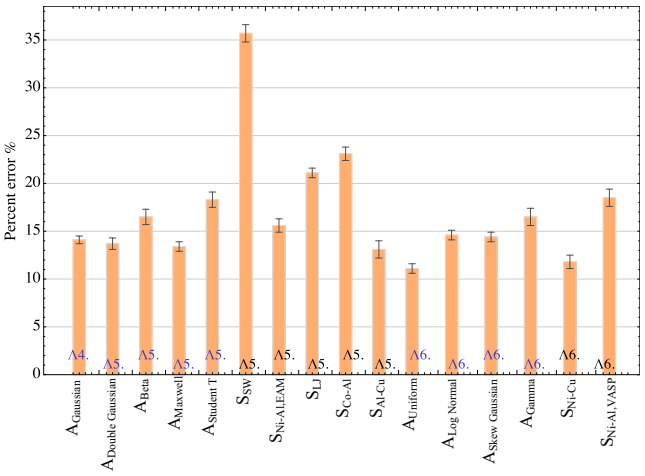


FIG. 11. Prediction error over 65% of the structures for the "toy" CE model (at 15% error added). The errors are ordered by Λ, the number of significant terms in the expansion. As expected, the values are close to the known model complexity (5 terms) and the ordering once more appears random.

exactly recover the original function with high probability as long as we have enough measurements and our representation basis is truly sparse. Once the cluster expansion stops converging, we lose sparsity and CS fails. This gives us confidence to use the CS framework as a predictive tool for CE robustness.

Provided the training structures are independent and identically distributed, we do not necessarily need hundreds of costly DFT calculations to tell us that the CE will not converge. Using our toy CE model, we discovered that for all error distributions, a training set size of 50 data points was sufficient to recover the actual model complexity (five terms) [43]. For actual DFT calculations, where relaxation was known to disrupt CE convergence, we saw a similar trend with about 100 data points needed to identify whether the CE would converge with more data or not.

We conclude that CE robustness for relaxed systems can be predicted with a *much* smaller number of data points than is typically needed for a good CE fit (on the order of 5–10% from our experience) [59]. The proposed heuristic to verify convergence of the relaxed CE, when trained with a limited dataset, is to examine the values of Λ and Ξ over a large number of independent fits. If the number of the exceptional clusters $\notin$ is significant compared to Λ, then it is likely that the CE will *not* converge on a larger dataset as shown in Fig. 12. Figure 13 highlights the CE fitting as a function of training set size. We observe small relaxation (black curve) correlates with a small number of coefficients; thus the CE can fit using a small number of Js even with 5% (25) to 10% (50) of the structures. On the other hand, red and blue curves, which have high relaxation, do not converge. By using a small relaxed dataset (50 to 100 structures), we can predict whether or not the computational cost of relaxing *many* structures is fruitful.

IV. CONCLUSIONS

Relaxation and error decrease the reliability of the cluster expansion fit because the CE model is no longer sparse. Nevertheless, until now, there has been no measure of relaxation that provides a heuristic as to when the CE fitting data is reliable. Using four different Hamiltonians (first-principles, Lennard-Jones, Stillinger-Weber, and embedded atom method), we show that the normalized mean-squared displacement of alloy configuration is a good measure of relaxation and CE predictability. A small displacement percent, e.g., less than 0.1%, will usually generate a reliable CE model. The number of cluster terms in the CE models can be an indicator of how well cluster expansions perform; we find that models with a large number of parameters have poor predictive capability and tend not to converge, even with more training data. CE tends to fail when the number of Js exceeds 80.

In our error analysis, we investigated the ability of the compressive sensing framework to obtain fits to a toy, cluster expansion model as the energy of relaxation changes in a predictable way. We used 16 relaxation error distributions (both analytic and simulated) and compared the prediction errors of the resulting CE fits for the relaxed vs unrelaxed case. No clear correlation appears between the statistical measures of distribution shape and the predictive errors. However, there are clear correlations between the predictive error, the complexity of the resulting CE model, and the number of significant terms in that model.

We cannot use the relaxation distributions alone to determine the viability of a CE fit in advance. However, the analysis does reveal that the majority of the clusters used by the unrelaxed CE fit will also be present in the relaxed case (albeit with adjusted J values) if the CE fit is viable. This suggests that it may be possible to decide whether the computational cost of full CE is worthwhile by making predictions for a few relaxed systems (50–100) and determining whether the error remains small enough.

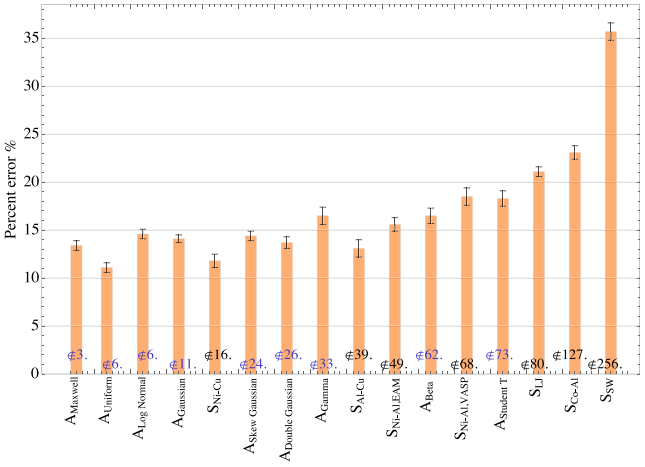


FIG. 12. Plot of predictive error over 65% of the structures for the "toy" problem (at 15% error added). The systems are ordered by $\notin$ the number of clusters that were used less than 25 times across all 100 CE fits. These are considered exceptions to the overall fit for the system. As for Fig. 10, there is a definite trend toward higher error for systems with more exceptional clusters.

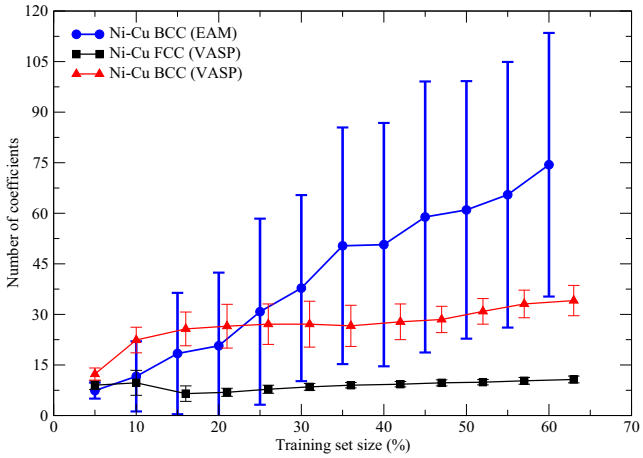


FIG. 13. For a reliable CE model, the number of coefficients converges as a function of the training set size. A total of 500 structure were available for training. The number of coefficients in a fit and its error bars give us an indication of the predictive power of CE with only a small training set. The black curve represents a good CE fit; only 25 to 50 (or 5 to 10%) of training structures were needed. On the other hand, the red and blue curves show that CE fails to fit the data due to a slowly converging expansion. The error bars on the blue points indicate extremely bad fitting.

ACKNOWLEDGMENTS

The authors thank V. Blum, L. J. Nelson, and M. K. Transtrum for useful discussions. This work was supported by funding from Office of Naval Research (MURI N00014-13-1-0635). We thank the Fulton Supercomputing Lab at Brigham Young University for allocation of computing time.

[1] J. M. Sanchez, F. Ducastelle, and D. Gratias, Generalized cluster description of multicomponent systems, Physica A: Statistical and Theoretical Physics **128**, 334 (1984).
[2] J. M. Sanchez, Cluster expansions and the configurational energy of alloys, Phys. Rev. B **48**, 14013 (1993).
[3] J. M. Sanchez, Cluster expansion and the configurational theory of alloys, Phys. Rev. B **81**, 224202 (2010).
[4] J. W. D. Connolly and A. R. Williams, Density-functional theory applied to phase transformations in transition-metal alloys, Phys. Rev. B **27**, 5169 (1983).
[5] L. G. Ferreira, A. A. Mbaye, and A. Zunger, Effect of chemical and elastic interactions on the phase diagrams of isostructural solids, Phys. Rev. B **35**, 6475 (1987).
[6] L. G. Ferreira, A. A. Mbaye, and A. Zunger, Chemical and elastic effects on isostructural phase diagrams: The e-G approach, Phys. Rev. B **37**, 10547 (1988).
[7] L. G. Ferreira, S.-H. Wei, and A. Zunger, First-principles calculation of alloy phase diagrams: The renormalized-interaction approach, Phys. Rev. B **40**, 3197 (1989).
[8] Z. W. Lu, S.-H. H. Wei, A. Zunger, S. Frota-Pessoa, and L. G. Ferreira, First-principles statistical mechanics of structural stability of intermetallic compounds, Phys. Rev. B **44**, 512 (1991).
[9] C. Wolverton and A. Zunger, Ni-Au: A testing ground for theories of phase stability, Comput. Mater. Sci. **8**, 107 (1997).
[10] A. van de Walle and G. Ceder, Automating first-principles phase diagram calculations, J. Phase Equilib. **23**, 348 (2002).
[11] S. V. Barabash, V. Blum, S. Müller, and A. Zunger, Prediction of unusual stable ordered structures of Au-Pd alloys via a first-principles cluster expansion, Phys. Rev. B **74**, 035108 (2006).
[12] M. Asato, H. Takahashi, T. Inagaki, N. Fujima, R. Tamura, and T. Hoshino, Cluster expansion approach for relative stability among different atomic structures in alloys: An approach from a dilute limit, Materials Transactions **48**, 1711 (2007).
[13] L. J. Nelson, V. Ozoliš, C. S. Reese, F. Zhou, and G. L. W. Hart, Cluster expansion made easy with Bayesian compressive sensing, Phys. Rev. B **88**, 155105 (2013).
[14] P. H. Gargano, P. R. Alonso, and G. H. Rubiolo, Ordering in the solid solution $U(Al, Si)_3$, Procedia Materials Science **9**, 239 (2015).
[15] M. Aldegunde, N. Zabaras, and J. Kristensen, Quantifying uncertainties in first-principles alloy thermodynamics using cluster expansions, J. Comput. Phys. **323**, 17 (2016).
[16] R. Chinnappan, B. K. Panigrahi, and A. van de Walle, First-principles study of phase equilibrium in TiV, TiNb, and TiTa alloys, Calphad **54**, 125 (2016).
[17] G. Ceder, A. Van Der Ven, C. Marianetti, and D. Morgan, First-principles alloy theory in oxides, Modell. Simul. Mater. Sci. Eng. **8**, 311 (2000).
[18] A. Seko, K. Yuge, F. Oba, A. Kuwabara, and I. Tanaka, Prediction of ground-state structures and order-disorder phase transitions in II-III spinel oxides: A combined cluster-expansion method and first-principles study, Phys. Rev. B **73**, 184117 (2006).
[19] A. Seko, A. Togo, F. Oba, and I. Tanaka, Structure and Stability of a Homologous Series of Tin Oxides, Phys. Rev. Lett. **100**, 045702 (2008).
[20] I. Tanaka, A. Togo, A. Seko, F. Oba, Y. Koyama, and A. Kuwabara, Thermodynamics and structures of oxide crystals by a systematic set of first principles calculations, J. Mater. Chem. **20**, 10335 (2010).
[21] I. Tanaka, A. Seko, A. Togo, Y. Koyama, and F. Oba, Phase relationships and structures of inorganic crystals by a combination of the cluster expansion method and first principles calculations, J. Phys.: Condens. Matter **22**, 384207 (2010).
[22] S. Müller, Bulk and surface ordering phenomena in binary metal alloys, J. Phys.: Condens. Matter **15**, R1429 (2003).
[23] R. Drautz, R. Singer, and M. Fähnle, Cluster expansion technique: An efficient tool to search for ground-state configurations of adatoms on plane surfaces, Phys. Rev. B **67**, 035418 (2003).
[24] S. Müller, M. Stöhr, and O. Wieckhorst, Structure and stability of binary alloy surfaces: Segregation, relaxation, and ordering from first-principles calculations, Appl. Phys. A **82**, 415 (2006).
[25] L. M. Herder, J. M. Bray, and W. F. Schneider, Comparison of cluster expansion fitting algorithms for interactions at surfaces, Surf. Sci. **640**, 104 (2015).
[26] R. Tanaka, K. Takeuchi, and K. Yuge, Application of grid increment cluster expansion to modeling potential energy surface of cu-based alloys, Mater. Trans. **56**, 1077 (2015).
[27] D. B. Laks, L. G. Ferreira, S. Froyen, and A. Zunger, Efficient cluster expansion for substitutional systems, Phys. Rev. B **46**, 12587 (1992).
[28] B. Arnold, A. D. Ortiz, G. L. W. Hart, and H. Dosch, Structure-property maps and optimal inversion in configurational thermodynamics, Phys. Rev. B **81**, 094116 (2010).
[29] A. Díaz-Ortiz and H. Dosch, Noise filtering of cluster expansions in multicomponent systems, Phys. Rev. B **76**, 012202 (2007).
[30] A. Díaz-Ortiz, H. Dosch, and R. Drautz, Cluster expansions in multicomponent systems: Precise expansions from noisy databases, J. Phys.: Condens. Matter **19**, 406206 (2007).
[31] S. M. Foiles, M. I. Baskes, and M. S. Daw, Embedded-atom-method functions for the fcc metals Cu, Ag, Au, Ni, Pd, Pt, and their alloys, Phys. Rev. B **33**, 7983 (1986).
[32] F. H. Stillinger and T. A. Weber, Computer simulation of local order in condensed phases of silicon, Phys. Rev. B **31**, 5262 (1985).
[33] G. Kresse and J. Hafner, Ab initio molecular dynamics for liquid metals, Phys. Rev. B **47**, 558 (1993).
[34] G. Kresse and J. Hafner, Ab initio molecular-dynamics simulation of the liquid-metalamorphous-semiconductor transition in germanium, Phys. Rev. B **49**, 14251 (1994).

[35] G. Kresse and J. Furthmüller, Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set, Comput. Mater. Sci. **6**, 15 (1996).

[36] G. Kresse and J. Furthmüller, Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set, Phys. Rev. B **54**, 11169 (1996).

[37] P. E. Blöchl, Projector augmented-wave method, Phys. Rev. B **50**, 17953 (1994).

[38] J. P. Perdew, K. Burke, and M. Ernzerhof, Generalized Gradient Approximation Made Simple, Phys. Rev. Lett. **77**, 3865 (1996).

[39] S. Froyen, Brillouin-zone integration by Fourier quadrature: Special points for superlattice and supercell calculations, Phys. Rev. B **39**, 3168 (1989).

[40] J. D. Gale, GULP: A computer program for the symmetry-adapted simulation of solids, J. Chem. Soc., Faraday Trans. **93**, 629 (1997).

[41] J. D. Gale and A. L. Rohl, The general utility lattice Program (GULP), Mol. Simul. **29**, 291 (2003).

[42] S. Plimpton, Fast Parallel Algorithms for Short Range Molecular Dynamics, J. Comput. Phys. **117**, 1 (1995).

[43] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevB.96.014107 for further details on cluster expansion, the parameterized potentials, other relaxation metrics, and additional plots at different levels of error.

[44] G. L. W. Hart and R. W. Forcade, Algorithm for generating derivative structures, Phys. Rev. B **77**, 224115 (2008).

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

[46] D. Lerch, O. Wieckhorst, G. L. W. Hart, R. W. Forcade, and S. Müller, UNCLE: A code for constructing cluster expansions for arbitrary lattices with minimal user-input, Modell. Simul. Mater. Sci. Eng. **17**, 055003 (2009).

[47] L. G. Ferreira, S.-H. Wei, and A. Zunger, Stability, Electronic structure, and phase diagrams of novel inter- semiconductor compounds, International Journal of High Performance Computing Applications **5**, 34 (1991).

[48] A. Zunger, L. G. Wang, G. L. W. Hart, and M. Sanati, Obtaining Ising-like expansions for binary alloys from first principles, Modell. Simul. Mater. Sci. Eng. **10**, 685 (2002).

[49] A. van de Walle, Methods for First-Principles Alloy Thermodynamics, JOM **65**, 1523 (2013).

[50] L. J. Nelson, G. L. W. Hart, F. Zhou, and V. Ozoliš, Compressive sensing as a paradigm for building physics models, Phys. Rev. B **87**, 035125 (2013).

[51] E. J. Candes and T. Tao, Near-optimal signal recovery from random projections: Universal encoding strategies?, IEEE Trans. Inf. Theory **52**, 5406 (2006).

[52] E. J. Candes and M. B. Wakin, An introduction to compressive sampling, IEEE Signal Processing Magazine **25**, 21 (2008).

[53] G. J. Ackland and A. P. Jones, Applications of local crystal structure measures in experiment and simulation, Phys. Rev. B **73**, 054104 (2006).

[54] P. J. Steinhardt, D. R. Nelson, and M. Ronchetti, Bond-orientational order in liquids and glasses, Phys. Rev. B **28**, 784 (1983).

[55] A. P. Bartók, R. Kondor, and G. Csányi, On representing chemical environments, Phys. Rev. B **87**, 184115 (2013).

[56] C. L. Kelchner, S. J. Plimpton, and J. C. Hamilton, Dislocation nucleation and defect structure during surface indentation, Phys. Rev. B **58**, 11085 (1998).

[57] In our experience, a percent error above 10% often gives an unreliable CE model.

[58] Optimization of the sparsity is discussed in Refs. [13] and [50].

[59] For typical binary CEs we typically need about 300–500 structures to get a good fit, which is then verified using an additional 200+ DFT calculations.

CHAPTER 4

Bridging the Experiment-Theory Gap

If a computational method predicts stable alloys at a certain composition, it does not necessarily guarantee that the alloy will be easy to make. Experimentally, we can control the composition and the temperatures/pressures at which the solution anneals. If an accurate phase diagram is available, the process is streamlined since the exact structure at each point in the diagram is known and the temperatures at which phase transitions occur are also known. Unfortunately, phase diagrams are not always reliable, especially when they are drawn from knowledge of only a few experimentally verified points.

Machine learning models such as Cluster Expansion (CE) can approximate the energy of a structure given its configuration. As soon as a fast method is available for calculating energies, statistical simulations (such as Monte Carlo) can be executed to approximate phase transition temperatures and the structures that occur at the stable minima on either side of the transition. Since computational experiments can always run to completion, this provides a valuable source of information if the models are accurate.

In contrast, if an experiment doesn't anneal for long enough at the transition temperature, the structure may be stuck in an intermediate phase or a combination of pre- and post-transition phases. If the phase diagram is incomplete, this presents a special problem because characterization of the sample for such intermediate phases may produce an unknown structure.

This study leveraged a fast CE model and metropolis Monte Carlo simulation to approximate the energy of $CuPt_3$ as a function of temperature. Using a group-theoretical analysis of the structures with order parameters of the symmetry-breaking subgroups, we were able to identify transition phases computationally. It is significant because it clarifies a phase diagram published earlier that was ambiguous as to the particular structures. We were able to verify the structures at experimental points in the phase diagram and then go beyond experiment by characterizing intermediate structures at the phase transitions.

The methodology presented in this work can be generally applied to any computational modeler that tries to discover structures within a phase diagram. Tools to discover the order parameters for distorted sub-groups can be applied generally to new systems [63]. Combined with automated alloy potential generation and nested sampling for phase discovery [64–66], we plan to use this methodology for structure characterization in computed phase diagrams.

Bridging the gap between experiment and theory requires tight collaboration. From the experimental side, sample preparation and characterization are both time-consuming, often requiring many iterations. The theoretical analysis in this case required combining advanced group theory with machine learning. I performed Monte Carlo simulations using a machine-learned Cluster Expansion model for CuPt to characterize the temperature-dependent structure of the phases. This leveraged structural characterization tools from group theory provided by Stokes and Campbell and Cluster Expansion expertise from Nelson and Hart. Combining these existing tools in a new way created a reusable tool, adding to the significance of the theoretical contribution to this work. The remaining authors were involved in experimental synthesis and characterization.

The following article is reproduced with permission. A license is on file with the Department of Physics and Astronomy.

Available online at www.sciencedirect.com

ScienceDirect

Acta Materialia 73 (2014) 326–336

Acta MATERIALIA

www.elsevier.com/locate/actamat

Revisiting the $CuPt_3$ prototype and the $L1_3$ structure

Chumani Mshumi [a], Candace I. Lang [a,1], Lauren R. Richey [b], K.C. Erb [b], Conrad W. Rosenbrock [b], Lance J. Nelson [b], Richard R. Vanfleet [b], Harold T. Stokes [b], Branton J. Campbell [b], Gus L.W. Hart [b,*]

[a] *Centre for Materials Engineering, Department of Mechanical Engineering, University of Cape Town, South Africa*
[b] *Department of Physics and Astronomy, Brigham Young University, Provo, UT 84602, USA*

Received 20 December 2013; received in revised form 9 March 2014; accepted 14 March 2014
Available online 9 May 2014

Abstract

Experimentally and computationally, the structure of Pt–Cu at 1:3 stoichiometry has a convoluted history. The $L1_3$ structure has been predicted to occur in binary alloy systems, but has not been linked to experimental observations. Using a combination of electron diffraction, synchrotron X-ray powder diffraction, and Monte Carlo simulations, we demonstrate that it is present in the Cu–Pt system at 1:3 stoichiometry. We also find that the 4-atom, fcc superstructure $L1_3$ is equivalent to the large 32-atom orthorhombic superstructure reported in older literature, resolving much of the confusion surrounding this composition. Quantitative Rietveld analysis of the X-ray data and qualitative trends in the electron-diffraction patterns reveal that the secondary $X_1^+(a, 0, 0)$ order parameter of the $L1_3$ phase is unexpectedly weak relative to the primary $L_1^+(a, a, 0, 0)$ order parameter, resulting in a partially-ordered $L1_3$ ordering, which we conclude to be the result of kinetic limitations. Monte Carlo simulations confirm the formation of a large cubic superstructure at high temperatures, and its eventual transformation to the $L1_3$ structure at lower temperature, but also provide evidence of other transitional orderings.

Keywords: Diffraction; Ordering; Alloys; Platinum; Catalysis

1. Introduction

The structure of Cu–Pt at 1:3 stoichiometry was first reported by Schneider and Esch in 1944 [1] as an orthorhombic ordering that can be visualized as a 32-atom fcc supercell (see Fig. 1(a)). This result was followed by conflicting reports over the next three decades [2–7]. In 1974, Miida and Watanabe resolved the contradictions by demonstrating that the 32-atom orthorhombic ordering is indeed the stable structure in $CuPt_3$ at room temperatures, but that rhombohedral and cubic orderings also appear in the phase diagram at adjacent compositions and temperatures [8].

On the theoretical side, the story is also rather convoluted. Based on theoretical considerations, Khachaturyan's formalism is consistent with a 4-atom orthorhombic unit cell as the prototype $CuPt_3$ structure [9] but he incorrectly cites the 32-atom cell of *cubic* symmetry proposed by Tang (see Ref. [2]), which can be condensed to an 8-atom primitive cell, and diagrams a *tetragonal* 32-atom cell that seems to be a hybrid of the orthorhombic and cubic structures; see also Fig. 1 panes (b) and (d). The Khachaturyan

* Corresponding author.
E-mail addresses: candace.lang@gmail.com (C.I. Lang), gus.hart@byu.edu (G.L.W. Hart).
[1] Current address: Department of Engineering, Macquarie University, Sydney, Australia.

http://dx.doi.org/10.1016/j.actamat.2014.03.029
1359-6454/

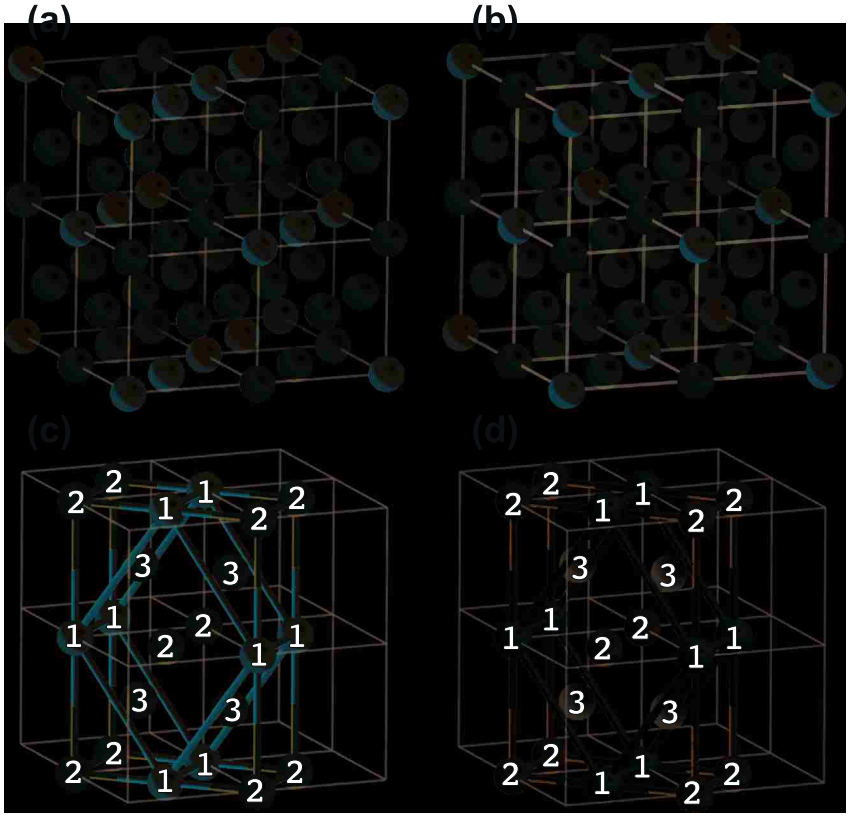


Fig. 1. (a) The originally proposed [1] orthorhombic structure for $CuPt_3$, where Pt atoms are shown in blue and Cu atoms in red. This 32-atom cell is not primitive (or even conventional) but clearly shows the ordering motif and the underlying fcc parent lattice. (b) A $CuPt_3$ ordering model proposed by Tang [2], wherein both the $2 \times 2 \times 2$ supercell and the parent cell have face-centered cubic symmetry. Despite the 3:1 stoichiometry, one of the sites (indicated by purple atoms) is disordered, i.e. randomly occupied by both Pt and Cu atoms. (c) The conventional *C*-centered orthorhombic unit cell of $L1_3$ first discussed by Khachaturyan and proposed as the structure of $CuPt_3$, which is crystallographically equivalent to that of (a). The smaller 4-atom primitive cell is indicated by red lines. The numerals 1, 2, and 3 indicate distinct Wyckoff positions and are discussed later. (d) The structure of $L1_3$ as determined from present experiments. Gray atoms indicate a disordered site, while purple atoms indicate a partially-ordered site that is Cu-rich but not pure Cu.

orthorhombic cell[2] is now referred to by the *Strukturbericht* symbol $L1_3$.[3]

Although, it was never recognized experimentally, $L1_3$ has long been employed as a hypothetical structure in the alloy community, due both to the work of Khachaturyan and the seminal work of Kanamori and Kakehashi where it is derived as a possible alloy structure on purely theoretical grounds [11]. No work in the experimental literature has discussed the primitive unit cell of the (original, 1944) 32-atom orthorhombic supercell, and no work in the computational/theoretical literature has recognized that the $L1_3$ structure is related to the experimental structure of Schneider [12]. Here, we make the simple observation that the ordering conveyed by Schneider's original orthorhombic 32-atom supercell is in fact equivalent to the 4-atom $L1_3$ structure (shown in Fig. 1(c)), which is also orthorhombic.

The $L1_3$ structure emerges naturally from the concentration wave formalism [9,13] as a Lifshitz structure associated with the *k*-points $\left(\frac{1}{2},\frac{1}{2},\frac{1}{2}\right)$ and $(1,0,0)$. It also emerged independently from the cluster expansion community, where ordered superstructures of a disordered fcc parent were enumerated for ground state searches using Ising models of alloys [11,14–17]; but it was not considered to be especially interesting until it was predicted to be a stable

[2] Although the 4-atom cell that Khachaturyan introduced in Ref. [9] is orthorhombic, he referred to it as a tetragonal cell. Subsequent references to this special Lifshitz structure in the theoretical literature recognize it as orthorhombic.

[3] This arbitrary Strukturbericht-like designation is motivated by other official designations. Strukturbericht designations for pure elements start with A; the face centered cubic (fcc) structure is A1. One-to-one structure designations start with B; the NaCl structure is designated B1, the 1 coming from A1, indicating fcc. Alloy structures are indicated by a beginning L and fcc-based alloys L1, again the 1 indicating fcc. A second, subscripted number in the Strukturbericht symbol, $L1_x$, indicates the order of discovery or assignment. For example, the designation for CuAu is $L1_0$, indicating that CuAu was the first fcc-based alloy to be given a symbol. Next, was the designation for equiatomic PtCu as $L1_1$, and so forth. The designation of the 4-atom, orthorhombic structure shown in Fig. 1(d) as $L1_3$ follows this convention. However, adding to the potential confusion, the Strukterbericht symbol $L1_3$ was already used in 1931 [10] for another structure (an 8-atom cell of 1:1 stoichiometry but not that referred to as D4 in the modern community). Apparently, the previous use of the symbol was forgotten by the modern community. It is possible that there are instances, besides [10] in the literature (past or current), where $L1_3$ designation is used for the previous structure but the authors are not aware of any.

configuration in the Ag–Pd system in 2001 (where it was called $L1_1^+$ because of its relationship to the $L1_1$ structure [18]). Curtarolo predicted it as a ground state structure in Pd–Pt and Cd–Pt [19]; and it was discussed as a likely "missing structure" in the enumeration-related work of Hart [20].

The interest in $CuPt_3$ is not merely academic—it has practical import. Much research has been done on the catalytic properties of platinum and platinum-based alloys because of their widespread use in the chemical and petroleum industries. Additionally, the use of platinum-alloys in the jewelery industry accounts for a sizeable fraction of the worldwide consumption of platinum alloys, about 30% over the last decade [21]. In both cases, knowing the composition and structure of stable compounds is useful for materials improvement and design. It is surprising then that so little is known about the structural and mechanical properties of these alloys, knowledge that could be used to improve Pt-based jewelery alloys [22] and catalysts.

The most common alloying element in platinum jewelery is Cu in relativly low concentration. Although Pt–Cu has been extensively used by jewelerers for more than 100 years, the influence of the cubic 7:1 phase [1], which can dramatically harden the alloy when it is present even in small volume fractions [22], was only recently confirmed [23]. It is important to remember that even well-known alloys harbor surprises, and that novel alloy orderings can have significant impact on practical material performance.

The $L1_3$ structure was very recently predicted to be the stable phase of Cu–Pt at the 1:3 stoichiometry in Ref. [24]. It was this prediction that led to the present re-examination of the room-temperature structure of $CuPt_3$, where electron diffraction and X-ray powder diffraction measurements unambiguously identify the supercell and quantify the Pt and Cu occupancy fractions at each site, which turn out to be roughly consistent with $L1_3$. This reexamination clarifies the apparent discrepancies in previous work, connects first-principles predictions and experimental evidence in the Cu–Pt system, and provides a pathway towards the engineering of Cu–Pt alloys with superior properties.

2. Methods

Buttons of Cu 75 at.% Pt were prepared by arc-melting on a copper hearth. The thickness of a cast button was first reduced by 50% in a rolling mill, after which the alloy was homogenised in argon at 1000 °C for 24 h, terminated by quenching. The buttons were then reduced a further 90% by rolling. We checked composition (1) by carrying out SEM–EDS on the as-cast button; (2) by carrying out TEM–EDS on the TEM foils which were used for imaging and diffraction. In each case, composition was evaluated at a number of points and averaged; the average was within 1 at.% of 25 at.% Cu, 75 at.% Pt.

Disks for transmission electron microscopy (TEM) were cut from this cold rolled sheet and subjected to heat treatments between 100 °C and 800 °C in an argon atmosphere, terminated by quenching. Grinding and dimpling were followed by final thinning to perforation using a Gatan Precision Ion Polishing System.

The $CuPt_3$ sample used to collect synchrotron powder X-ray diffraction (PXD) data was prepared by 90% cold working and subsequent annealing at 350 °C for 2 months. Quantitative Rietveld analysis was performed using the TOPAS Academic (TA) software package.

Electron microscopy images and electron diffraction patterns were collected using a Tecnai F20 TEM, operating at 200 kV, by combining selected-area electron diffraction (SAED) and dark-field (DF) imaging. In situ heating was performed using a Gatan heating specimen stage in the TEM. For comparison with experimental results, the candidate structures and associated electron diffraction patterns were generated using CrystalMaker and SingleCrystal^{TM} software respectively.

Laboratory PXD data from a rolled foil of 1 mm thickness proved inadequate for Rietveld analysis due to the highly-oriented rolling texture and due to the weakness of the superlattice reflections. The samples were too small and expensive to grind into powders in amounts sufficient for flat-plate reflection-geometry experiments, but we did attempt a transmission-geometry experiment with a finely-ground powder that was lightly distributed over the surface of a Kapton capillary tube (the $CuPt_3$ absorption length is approximately 3 μm at an X-ray wavelength of 1.54 Å); because the sample density was very low, the relative scattering contribution from the Kapton introduced far too much background and noise to allow the investigation of weak superlattice peaks.

To overcome the challenges presented by small samples of strongly-absorbing and highly-oriented materials, we designed and built a double-axis sample spinner (DASS) (see Fig. 2(a)) in order to orientationally average a small polycrystal in transmission mode, and also utilized high-energy (30 keV, $\lambda = 0.41346$ Å) synchrotron X-rays at beamline 11BM at the Advanced Photon Source at Argonne National Laboratory. Because the absorption length at this energy is 20 μm, a small button sample was polished down into a foil of 20 μm thickness and trimmed into a 2.5 mm × 200 μm rectangle. In order to render the X-ray absorption as isotropic as possible, the rectangular foil was then roughly shaped into a half-cylinder with its axis parallel to the 2.5 mm dimension, and mounted on the tip of a steel needle (see Fig. 2(b)). The omega axis of the DASS was oriented perpendicular to the X-ray beam and parallel to the lab floor and rotated at a speed of 1 Hz, while the phi-axis, which is affixed to the moving omega stage, was inclined 54.74 ° relative to the omega axis and rotated at a speed of 10 Hz. The 2θ multi-detector bank was scanned in 0.005 ° increments while collecting data for 4 s per step. The time per point was intentionally set to an integer multiple of the DASS omega-axis period; incommensurability in this ratio results in undesirable cyclic background variations due to incomplete orientational averaging at each point. Because the detector bank had

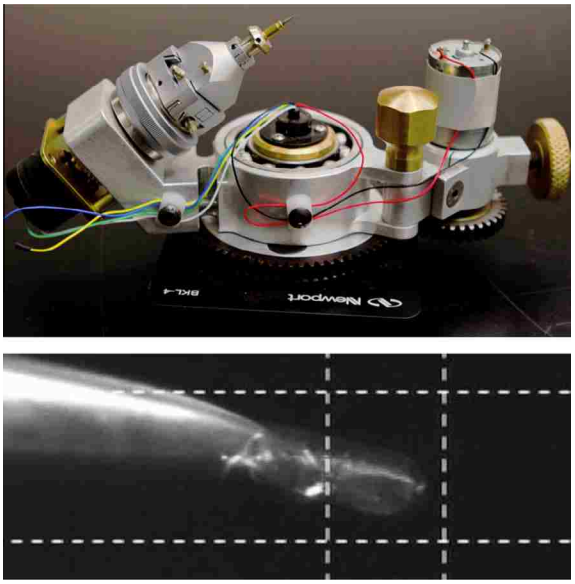


Fig. 2. (a) Gandolfi-type double-axis sample spinner for generating powder diffraction patterns from small polycrystals. The central omega axis (left) and inclined phi axis (right) are 54.74° apart for optimal orientational averaging and have independent motors that also act as counterweights. (b) Platinum alloy sample (mostly inside the center box) mounted to a steel needle tip (left side) at beamline 11BM. The 300 μm X-ray beam is smaller than the region marked by the dashed lines. The sample spins around two axes while maintaining a fixed point at the center of the beam.

12 detectors spaced 2 ° apart, a 2 ° scan covered 24 °, and two such scans provided coverage out to 48 °. Each scan was collected 6 times and averaged, requiring about 6 h of collection time.

We used the UNCLE [25] software package to perform a classical, thermodynamic Monte Carlo simulation on a supercell that repeated the primitive unit cell 32 times in each direction. The simulation used 9×10^5 flips per averaging step, with each step terminating after energy convergence within 0.5 meV. Order parameters for the supercell at each temperature step were calculated using the mean of the charge occupancies (based on atomic number) at each site in a given crystallographic direction.

3. Results and discussion

3.1. Transmission electron microscopy

The electron diffraction patterns from initially cold worked Cu 75 at.% Pt in Fig. 3(a), as observed along the [100], [110], [112] and [103] zone axes, show the fundamental reflections expected from a disordered fcc alloy. After heat treatment at 350 °C, additional reflections were observed (see Fig. 3(b)) halfway (1) along the {200} and {220} type directions in the [100] zone axis diffraction pattern, (2) along {200}, {111} and {220} in the $[110]_{fcc}$ pattern, (3) along {220} {111} and {131} in the $[112]_{fcc}$ pattern, and (4) along {131} and {200} in the $[103]_{fcc}$ pattern. These reflections were observed following each heat

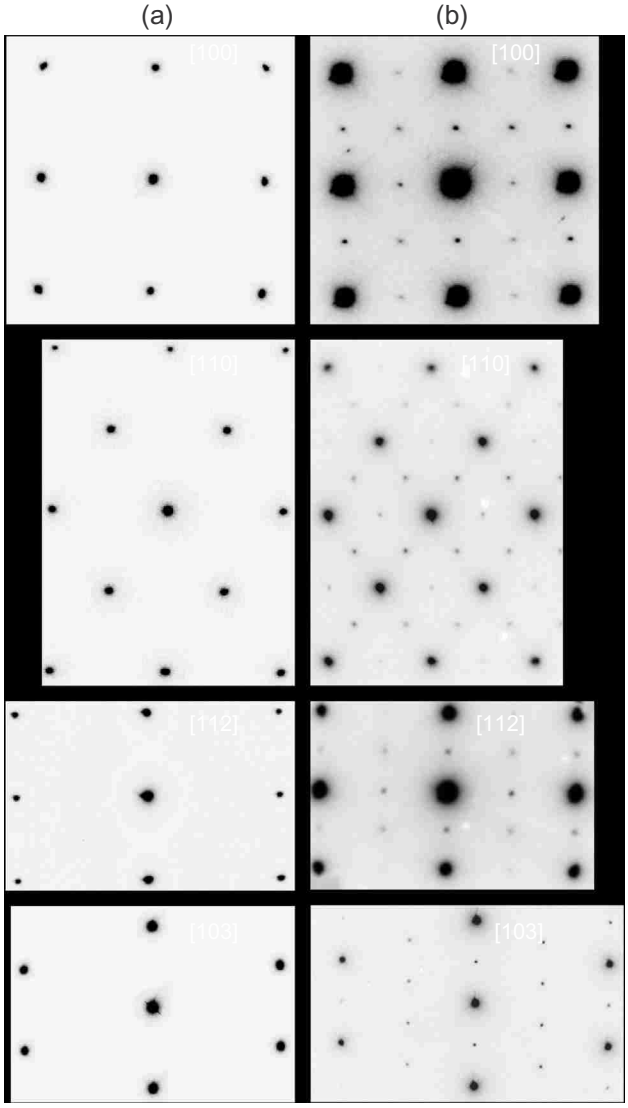


Fig. 3. Electron diffraction patterns from Cu 75 at.% Pt (a) initially cold worked; (b) after heat treatment at 350 °C for 3 h; showing $[100]_{fcc}$, $[110]_{fcc}$, $[112]_{fcc}$, and $[103]_{fcc}$ zone axes. The heat treated specimen shows clear signs of ordering.

treatment between 200 °C and 400 °C, indicating that ordering had taken place. Note that the $\frac{1}{2}\{220\}$ and $\frac{1}{2}\{131\}$ reflections are related by translations of the parent-fcc reciprocal lattice to the $\frac{1}{2}\{200\}$ and $\frac{1}{2}\{111\}$ reflections in the first Brillouin zone. In fact, all of the observed superlattice reflections are related to either $\frac{1}{2}\{200\}$ or $\frac{1}{2}\{111\}$ by translations of the parent-fcc reciprocal lattice. Thus, it is not possible for the intensities at these points to arise from the double-diffraction of two parent-lattice reflections.

The $L1_3$ structure has several inequivalent viewing directions or variants that contain the same fundamental reflections but different superlattice reflections. Fig. 4 shows three $L1_3$ variants with the same $[100]_{fcc}$ zone axis. We find that the electron diffraction patterns shown in Fig. 3(b) are consistent with those expected from the $L1_3$ structure based on simulations. However, we also find that they are

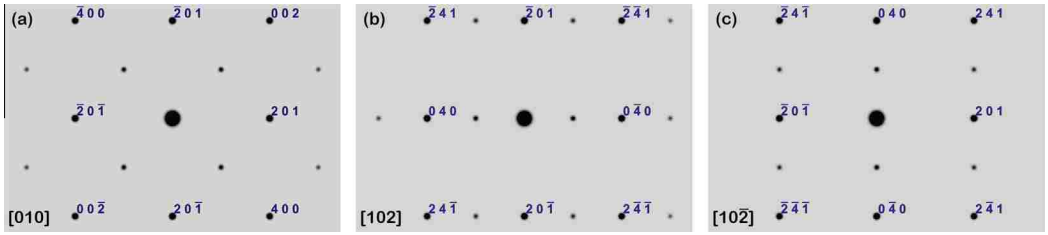


Fig. 4. Simulated electron diffraction patterns for three variant views of the $[100]_{fcc}$ zone axis diffraction pattern of the $L1_3$ structure of $CuPt_3$. Only the fundamental reflections of the fcc parent structure are labeled, but using the setting of the ordered structure.

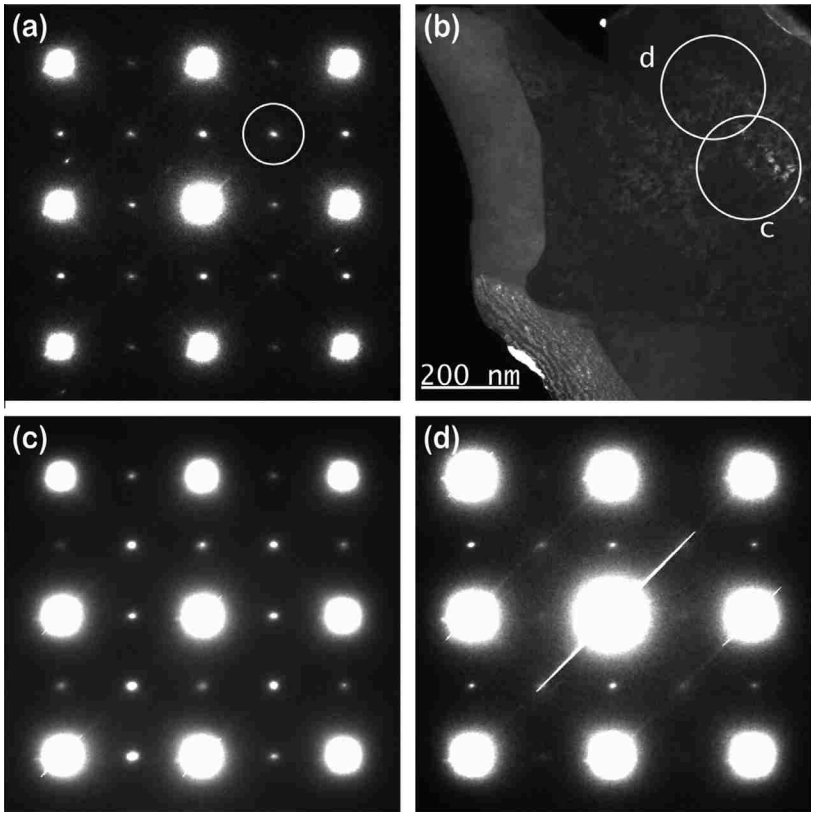


Fig. 5. Dark field images and selected area diffraction at different locations of the dark field image from $CuPt_3$ after heat treatment for 3 h at 350 °C: (a) shows a $[100]_{fcc}$ zone axis electron diffraction pattern from the whole grain; (b) is the dark field image made using the circled reflection in (a), with circled areas showing the regions from which selected area diffraction patterns (c) and (d) were acquired.

consistent with the 32-atom orthorhombic $CuPt_3$ structure of Schneider and Esch [1]. On closer inspection of Fig. 1(a and c), we discovered that the two orderings are crystallographically equivalent, a fact that does not appear to have been reported or discussed in previous literature. Thus, any subsequent mention of the *orthorhombic* ordering will implicitly refer to $L1_3$.

Each simulated pattern in Fig. 4 corresponds to the same experimental $[100]_{fcc}$ zone-axis pattern from Fig. 3(b), but lacks some of the experimentally-observed superlattice reflections. This is because the compositional ordering has a larger unit cell than the fcc parent structure, and therefore has distinct orientational variants. Each simulated pattern simulates only one of these variants. Because there is no orientation preference for nucleation of the $L1_3$ structure in the disordered alloy, the variant that appears within a given nucleating grain will be essentially random. Thus, any large-area diffraction pattern, such as those seen in Fig. 3(b), will sample all possible variants and will simultaneously exhibit all of their superlattice reflections. One critical feature of the work by Miida and Watanabe [8] was creating large enough ordered grains that distinctions between the variants could be observed. The presence of variants makes it possible to distinguish $L1_3$ from the $2 \times 2 \times 2$ cubic supercell, for which the $[100]_{fcc}$ zone axis diffraction pattern simultaneously contains the superlattice reflections from all three of the variants of Fig. 4.

Because the patterns in Fig. 3(b) are the result of averaging over multiple orientational variants of the ordered $L1_3$ structure, we collected dark-field images with a single

$\frac{1}{2}\{220\}$ type superlattice reflection within the $[100]_{fcc}$ pattern (Fig. 5(a)) to separate the distinct variants that give rise to $\frac{1}{2}\{200\}$ and $\frac{1}{2}\{220\}$ type superlattice reflections. The expected diffraction patterns of these variants are simulated in Fig. 4. Fig. 5(a) shows the diffraction from a large $[100]_{fcc}$ grain with the $\frac{1}{2}(220)$ used for the dark-field image circled with Fig. 5(b) showing the dark-field image with the central region being the $[100]_{fcc}$ grain. Sharp boundaries between intensities delineate different fcc grains and the mottled appearance within the $[100]_{fcc}$ grain is fluctuations of the $\frac{1}{2}\{220\}$ intensities. The two circles in Fig. 5(b) show the placement of a small SA aperture and indicate the regions from which diffraction patterns in 5(c) and 5(d) are taken. Because the ordered domains were roughly 10 nm in diameter, the size limitations of the microscope SA aperture made it difficult to fully isolate a single reflection type. However, regions c and d show a different mix of intensity in the additional superlattice refections. Region c was chosen to contain both the brightest and darkest regions of the dark-field image and shows approximately equal intensities in the possible superlattice peaks while region d, which was chosen for its intermediate intensity, shows a strong set of $\frac{1}{2}\{200\}$ peaks above and below the central peak and a much weaker set of $\frac{1}{2}\{200\}$ peaks to the left and right of the central peak; the $\frac{1}{2}\{220\}$ peaks are also weak. This mix of intensities demonstrates that the variants required of the $L1_3$ structure are indeed present in our sample.

Following a heat treatment at or above roughly 400 °C, the pattern of superlattice reflections changes to that indicated in Fig. 6, where the $\frac{1}{2}\{200\}$ and related reflections are unexpectedly weak relative to the $\frac{1}{2}\{111\}$ and related reflections. In the 350 °C-annealed samples of Fig. 3, however, these two types of reflections have more similar intensities.

In Figs. 3, 5 and 6, some strong Bragg reflections exhibit sharp streaks along $\{110\}$ directions. In most cases, (e.g., Fig. 5(d)), this is clearly a CCD-saturation artifact. Some of the weaker streaks are due to sweeping the diffraction pattern onto the CCD detector during a relatively short exposure. Because of the intense peaks in the diffraction pattern and the short exposure times, the sweep across the CCD leaves artifacts that can be seen in this case. We also note the presence of an unaccounted pair of reflections in the [100] panels of Fig. 3(b) and 5(a), which appear to be $\{200\}$ reflections from a second grain.

3.2. X-ray diffraction

The synchrotron PXD data from samples annealed for two months at 350° were of exceptionally high quality and permitted the observation of many superlattice peaks, all of which could be indexed using the supercell associated with the $L1_3$ ordered structure (Fig. 1(c)): a *C*-centered conventional cell with basis vectors $(1, 0, 1), (0, 2, 0), (-\frac{1}{2}, 0, \frac{1}{2})$, and origin $(0, 0, 0)$ relative to the conventional basis vectors of the fcc parent. In fact, the $L1_3$ ordering is the only binary decoration of an fcc lattice consistent with this supercell. Thus, we used the $L1_3$ model as a starting point for quantitative Rietveld analysis. The final fit is shown in Fig. 7.

Because the background was somewhat structured, we fit it with a combination of a $1/x$ term for air scattering, a Chebychev polynomial and several extremely-broad

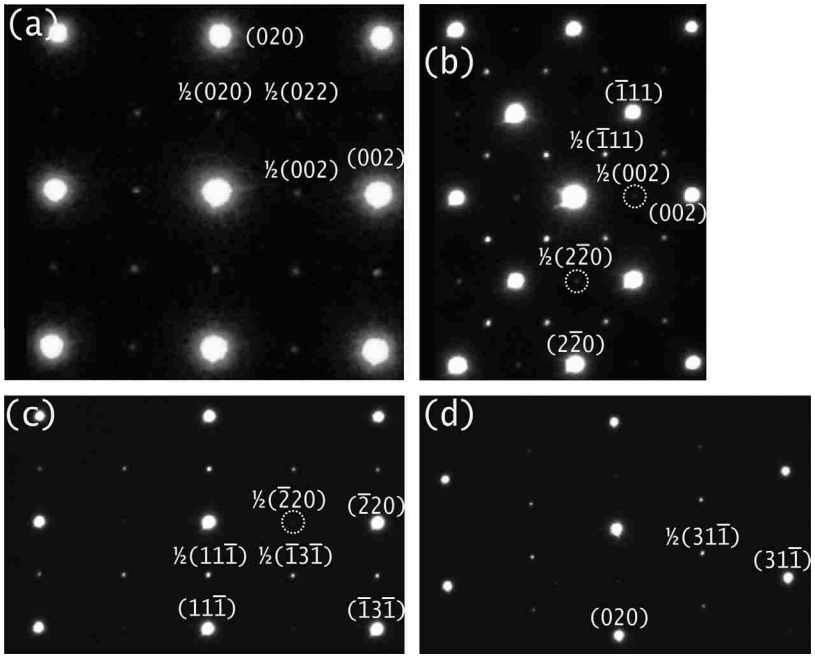


Fig. 6. Electron diffraction patterns from $CuPt_3$ after heat treatment for 3 h at 600 °C, showing (a) $[100]_{fcc}$, (b) $[110]_{fcc}$, (c) $[112]_{fcc}$, and (d) $[103]_{fcc}$ zone axes.

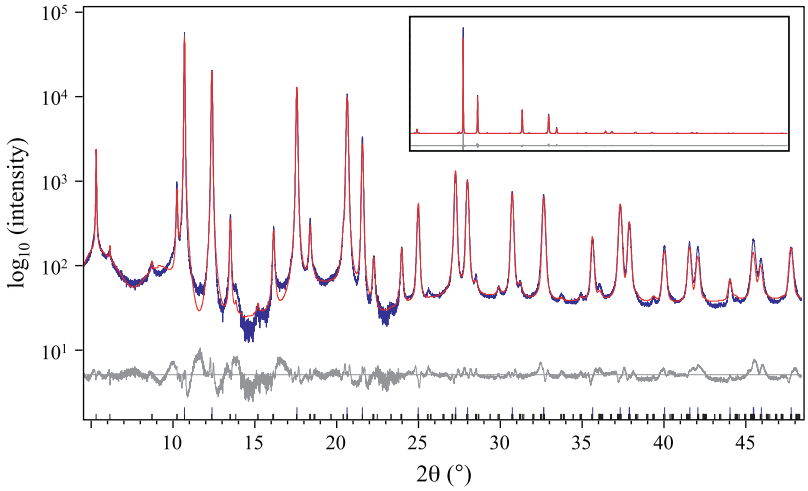


Fig. 7. Rietveld fit of a partially-ordered $L1_3$ model against synchrotron X-ray powder diffraction data ($\lambda = 0.4135$ Å) from $CuPt_3$ after heat treatment for 2 months at 350 °C. Blue, red, and gray curves indicate the experimental data, the model calculation, and the difference, respectively. The short vertical black lines at the bottom indicate supercell peak positions, while the longer vertical blue rods indicate parent fcc peak positions. Intensities are presented on a log scale because all but a few of the superlattice reflections are otherwise too weak to see. Apparently large discrepancies on the weaker peaks are actually very small. The inset contains a view of the same fit on a linear vertical scale. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

low-amplitude peaks. The peak shape was entirely strain dominated with widths that adhered to a distinct FWHM $= s\tan(\theta)$ trend. Furthermore, the superlattice peaks were distinctly broader than the fcc parent peaks, and were thus fitted with different shapes. The parent peaks exhibited a convolution of Lorentzian ($s = 0.442(2)^\circ$) and Gaussian ($s = 0.185(1)^\circ$) contributions, while the superlattice peaks were fitted with a purely Lorentzian ($s = 0.83(2)^\circ$) shape.

Despite an obviously correct supercell and a reasonably well-matched peak shape, the intensities were not well matched by a simple $L1_3$ ordering. The $L1_3$ structure has *Cmmm* space-group symmetry and three distinct crystallographic sites (see Fig. 1(c)): #1 the two atoms at the cell corner and the center of the *c*-face, #2 the two atoms at the middle of the *a* and *b* cell edges, and #3 the four atoms within the interior of the supercell. Allowing these three site occupancies to vary, under the constraints that the total Pt + Cu occupancy at each site remains equal to 1 and the overall stoichiometry remains at Cu:Pt $= 1{:}3$, proved to be important; it lowered the R_{wp} residual factor from 55.4% to 28.5%. The result was a complete Pt order at site #2, but partial or complete disorder at the other two sites[4].

Switching from isotropic (u_{iso}) to anisotropic (u_{ij}) thermal parameters further improved the fit, though the resulting anisotropy was so strong as to suggest the presence of site disorder. For this reason, we allowed each atom to fragment into a set of symmetry-related partial-occupancy pieces surrounding its ideal high-symmetry position. Combined, with the anisotropic thermal parameters, this sophistication visibly improved the fit. Note that the site disorder only tended to displace atoms in the x and y direction of the supercell, leading us to fix the z-axis displacements to zero. Furthermore, strong correlations between the off-site displacements and the anisotropic thermal parameters required us to fix the u_{22} and u_{33} parameters of each site. Not only were the microscopic parameters of the Pt and Cu atoms sharing a common site tied together, but the same Δx, Δy, and u_{11} were shared by all three sites (u_{12} only applies to site #3). It seems reasonable to conclude that the compositional disorder on sites #1 and #3, coupled with the different relative atomic radii of Pt and Cu, results in local size-effect displacements that are manifested as displacive disorder and anisotropic thermal parameters. Including these effects lowered R_{wp} to 17.7%.

The peak-height discrepancies that remained showed a clearly sinusoidal trend that completed more than one full period across the diffraction pattern, and were not resolvable using any combination of parameters available to the atoms of the supercell. Speculating that this oscillation is further evidence of large size-effect displacements that cannot be accommodated by a simple atomistic average structure, we elected to accommodate this trend by multiplying all peak intensities by an empirical sinusoidal envelope of the form $1 + A\cos^2(B\theta - C)$, where $A = 0.62, B = 0.36$, and $C = 6.5$ were fitting parameters. While such a term is unorthodox, it produced a clean overall fit with $R_{wp} = 10.9\%$ that improved our confidence in the other structural parameters.

The values of all other refined structural parameters described above appear in Table I, where statistical error estimates appear in parentheses. The compositional disorder on sites #1 and #3 clearly leads to substantial size-effect-induced displacive disorder. It is interesting that

[4] Because the *Cmmm* symmetry provides no displacive degrees of freedom to the $L1_3$ structure, we also tried lowering the symmetry to *P*1 within the primitive supercell, and used simulated annealing to optimize small displacements of 8 atoms in the supercell; but this did not reliably improve the fit. So we returned to the *Cmmm* symmetry in all subsequent attempts.

Table I
Tabulated results from the Rietveld refinement of the orthorhombic $L1_3$ model against synchrotron PXD data from $CuPt_3$, where $a = 7.6763(1), b = 5.4405(2), c = 2.7204(1), \Delta y = 0.0428(1), \Delta z = 0.0394(3), u_{11} = 0.0131(2), u_{12} = 0.0108(5)$, and $R_{wp} = 10.9\%$.

Atom	x	y	z	occ	u_{11}	u_{22}	u_{33}	u_{12}	u_{13}	u_{23}
Pt1	0	$0+\Delta y$	$0+\Delta z$	0.497(4)	u_{11}	0*	0*	0	0	0
Cu1	0	$0+\Delta y$	$0+\Delta z$	0.503(4)	u_{11}	0*	0*	0	0	0
Pt2	0	$1/2+\Delta y$	$0+\Delta z$	1.073(4)	u_{11}	0*	0*	0	0	0
Cu2	0	$1/2+\Delta y$	$0+\Delta z$	0.073(4)	u_{11}	0*	0*	0	0	0
Pt3	$\frac{1}{4}$	$\frac{1}{4}+\Delta y$	$1/2+\Delta z$	0.712(3)	u_{11}	0*	0*	u_{12}	0	0
Cu3	$\frac{1}{4}$	$\frac{1}{4}+\Delta y$	$1/2+\Delta z$	0.288(3)	u_{11}	0*	0*	u_{12}	0	0

the atomic displacements have magnitudes on the order of 0.1 to 0.25 Å, which are comparable to the difference between the nearest-neighbor distances in fcc Pt (2.775 Å) and Cu (2.556 Å) at room temperature.

3.3. Interpretation

Assuming that the total occupancy (Pt + Cu) of each symmetry-unique site is constrained to equal 1, and that the overall stoichiometry is fixed during the composition-ordering process, a complete description of the $L1_3$ ordering (space group *Cmmm*) has two independent variables. Using group representational analysis, we associated these variables with irreducible matrix representations (IRs) of the parent $Fm\bar{3}m$ space group. Using the ISODISTORT software package [26], we find that the first variable is an $(a, a, 0, 0)$ order parameter of the L_1^+ IR at the reciprocal-space $L(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ point, while the second variable is an $(a, 0, 0)$ order parameter of the X_1^+ IR at the reciprocal-space $X(1, 0, 0)$ point. Of the two contributing order parameters, $L_1^+(a, a, 0, 0)$ is primary in the sense that its action alone is sufficient to produce the observed supercell and space-group symmetry. $X_1^+(a, 0, 0)$ is secondary because it is consistent with the symmetry of the primary order parameter, but cannot achieve such a low symmetry by itself. $X_1^+(a, 0, 0)$, when acting alone, would result in the $L1_0$ structure, which is the most common ordering among all 1:1 binary alloys. Each superlattice reflection that results from the ordering differs from one of these two points by a parent lattice vector, and is therefore associated with the corresponding order parameter. The relative intensities of the L and X-type superlattice reflections then gauge the relative contributions of their respective order parameters. Together, these two order parameters provide a natural symmetry-based description of deviations from the fcc structure.

Static concentration waves at the L and X points of $Fm\bar{3}m$ have been used previously to describe ordering in copper platinum alloys [9,27,28], though they were not labeled in this way. A proper group-representational description of the concentration waves is completely consistent with their observations, but reveals fundamental problems with their terminology, wherein they incorrectly associated all L-point waves with the label $L1_1$ and all X-point waves with the label $L1_2$.

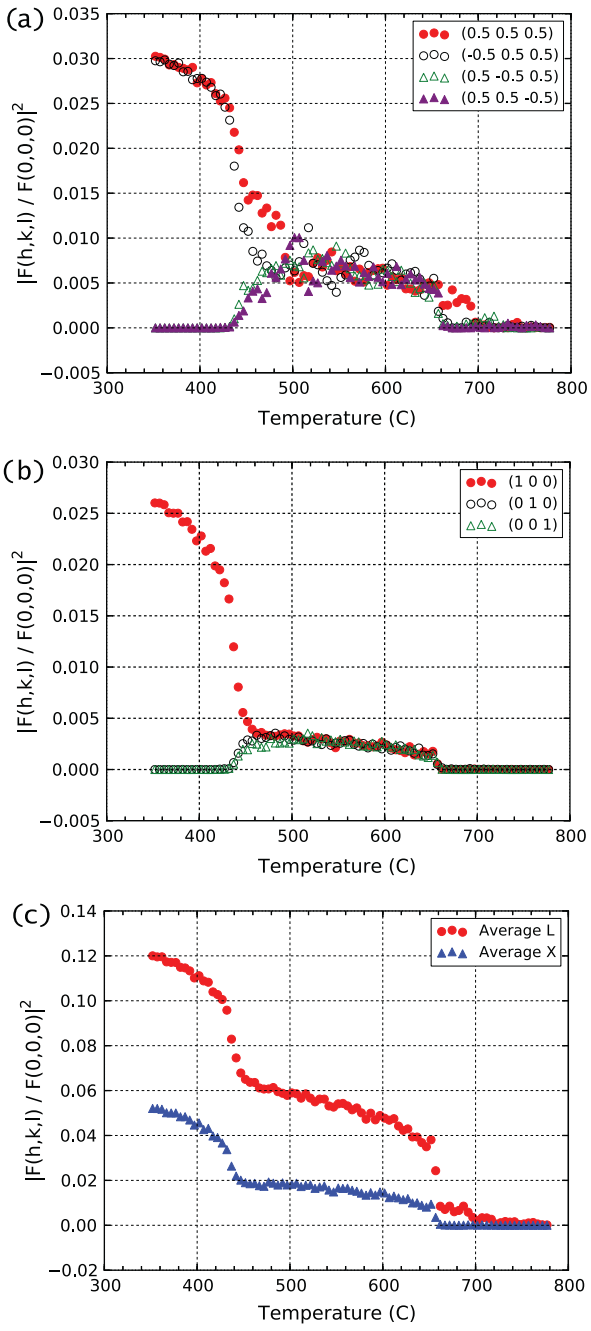


Fig. 8. Simulated kinematic superlattice intensities associated with the X and L-points derived from Monte-Carlo simulations: (a) the four L-point reflections, (b) the three X-point reflections, (c) powder averages of the X and L-point intensities taking reflection multiplicity into account.

The star of the L point includes four vectors: $(\frac{1}{2},\frac{1}{2},\frac{1}{2})$, $(\frac{1}{2},-\frac{1}{2},-\frac{1}{2})$, $(-\frac{1}{2},\frac{1}{2},-\frac{1}{2})$, and $(-\frac{1}{2},-\frac{1}{2},\frac{1}{2})$; and the star of the X point includes three vectors: $(1, 0, 0), (0, 1, 0)$, and $(0, 0, 1)$. $L_1^+(a, 0, 0, 0)$ is primary for the $L1_1$ ordering stable at 50% Pt, and uses only one L-point vector. $L_1^+(a, a, 0, 0)$ is primary for the $L1_3$ ordering near 75% Pt, and uses two L-point vectors. $L_1^+(a, a, a, a)$ is primary for the large-cubic ordering just above 75% Pt, and uses all four L-point vectors. $L_1^+(a, b, b, b)$ is primary for the large-rhombohedral ordering just below 75% Pt, and uses all four L-point vectors; it can be viewed as a superposition of $L1_1$ and the large-cubic ordering. $X_1^+(a, a, a)$ is primary for $L1_2$ order, which is reported to be stable at 25% Pt (Ref. [8]); it uses all three X-point vectors; it is secondary to both the large-cubic and large-rhombohedral orderings. $X_1^+(a, 0, 0)$, which uses only one X-point vector, is primary for $L1_0$ order but not observed at 50% Pt; it is also secondary for $L1_3$ ordering near 75% Pt. This more complete analysis reveals that the Cu–Pt phase diagram involves many different structure types, and cannot be summarized merely in terms of the labels $L1_1$ and $L1_2$, as was done previously [27,28].

The temperature-dependent Monte Carlo simulations of Fig. 8 indicate at least two phase transitions upon cooling from the high-temperature disordered state, though we are not able to infer accurate phase-transition temperatures from this output. The first panel tracks the relative intensities of the four $\frac{1}{2}[1\,1\,1]$ reciprocal-space reflections in the kinematic approximation, which correspond to the four components of the $L_1^+(a, b, c, d)$ order parameter. The second panel tracks the relative intensities of the three $\frac{1}{2}[2\,0\,0]$ reflections, which correspond to the three components of the $X_1^+(a, b, c)$ order parameter. The third panel tracks the powder-average intensities associated with the L and X reflections. Upon cooling, near 660 °C, all four L components take on comparable values and all three X component values take on comparable values, indicating the formation of the large 2×2×2 cubic supercell of Tang [2]. Then in the vicinity of 480 °C, two of the L components begin to rise towards a new maximum while the other two drop to zero, and one of the X components begins to rise towards a new maximum, while the other two drop to zero; this indicates a transition to the $L1_3$ structure.

This computational result is consistent with the basic features of the phase diagram of Miida and Watanabe [8]. However, a closer inspection of the simulation output reveals additional subtleties; one of the L components clearly activates at a high temperature than the other L or X components indicating the narrowly limited presence of an $L1_1$ transitional phase between 660 °C and 690 °C. Furthermore, between 480 °C and 520 °C, one of the L components is significantly larger than the other three (though the various components appear to take turns being the large one), indicating the presence of a transitional phase with the large-rhombohedral supercell. Finally, between 430 °C and 480 °C, the two large L components don't immediately acquire the same intensity, indicating a further lowering of the point symmetry of the transitional phase to monoclinic. Our experimental mesh of temperatures was much too coarse for the detection of such subtle phase variations, though an attempt to observe them might be worthwhile.

The PXD Rietveld analysis is only sensitive to the average L and X-component intensities due to overlap of equivalent reflections. For the refinement of the disordered $L1_3$ phase, rather than refining each atomic occupancy independently, we refined the order parameters directly, which are related to the Cu occupancies as follows.

$$\mathrm{occ(Cu1)} = \frac{1}{4} + \frac{1}{2}S_L + \frac{1}{4}S_X$$
$$\mathrm{occ(Cu2)} = \frac{1}{4} - \frac{1}{2}S_L + \frac{1}{4}S_X$$
$$\mathrm{occ(Cu3)} = \frac{1}{4} - \frac{1}{4}S_X$$

Here, S_L is a factor of $\sqrt{2}$ larger than the normalized $L_1^+(a, a, 0, 0)$ order parameter used by ISODISTORT, and S_X is $\sqrt{2}$ times larger than the normalized $X_1^+(a, 0, 0)$ order parameter used by ISODISTORT. The unnormalized parameters conveniently run from 0 in the case of complete disorder to 1 in the case of long-range order, and are similar to those used in the static concentration-wave theory of Khachaturyan [9], who used the symbol η rather than S. Because all of the atomic occupancies must lie between 0 and 1, we simply require that $S_L \leqslant \frac{1}{2} + \frac{1}{2}S_X$.

From the Rietveld analysis of the sample annealed for two months at 350°, the fitted values in Table I show that $S_L = 0.430$ has a large value, whereas $S_X = 0.051$ is quite small in comparison, resulting in a partially-ordered version of $L1_3$, as illustrated in Fig. 1(d). For fully ordered $L1_3$, we would instead have $S_L = S_X = 1$. The role of the $L_1^+(a, a, 0, 0)$ is to shift Cu from site #2 to site #1, which can only proceed alone until site #2 is completely empty of Cu. If $S_X = 0$ for $CuPt_3$, the limiting value of S_L is $\frac{1}{2} + \frac{1}{2}\cdot 0 = 0.5$, which enriches site #1 to the level of 50% Cu without affecting the composition of site #3. $X_1^+(a, 0, 0)$ should complete the $L1_3$ structure by transferring the residual 25% Cu on site #3 to site #1; despite its failure to do so, the overall Cu fraction remains at $\frac{1}{8}\left(2\cdot\frac{1}{2} + 2\cdot 0 + 4\cdot\frac{1}{4}\right) = \frac{1}{4}$, as expected, where site multiplicities have been taken into account. The fitted values of $S_X \approx 0$ and $S_L \approx 0.5$ indicate that $L_1^+(a, a, 0, 0)$ is contributing close to the maximum amount possible given the small value of $X_1^+(a, 0, 0)$.

Unexpectedly small values for the X-point order parameters were also evident in electron diffraction data from samples annealed at temperatures higher than 350 °C, including those in the range where the large-cubic ordering is expected, e.g. Fig. 6. The Monte Carlo simulations of Fig. 8 don't support a thermodynamically stable phase with such a small X/L intensity ratio, even in the transitional region between the $L1_3$ and large-cubic phases. The expected ratios are considerably less than one, but the observed intensity ratios are at least four times smaller

than expected. And even if such an ordering were stable, it would be very unusual for a secondary order parameter to become active at a substantially lower temperature than its primary; the opposite is normally true, as seen, for example, in the $CuMnPt_6$ system [28,29]. For this reason, we suspect that the small X/L order-parameter ratio and its associated compositional disorder are due to kinetic limitations. We find that the energy of the $L1_3$ structure is far more sensitive to $L_1^+(a,a,0,0)$ order parameter than it is to $X_1^+(a,0,0)$, which provides one possible explanation for the relative kinetic difficulty of forming the X order parameter. But we don't understand why such limitations would be more restrictive for the samples that were annealed at higher temperatures. It could also be that slight cold-working and polishing during sample preparation (in the case of the sample used for Rietveld analysis) disrupted the X order parameter.

4. Summary

The ordered $CuPt_3$ alloy was first proposed and presented as a 32-atom orthorhombic superstructure [1] based on X-ray diffraction data, but subsequent experimental identification of two other distinct phases at about this stoichiometry have confused the issue. Work by Miida [8], and also our cluster-expansion-based Monte Carlo simulations, have shown the 32-atom orthorhombic ordering is stable for room-temperature $CuPt_3$, but that these other phases are present in nearby regions of the phase diagram.

To date, no ordered alloy has been experimentally associated with the hypothetical $L1_3$ structure employed in the computational arena. It is of particular interest that, prior to the present work, the $L1_3$ structure has not been experimentally associated with the $CuPt_3$ alloy. Instead, the 32-atom orthorhombic structure is often referenced as the stable structure for this stoichiometry. We observe that the two structures have identical simulated diffraction patterns, and that the atomic arrangements are in fact crystallographically equivalent. The fact that years of work in the computational arena failed to make this connection strongly suggests that the primitive unit cells of other alloy systems should be reevaluated. Identifying these two structures as one and the same should help to clear up past confusion surrounding the Pt-rich side of the Cu–Pt phase diagram.

TEM images and electron diffraction patterns show that our cold-worked samples of $CuPt_3$ became compositionally ordered after annealing at relatively low temperatures. Bright and dark-field electron diffraction patterns from multiple orientational variants confirmed the primitive unit cell to be that of $L1_3$ at 350 °C. Monte Carlo simulations show that $CuPt_3$ first forms a large-cubic supercell upon cooling from the disordered state, and subsequently forms the $L1_3$ phase, in basic agreement with the phase diagram of Miida and Watanabe [8]. These simulations also indicate the stability of $L1_1$ and large-rhombohedral transitional phases within narrow temperature ranges.

An innovative data collection strategy yielded high-quality synchrotron powder-diffraction data from small poly-crystalline foil fragments that suffer from extreme preferred orientation. We are not aware of comparable experiments involving metallic alloys. Using a sample annealed at 350 °C for two months, X-ray Rietveld analysis confirmed that essentially 100% of the sample material had ordered, though the ordering itself was not incomplete due to an unexpectedly low value of the $X_1^+(a,0,0)$ over $L_1^+(a,a,0,0)$ ratio, which we judge to result from non-thermodynamic considerations.

Acknowledgments

Financial support for CM, RRV, and CIL from the National Research Foundation (South Africa) is gratefully acknowledged. GLWH, LRR, BJC, CWR, and KCE are grateful for support from the National Science Foundation, DMR-0908753. RRV is grateful for support from the National Science Foundation, DMR-0906385. Use of the Advanced Photon Source, an Office of Science User Facility operated for the U.S. Department of Energy (DOE) Office of Science by Argonne National Laboratory, was supported by the U.S. DOE under Contract No. DE-AC02–06CH11357. A discussion with Roman Chepulskii regarding the concentration wave formalism is appreciated. GLWH and CWR are grateful for use of the Fulton Supercomputer Lab at Brigham Young University.

References

[1] Schneider A, Esch U. Z Elektrochem 1944;50.
[2] Tang Y-C. Acta Crystallogr 1951;4:377. ISSN 0365-110X.
[3] Wu N-C, Iwasaki H, Ogawa S. Trans Jpn Inst Metals 1973;14:309.
[4] Hansen M, Anderko K. Constitution of binary alloys. Metallurgy and metallurgical engineering series, vol. 1. McGraw-Hill; 1965.
[5] Hirone T, Adachi K. Sci Rep RITU 1955;A-7:282.
[6] Nagaski S, Hirone T, Kono H. Phys Soc Jpn; 1955 [unpublished].
[7] Ogawa S, Iwasaki H, Terada A. J Phys Soc Jpn 1973;34:384.
[8] Miida R, Watanabe D. J Appl Crystall 1974;7:50.
[9] Khachaturyan AG. Progress Mater Sci 1978;22:1. ISSN 0079-6425.
[10] Ewald VPP, Hermann C. Strukturbericht 1913–1928 (Akademische Verlagsgesellschaft M.B.H., 1931).
[11] Kanamori J, Kakehashi Y. J Phys 1977;38:274.
[12] Hart GLW. Phys Rev B 2009;80:014106.
[13] Fontaine DD. Cluster approach to order–disorder transformations in alloys, vol. 47. New York: Academic Press; 1994.
[14] Ferreira LG, Wei S-H, Zunger A. Int J Supercomput Appl 1991;5:34.
[15] Hart GLW, Forcade RW. Phys Rev B 2008;77:224115.
[16] Hart GLW, Forcade RW. Phys Rev B 2009;80:014120.
[17] Hart GLW, Nelson LJ, Forcade RW. Comp Mater Sci 2012;59:101.
[18] Müller S, Zunger A. Phys Rev Lett 2001;87:165502.
[19] Curtarolo S, Morgan D, Ceder G. Calphad 2005;29:163.
[20] Hart GLW. Nat Mater 2007;6:941.
[21] Jollie D. Tech Rep, Johnson Matthey; 2009.
[22] Carelse M, Lang CI. Scripta Mater 2006;54:1311. ISSN 1359-6462.
[23] Saha DK, Shishido T, Iwasaki H, Ohshima K. J Phys Soc Jpn 2003;72:1670.
[24] Nelson LJ, Hart GLW, Curtarolo S. Phys Rev B 2012;85:054203.
[25] Lerch D, Wieckhorst O, Hart GLW, Forcade RW, Müller S. Model Simul Mater Sci Eng 2009;17:055003.

[26] Campbell BJ, Stokes HT, Tanner DE, Hatch DM. J Appl Crystallogr 2006;39:607.
[27] Iwasaki H, Ohshima K. J Phys Soc Jpn 2005;74:2496.
[28] Takahashi M, Das AK, Sembiring T, Iwasaki H, Ohshima K-I. Physica B – Condensed Matter 385:2006;130. In: 8th International conference (ICNS 2005), Sydney, Australia, November 27–December 02; 2005.
[29] Takahashi M, Sembiring T, Yashima M, Shishido T, Ohshima K. J Phys Soc Jpn 2002;71:681. ISSN 0031-9015.

CHAPTER 5

Machine Learning Grain Boundaries

As described briefly above, grain boundaries (GBs) exert substantial influence on material properties that are of interest for engineering. Unfortunately, optimizing GBs cannot be done easily without first understanding the physics underlying their behavior. For example, in nuclear applications, GBs can act as sinks for vacancy defects and improve radiation resistance, but not all GBs are created equal and some GBs are more efficient sinks than others [67]. Similarly, in stainless steels used in a variety of applications, certain GBs resist stress corrosion cracking better than others [68].

Optimizing these materials leads to improved performance but not without an understanding of what needs to be optimized. Unfortunately, optimization has been difficult because of the wide range of macroscopic and microscopic variables influencing a GB. Macroscopically, GBs have 5 crystallographic degrees of freedom that cause their properties to vary. Microscopically, GBs have a high-dimensional phase space because of the positional degrees of freedom of atoms in a GB. Together, these microscopic and macroscopic degrees of freedom make the optimization space seem so large and complex as to be intractable.

Despite these difficulties, we were able to develop a representation for machine learning grain boundaries that goes beyond merely approximating these high-dimensional GB functions—it also provides insight into the physics that governs GB properties [69]. Without including important structures identified in the literature (such as stacking faults and edge dislocations), the machine learning was able to identify these structures as critical in property prediction. Perhaps more important, however, is that several new atomic environments were discovered that are highly correlated with material properties, despite the fact that these structures have not been identified in the literature as being important. This approach provides a general method to identify the physics that controls GB structure-property relationships.

Because of the general applicability of the method, several exciting projects have spawned that continue to use these pioneering ideas. For example, we are part of a collaboration to identify the physics behind solute deposition in grain boundaries [70] and are working with a different group to discover sites within grain boundaries for preferential hydrogen adsorption in iron [71]. Our project to automate alloy potential creation will open new avenues for creating grain boundary databases for multi-component systems as well. This methodology and the code package to apply it will be ready to use these databases and thereby expand our knowledge of the physics of grain boundary systems.

This publication evolved from my class project for a materials modeling course taught by Eric Homer. Because the initial results were already good, we collaborated to extract physical meaning from the machine learning models. This required many iterations and included many failed attempts. As discussed in the Author Contributions section of this article, I conceived the idea and performed all the calculations. Homer and Hart provided essential expertise in guiding the project and interpreting results.

The following article is reproduced with permission. A license is on file with the Department of Physics and Astronomy.

npj | Computational Materials

www.nature.com/npjcompumats

ARTICLE OPEN

Discovering the building blocks of atomic systems using machine learning: application to grain boundaries

Conrad W. Rosenbrock[1], Eric R. Homer[2], Gábor Csányi[3] and Gus L. W. Hart[1]

Machine learning has proven to be a valuable tool to approximate functions in high-dimensional spaces. Unfortunately, analysis of these models to extract the relevant physics is never as easy as applying machine learning to a large data set in the first place. Here we present a description of atomic systems that generates machine learning representations with a direct path to physical interpretation. As an example, we demonstrate its usefulness as a universal descriptor of grain boundary systems. Grain boundaries in crystalline materials are a quintessential example of a complex, high-dimensional system with broad impact on many physical properties including strength, ductility, corrosion resistance, crack resistance, and conductivity. In addition to modeling such properties, the method also provides insight into the physical "building blocks" that influence them. This opens the way to discover the underlying physics behind behaviors by understanding which building blocks map to particular properties. Once the structures are understood, they can then be optimized for desirable behaviors.

npj Computational Materials (2017)3:29 ; doi:10.1038/s41524-017-0027-x

INTRODUCTION

Although interactions between small, isolated atomic systems can be studied experimentally and then modeled, real-world systems are exponentially more complex because of multi-scale, many-body interactions between all the atoms. Approximate, statistical methods are then necessary in the quest for deeper understanding. Machine learning is a powerful statistical tool for extracting correlations from high-dimensional data sets; unfortunately, it often suffers from a lack of interpretability. Researchers can create models that approximate the physics well enough, but the physical intuition usually provided by models may be hidden within the complexity of the model (the black-box problem). Here, we present a general method for representing atomic systems for machine learning so that there is a clear path to physical interpretation, or the discovery of those "building blocks" that govern the properties of these systems.

We choose to demonstrate the method for crystalline interfaces because of their inherent complexity, high-dimensionality, and broad impact on many physical properties. Crystalline building blocks are well known and can be classified by a finite set of possible structures. Disordered atomic structures on the other hand are difficult to classify and there is no well-defined set of possible structures or building blocks. Furthermore, these disordered atomic structures often exhibit an oversized influence on material properties because they break the symmetry of the crystals. Crystalline interfaces, more commonly called grain boundaries (GBs), are excellent examples of disordered atomic structures that exert significant influence on a variety of material properties including strength, ductility, corrosion resistance, crack resistance, and conductivity.[1–9] They have macroscopic, crystallographic degrees of freedom that constrain the configuration between the two adjoining crystals.[10, 11] GBs also have microscopic degrees of freedom that define the atomic structure of the GB.[12–15] While often classified experimentally using the crystallography, the crystallography is only a constraint, and it is the atomic structure that controls the GB properties.

In this article, we examine the local atomic environments of GBs in an effort to discover their building blocks and influence on material properties. This is achieved by machine learning on the space of the atomic environments to make property predictions of GB energy, temperature-dependent mobility trends, and shear coupling. The implications of the work are significant; despite the immense number of degrees of freedom, it appears that GBs in face-centered cubic (FCC) nickel are constructed with a relatively small set of local atomic environments. This means that the space of possible GB structures is not only searchable, but that it is possible to find the atomic environments that give desired properties and behaviors. We emphasize that in addition to being successful for modeling GBs, the methodology presented here could be applied generally to many atomic systems.

Atomic structures in GBs have been examined for decades using a variety of structural metrics[12, 16–26] with the goal of obtaining structure-property relationships.[10, 11, 27–32] Each of the efforts has given unique insight, but none has provided a universally applicable method to find relationships between atomic structure and specific material properties.

Large databases of GB structures have produced property trends[12, 33–35] and macroscopic *crystallographic* structure-property relationships,[36, 37] but no *atomic* structure-property relationships. Machine learning of GBs by Kiyohara et al.[38] has been used to make predictions of GB energy from atomic structures, but we are still left without an understanding of what is important in making the predictions, and how that affects our understanding of the underlying physics and the building blocks that control properties and behaviors. We now present a method to address these limitations.

[1]Department of Physics and Astronomy, Brigham Young University, Provo, UT 84602, USA; [2]Department of Mechanical Engineering, Brigham Young University, Provo, UT 84602, USA and [3]Engineering Laboratory, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK
Correspondence: Conrad W. Rosenbrock (rosenbrockc@gmail.com) or Eric R. Homer (eric.homer@byu.edu)

Received: 5 April 2017 Revised: 23 May 2017 Accepted: 11 June 2017
Published online: 03 August 2017

METHODS

To examine atomic structures, we adopt a descriptor for single-species GBs based on the Smooth Overlap of Atomic Positions (SOAP) descriptor.[39, 40] The SOAP descriptor uses a combination of radial and spherical spectral bases, including spherical harmonics. It places Gaussian density distributions at the location of each atom, and forms the spherical power spectrum corresponding to the neighbor density. The descriptor can be expanded to any accuracy desired and goes smoothly to zero at a finite distance, so that it has compact support.

The SOAP descriptor has the following qualities that make it ideal for Local Atomic Environment (LAE) characterization. Specifically, within GBs, the SOAP descriptor (1) is agnostic to the grains' specific underlying lattices (including the loss of periodicity at the GB); (2) has invariance to global translation, global rotation, and permutations of identical atoms; (3) leads to a metric that is smooth and stable against deformations. SOAP vectors are part of a normed vector space so that similarity uses a simple dot product. This dot product can be used to produce a symmetric dissimilarity s, defined as

$$s = \left| \frac{\|\vec{a}\| + \left\|\vec{b}\right\|}{2} - \vec{a} \cdot \vec{b} \right|, \tag{1}$$

that is sensitive to the norm of each SOAP vector. Normally, SOAP similarity uses a dot product on normalized SOAP vectors; however, in our experience this reduces the discriminative ability of the representation.

In GBs, the SOAP descriptor has advantages over other structural metrics in that it requires no predefined set of structures, and a small change in atomic positions produces a correspondingly small (and smooth) change in the SOAP dissimilarity s (see Eq. 1).[17, 18, 20, 23, 24] Moreover, the SOAP vector is complete in the sense that any given LAE can be reconstructed from its SOAP descriptor.

Figure 1 illustrates the process for determining the SOAP descriptor for a GB. First, GB atoms and some surrounding bulk atoms are isolated from their surroundings; a SOAP descriptor for each atom in the set is calculated and represented as a vector of coefficients. The matrix of these vectors, one for each LAE, is the full SOAP representation for each GB. The SOAP vector can be expanded to resolve any desired features by increasing the number of terms in the basis expansion of the neighbor density at fixed cutoff. For the present work, a cutoff distance of 5 Å (≈1.4 lattice parameters) and vector of length 3250 elements produced good results; the selection of SOAP parameters is discussed in Section I of the Supplementary Information. The computed GBs studied in this work are the 388 Ni GBs created by Olmsted, Foiles, and Holm,[12] using the Foiles-Hoyt embedded atom method potential.[41]

We investigate two approaches for applying machine learning to the GB SOAP matrices. For the first option, we average the SOAP vectors, or coefficients, of all the atoms in a single GB to obtain one averaged SOAP vector that is a measure of the whole GB as shown in Fig. 2. In other words, it is a single description of the average LAE for the whole GB structure. We refer to this single averaged vector representation as the Averaged SOAP Representation (ASR). The ASR for a collection of GBs becomes the feature matrix for machine learning.

Alternatively, we can compile an exhaustive set of *unique* LAEs by comparing the environment of every atom in every GB to all other environments using the dissimilarity metric s (from Eq. (1)) and a numerical similarity parameter ε (see Fig. 2). Two LAEs are considered to belong to the same, unique class of LAEs if $s < \varepsilon$. A SOAP vector will produce a value $s = 0$ when compared with itself.

Using an n^2 search over all LAEs in all GBs produces the set U of unique LAE classes, each with a representative LAE, for the GB system. For a sufficiently small ε each GB will be characterized by a unique fingerprint in terms of the LAEs it contains. As ε gets smaller, the number of unique LAEs that characterize a GB increases exponentially. When an LAE is sufficiently dissimilar to all others in the set, it is added and becomes the representative LAE for the class of all other LAEs that are similar to it. Any of the LAEs in the class could be the representative LAE since they are all similar. As additional data becomes available, this set of U LAEs may increase in size if new LAE classes are discovered. Section III in the Supplementary Information presents additional details.

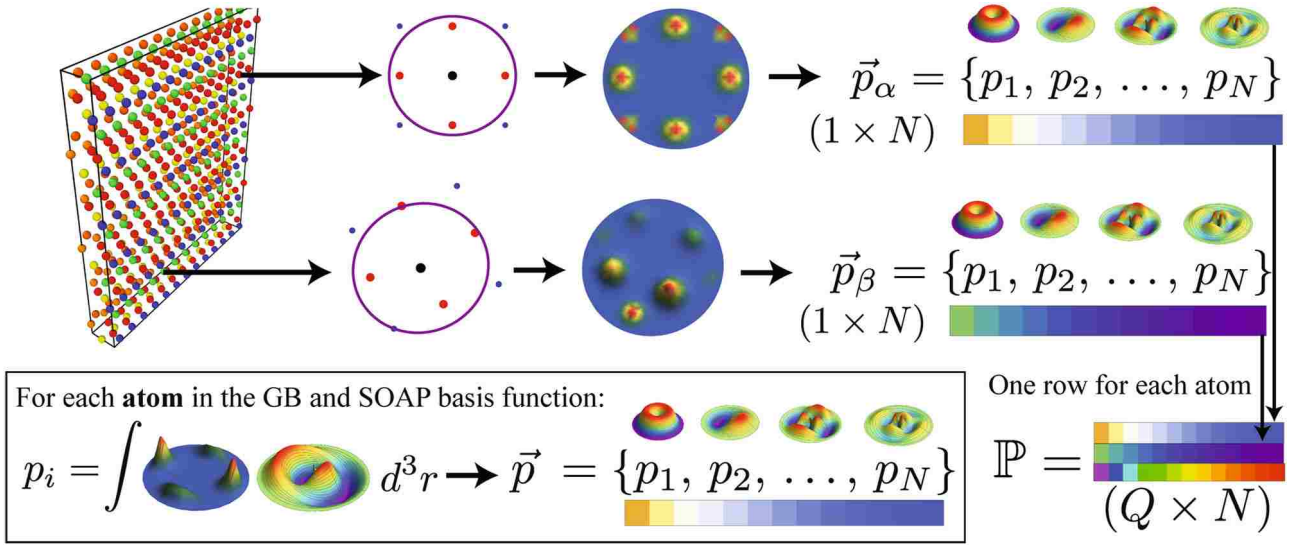


Fig. 1 Illustration of the process for extracting a SOAP matrix $\mathbb{P}$ for a single GB. Given a single atom in the GB, we place a Gaussian particle density function at the location of each atom within a local environment sphere around the atom. Next, the total density function produced by the neighbors is projected into a spectral basis consisting of radial basis functions and the spherical harmonics, as shown in the *boxed* region. Each basis function produces a single coefficient p_i in the SOAP vector $\vec{p}$ for the atom, the magnitude of which is represented in the figure by the colors of the arrays. Once a SOAP vector is available for all Q atoms in the GB, we collect them into a single matrix $\mathbb{P}$ that represents the GB. A value of $N = 3250$ components in $\vec{p}$ is representative for the present work

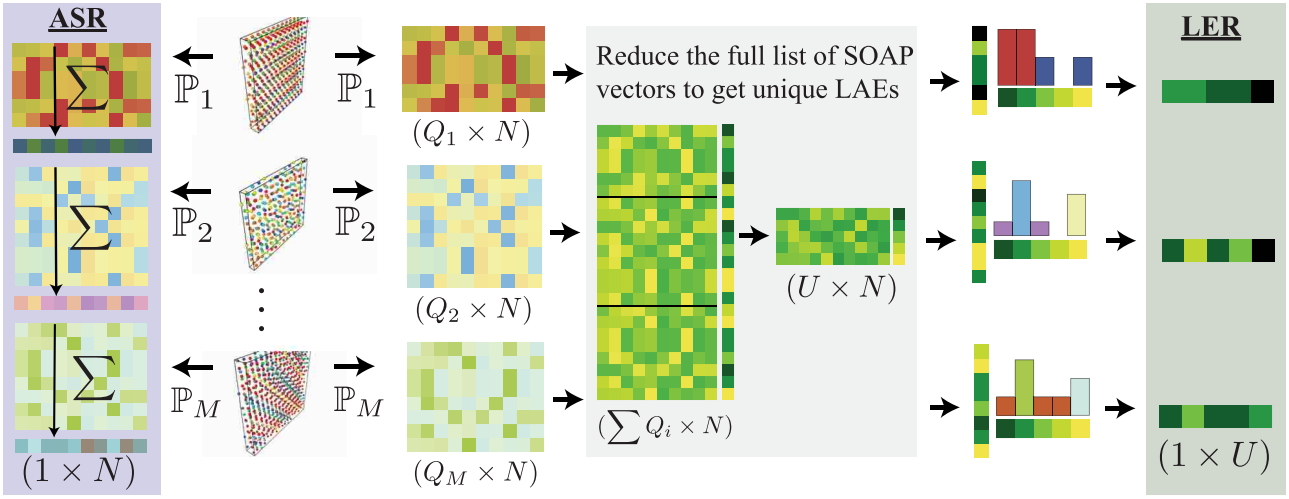


Fig. 2 Illustration of the process for construction of the ASR and LER for a collection of GBs. First, a SOAP matrix $\mathbb{P}$ is formed (as shown in Fig. 1). ASR: A sum down each of the Q columns in the matrix produces an averaged SOAP vector that is representative of the whole GB. The ASR feature matrix is then the collection of averaged SOAP vectors for all M GBs of interest ($M \times N$). LER: The SOAP vectors from all M GBs in the collection are grouped together and reduced to a set U of *unique* vectors using the SOAP similarity metric, of which each unique vector represents a unique LAE. A histogram can then be constructed for each GB counting how many examples of each unique vector are present in the GB. This histogram produces a new vector (the LER) of fractional abundances, whose components sum to 1. The LER feature matrix is then the collection of histograms of unique LEA for the M GBs in the collection ($M \times U$)

Table 1. Predictive performance of the machine learning models trained on the ASR and LER representation, respectively

Property	ASR (ML model)	LER (ML model)	Random
GB energy	89.2 ± 0.7% (RBF SVM)	88.5 ± 0.9% (GBT)	70.4 ± 1.6%
Temperature-dependent mobility	77.4 ± 2.5% (linear SVM)	74.3 ± 2.7% (GBT)	38.5 ± 2.0%
Shear coupling	61.3 ± 0.6% (linear SVM)	61.4 ± 0% (GBT)	52.0 ± 2.5%

The models were trained on 50% (194) of the available 388 GBs and then validated on the remaining 194 GBs that the model had never seen. Percent error is relative to the mean. Error bars represent the standard deviation over 50 independent, random samplings (including different combinations of the 50% split), and re-fits of the data set. For the random column, energies were guessed by drawing values from a normal distribution that had the same mean and standard deviation as the 50% training data, and then compared to the actual energies in the validation data. For the classification problems, random choices from the 50% training data class labels were compared to the validation data. The machine learning models used were (1) support vector machine (SVM) with either a linear or radial basis function (RBF) kernel; or (2) gradient-boosted decision tree (GBT). Parameters for each model are discussed in the Supplementary Information

In the present work, 800,000 LAEs from the atoms in 388 GBs are reduced to 145 unique LAEs. This is a considerable reduction in dimensionality for a machine learning approach. More importantly, these 145 unique LAEs mean that there may be a relatively small, finite set of LAEs used to construct every possible GB in Ni. Using the reduced set of unique LAEs, we represent each GB as a vector whose components are the fraction of each globally unique LAE in that GB. This GB representation is referred to as the Local Environment Representation (LER), and the matrix of LER vectors representing a collection of GBs is also a feature matrix for machine learning. The 145 unique LAEs give a bounded 145-dimensional space, which is a significant improvement over the $3 \times 800{,}000$-dimensional space of the GB data set.

These two approaches are used because they are complementary: physical quantities such as energy, mobility, and shear coupling are best learned from the ASR, while physical interpretability is accessible using the LER, with only marginal loss in predictive power. Because we desire to discover the underlying physics and not just provide a black-box for property prediction, we use the LER to deepen our understanding of which LAEs are most important in predicting material properties such as mobility and shear coupling.

GB energy is measured as the excess energy of a grain boundary relative to the bulk energy as a result of the irregular structure of the atoms in the GB.[12, 42] GB energy is a static property of the system measured at 0 K, and all atomistic structures examined in the machine learning are the 0 K structures associated with this calculation.

Temperature-dependent mobility and shear coupled GB migration are two dynamic properties related to the behavior of a GB when it migrates. The temperature-dependent mobility trend classifies each GB as having (i) *thermally activated*, (ii) *athermal*, or (iii) *thermally damped* mobility, depending on whether the mobility of the GB (related to the migration rate) increases, is constant, or decreases with increasing temperature.[35] GBs that do not move under any of these conditions are classified as being (iv) *immobile*. In addition, when GBs migrate, they can also exhibit a coupled shear motion, in which the motion of a GB normal to its surface couples with lateral motion of one of the two crystals.[34, 43] GBs are then classified as either exhibiting shear coupling or not.

GB energy is a continuous quantity, while temperature-dependent mobility trend and shear coupling are classification properties. Additional details regarding these properties are available in the publications pertaining to their measurements[12,

[34, 35] and in Section IV of the Supplementary Information. For the mobility and shear coupling classification, the data set suffered from imbalanced classes; we used standard machine learning resampling techniques to help mitigate the problem.[44–46]

RESULTS

A summary of the machine learning predictions by the various methods is provided in Table 1. Machine learning was performed using the ASR and LER descriptions of the GBs and the properties of interest for the learning and prediction are GB energy, temperature-dependent mobility, and shear coupled GB migration (obtained from the computed Ni GBs). Table 1 also includes the results of attempting to predict these properties by "educated" random guessing using knowledge of the statistical behavior of the training set. For example, GB energies were guessed by drawing values from a normal distribution that had the same mean and standard deviation as the 50% training data; for the classification problems, random choices from the class labels in the training data were used. In all cases, the machine learning predictions are significantly better than random draws from distributions.

At first glance, the performance for mobility trend and shear-coupling classification (reported in Table 1) may seem mediocre. The results are significant, however, because mobility trend and shear-coupling are *dynamic* quantities, but they were predicted using a representation based on the *static*, 0 K GB structures. The mobility trend results are exceptional because the authors are unaware of any other models that can predict mobility using only knowledge of the atomic positions at the GB.

Shear coupling predictions are a little disappointing, but show some important limitations of the approach and suggest possible physical insights. Since little correlation was found between local environment descriptions and shear coupling, it may imply that the physical phenomenon must be multi-scale. Both the ASR and LER use knowledge of the *local* environments around atoms, but do not consider longer-range interactions between LAEs. Thus, only physical information within the cutoff (5 Å in this case) is considered. A future avenue of research could investigate whether connectivity of LAEs at multiple length scales or the full GB network are responsible for shear coupling.

Unfortunately, the size of the data set is a limiting factor in the performance of the machine learning models. In Table 1, we used only half of the available 388 GBs for training. As we increase the amount of training data given to the machine, the learning rates change as shown in Fig. 3. Although it is common practice to use up to 90% of the available data in a small data set for training (with suitable cross validation), we chose to use a lower (pessimistic) split to guarantee that we are not overfitting to non-physical features. Larger data sets would certainly improve the models and our confidence in the physics they illuminate.

DISCUSSION

For small data sets, ASR does slightly better in predicting energy and temperature-dependent mobility trend; ASR and LER are essentially equivalent for shear coupling. However, the ASR methodology suffers from a lack of interpretability because (1) its vectors and similarity metric live in the abstract SOAP space, which is large and less intuitive; (2) the results reported for ASR were obtained using a support vector machine (SVM), which is not easily interpretable. Details on the algorithm types and interpretation are included in the Supplementary Information. The LER, on the other hand, has direct analogues in LAEs that can be analyzed in their original physical context. The best-performing algorithms for the LER are gradient-boosted decision trees, which lend themselves to easy interpretation. The fitted, gradient-boosted decision trees can be analyzed to determine which of the

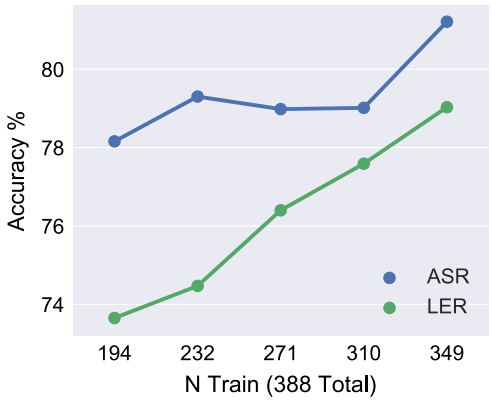


Fig. 3 Learning rate of ASR vs. LER for mobility classification. The *x*-axis is the number of GBs used in the training set, with the remaining GBs held out for validation. The accuracy was calculated over 25 independent fits. It appears that the LER accuracy increases slightly faster with more data, though a larger data set is necessary to confidently establish this point

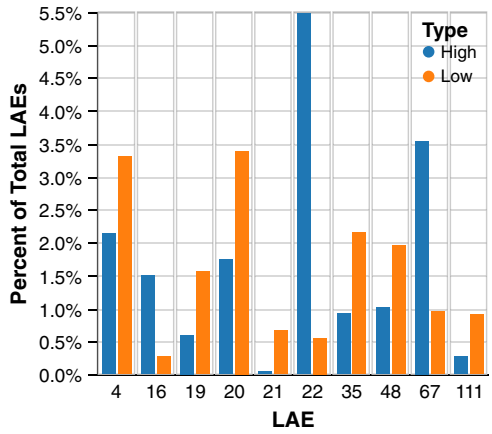


Fig. 4 Histogram comparing the fraction of total LAEs for high and low GB energy. The 15 highest-energy GBs are compared against the 15 lowest-energy GBs using the 10 most-important LAEs for energy prediction. There are clear differences between the relative abundances of these LAEs in deciding whether a GB will have high or low energy

LAEs in the LER are most important. We used information gain as the metric for determining LAE importance, and an example is discussed in Section II of the Supplementary Information. Thus, even at slightly lower accuracy, the physical insights generated by the LER make it the superior choice.

In Fig. 4, we compare the relative abundances of the most important LAEs for high and low energy classification in GBs. The 15 highest- and lowest-energy GBs are compared by calculating the fraction of their LAEs which are in the same class as the 10 most important LAEs for energy prediction. The most important LAEs selected by the machine learning algorithm are good at distinguishing between high and low energy GBs.

In Figs. 5 and 6, we plot some of the most important environments for determining whether a grain boundary will exhibit thermally activated mobility or not (Fig. 5) or thermally damped mobility or not (Fig. 6). These most important LAEs are classified as such because their presence or absence in any of the GBs in the entire data set is highly correlated with the decision to classify them as thermally activated or not, or thermally damped or not. Since such global correlations must be true for all GBs in

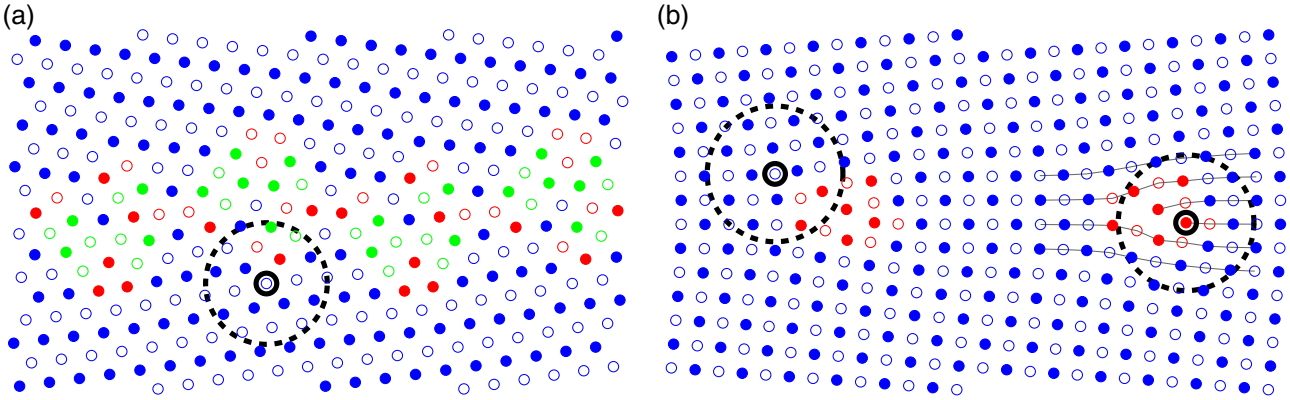


Fig. 5 Illustration of important LAEs for classifying thermally activated GB mobility, as identified in two different GBs. The GB shown in **a** is a Σ51a (16.1° symmetric tilt about the [110] axis, $\{1\bar{1}10\}$ boundary planes) GB, and has one LAE identified. The LAE shown in **a** has a relative importance of 3% over the entire system and includes a leading partial dislocation that originates from the GB. The GB shown in **b** is a Σ85a (8.8° symmetric tilt about the [100] axis, $\{0\bar{1}13\}$ boundary planes) GB, and has two LAEs identified. The *leftmost* LAE has a relative importance of 9% (for all GBs in the data set) but its structural importance is not immediately clear, offering an exciting opportunity to discover new physics. The second LAE in **b** encloses edge dislocations, which are regularly spaced to form a tilt GB, (relative importance of 2.7% across all GBs). The *open* and *filled circles* denote atoms on the two unique stacking planes along the [100] or [110] direction. The atoms are colored according to common neighbor analysis (CNA) such that *blue*, *green*, and *red* atoms have a local environment that is FCC, HCP, or unclassifiable

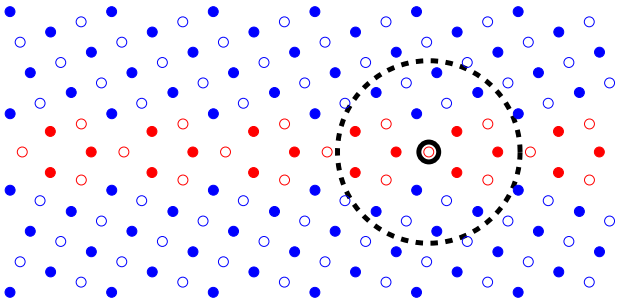


Fig. 6 Illustration of the most important LAE for classifying thermally damped GB mobility, as identified in a Σ5 (36.9° symmetric tilt about the [100] axis, $\{0\bar{1}3\}$ boundary planes) GB. The LAE shown has a relative importance of 6.8% and is centered at the point of a kite structure but includes parts of the kites on either side. These kite structures are "C" structural units that are regularly observed in [100] axis symmetric tilt GBs. The *open* and *filled* circles denote atoms on the two unique stacking planes along the [100] direction. The atoms are colored according to common neighbor analysis (CNA) such that blue, and red atoms have a local environment that is FCC, or unclassifiable

the system, we assume that they are tied to underlying physical processes.

Figure 5a shows a LAE centered around a leading partial dislocation. GBs with partial dislocations emerging from the structure have been associated with thermally activated mobility and immobility, depending upon their presence in simple or complex GB structures;[34] in addition, these structures have also been associated with shear coupled motion or the lack thereof. We now know that there is a strong correlation between the presence of these LAEs and their mobility type, though the presence of other structures is also important in the determination of the exact mobility type. This LAE was presented on equal footing with all others in the feature matrix that trained the machine. In the training, it was selected as important and we can easily see that it has relevant physical meaning.

In Fig. 5b, another LAE has obvious physical meaning as it captures edge dislocations in the environment of the selected atom. Interestingly, arrays of these edge dislocations, as in Fig. 5b, are the basis for the energetic structure-property relationship of the Read-Shockley model.[27]

Thus, in these first two cases, we see that the LER approach discovers well-known, and physically important structures or defects that are commonly identified in metallic structures. Perhaps even more interesting is the second LAE in Fig. 5b, which has the highest relative importance of all (≈9%). The centro-symmetry parameter (CSP) for the atom at the center of the LAE is 0.125, or close to a perfectly structured FCC lattice, as visual inspection of the LAE would suggest. However, the CSP cannot be directly compared with the LAE because CSP examines only nearest neighbors while the LAE encompasses a larger environment, including the defect at the edge of the LAE.[47] Most importantly, this structure may not be immediately identified with any known metallic defect, but we know that it is highly correlated with thermally activated mobility across *all* the GBs in the data set.

In Fig. 6, the most important LAE for predicting thermally damped mobility is shown. Interestingly, it has found the "C" structural unit that is readily found in [100] axis symmetric tilt GBs,[26] though the LAE spans multiple kite structures. More important to note, however, is the fact that most of the important LAEs for predicting thermally damped mobility, are LAEs that are *not* present in thermally damped GBs. In other words, the machine learning algorithm is able to determine which structures will exhibit thermally damped mobility by the lack of certain LAEs in those structures.

The machine can determine some LAEs that are associated with well-known structures and properties, while also finding other LAEs that are not readily recognizable but are apparently important. This fact offers an exciting avenue to discover new mechanisms and structures governing physical properties. The physical nature of those LAEs that we already understand suggests that these are the building blocks underlying important physical properties and that we may be on the precipice of understanding the atomic building blocks of GBs.

Despite the formidable dimensionality of a raw grain boundary system, machine learning using SOAP-based representations makes the problem tractable. In addition to learning useful physical properties, the models provide access to a finite set of physical building blocks that are correlated with those properties throughout the high-dimensional GB space. Thus, the machine learning is not just a black box for predictions that we do not understand. The work shows that analyzing big data regarding materials science problems can provide insight into physical

structures that are likely associated with specific mechanisms, processes, and properties but which would otherwise be difficult to identify. Accessing these building blocks opens a broad spectrum of possibilities. For example, the reduced space can now be searched for extremal properties that are unique (i.e., special GBs). Poor behavior in certain properties can be compensated for by searching for combinations of other properties. In short, a path is now available to develop methods that optimize GBs (at least theoretically) at the atomic-structure scale. These methods may also provide a route to connect the crystallographic and atomic structure spaces so that existing expertize in the crystallographic space can be further optimized atomistically or vice versa.

While this is exciting within grain boundary science, the methodology presented here (and the SOAP descriptor in particular) has general applicability for building order parameters while studying changes that involve local structure. For example, it can be applied in studying phase change materials, point defects in solids, amorphous materials, cheminformatics, and drug binding. The physical interpretability of the machine learning representations, in terms of atomic environments, will also transfer well to new applications. This can lead to increased physical intuition across many fields of research that are confronted with the same, formidable complexity as seen in grain boundary science.

Data availability

Additional details about the machine learning models and data are described in the accompanying Supplementary Information. The feature matrices and code to generate them are available on request.

ACKNOWLEDGEMENTS

C.W.R. and G.L.W.H. were supported under ONR (MURI N00014-13-1-0635). E.R.H. is supported by the U.S. Department of Energy, Office of Science, Basic Energy Sciences under Award #DE-SC0016441.

AUTHOR CONTRIBUTIONS

C.W.R. conceived the idea, performed all the calculations, and wrote a significant portion of the paper. E.R.H. was responsible for interpretation of the results and guidance of the project, and also wrote a significant portion of the paper. G.C. provided code, guidance and expertize in applying SOAP to the GBs. G.L.W.H. contributed many ideas and critique to help guide the project, and helped write the paper.

ADDITIONAL INFORMATION

Supplementary Information accompanies the paper on the *npj Computational Materials* website (doi:10.1038/s41524-017-0027-x).

Competing interests: The authors declare that they have no competing financial interests.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

REFERENCES

1. Hall, E. O. The deformation and ageing of mild steel: III discussion of results. *Proc. Phys. Soc. B* **64**, 747–753 (1951).
2. Petch, N. J. The cleavage strength of polycrystals. *J. Iron Steel Inst.* **174**, 25–28 (1953).
3. Hansen, N. Hall–Petch relation and boundary strengthening. *Scripta Mater.* **51**, 801–806 (2004).
4. Chiba, A., Hanada, S., Watanabe, S., Abe, T. & T, Obana Relation between ductility and grain-boundary character distributions in Ni_3Al. *Acta Metall. Mater.* **42**, 1733–1738 (1994).
5. Fang, T. H., Li, W. L., Tao, N. R. & Lu, K. Revealing extraordinary intrinsic tensile plasticity in gradient nano-grained copper. *Science* **331**, 1587–1590 (2011).
6. Shimada, M., Kokawa, H., Wang, Z. J., Sato, Y. S. & Karibe, I. Optimization of grain boundary character distribution for intergranular corrosion resistant 304 stainless steel by twin-induced grain boundary engineering. *Acta Mater.* **50**, 2331–2341 (2002).
7. Lu, L. Ultrahigh strength and high electrical conductivity in copper. *Science* **304**, 422–426 (2004).
8. Bagri, A., Kim, S.-P., Ruoff, R. S. & Shenoy, V. B. Thermal transport across twin grain boundaries in polycrystalline graphene from nonequilibrium molecular dynamics simulations. *Nano Lett.* **11**, 3917–3921 (2011).
9. Meyers, M. A., Mishra, A. & Benson, D. J. Mechanical properties of nanocrystalline materials. *Progr. Mat. Sci.* **51**, 427–556 (2006).
10. Wolf, D. & Yip, S. (eds.) *Materials Interfaces: Atomic-Level Structure and Properties* (Chapman & Hall, London, 1992).
11. Sutton, A. & Balluffi, R. *Interfaces in Crystalline Materials* (Oxford University Press, 1995).
12. Olmsted, D. L., Foiles, S. M. & Holm, E. A. Survey of computed grain boundary properties in face-centered cubic metals: I. Grain boundary energy. *Acta Mater.* **57**, 3694–3703 (2009).
13. Cantwell, P. R. et al. Grain boundary complexions. *Acta Mater.* **62**, 1–48 (2014).
14. The interplay between grain boundary structure and defect sink/annealing behavior *IOP Conference Series: Materials Science and Engineering* **89**, 012004 (2015).
15. Dillon, S. J., Tai, K. & Chen, S. The importance of grain boundary complexions in affecting physical properties of polycrystals. *Curr. Opin. Solid State Mater. Sci.* **20**, 324–335 (2016).
16. Weins, M., Chalmers, B., Gleiter, H. & ASHBY, M. Structure of high angle grain boundaries. *Scripta Metall. Mater.* **3**, 601–603 (1969).
17. Ashby, M. F., Spaepen, F. & Williams, S. Structure of grain boundaries described as a packing of polyhedra. *Acta Metall. Mater.* **26**, 1647–1663 (1978).
18. Gleiter, H. On the structure of grain boundaries in metals. *Mater. Sci. Eng.* **52**, 91–131 (1982).
19. Frost, H. J., Ashby, M. F. & Spaepen, F. A catalogue of [100], [110], and [111] symmetric tilt boundaries in face-centered cubic hard sphere crystals. *Harvard Div. Appl. Sci.* 1–216 (1982).
20. Sutton, A. P. On the structural unit model of grain boundary structure. *Phil. Mag. Lett.* **59**, 53–59 (1989).
21. Wolf, D. Structure-energy correlation for grain boundaries in FCC metals—III. Symmetrical tilt boundaries. *Acta Metall. Mater.* **38**, 781–790 (1990).
22. Tschopp, M. A., Tucker, G. J. & McDowell, D. L. Structure and free volume of symmetric tilt grain boundaries with the E structural unit. *Acta. Mater.* **55**, 3959–3969 (2007).
23. Tschopp, M. A. & McDowell, D. L. Structural unit and faceting description of Sigma 3 asymmetric tilt grain boundaries. *J. Mater. Sci.* **42**, 7806–7811 (2007).
24. Spearot, D. E. Evolution of the E structural unit during uniaxial and constrained tensile deformation. *Acta Mater.* **35**, 81–88 (2008).
25. Bandaki, A. D. & Patala, S. A three-dimensional polyhedral unit model for grain boundary structure in fcc metals. *Npj Comput. Mater.* **3**, 13 (2017).
26. Han, J., Vitek, V. & Srolovitz, D. J. The grain-boundary structural unit model redux. *Acta Mater.* **133**, 186–199 (2017).
27. Read, W. & Shockley, W. Dislocation models of crystal grain boundaries. *Phys. Rev.* **78**, 275–289 (1950).
28. Frank, F. C. Martensite. *Acta Metall. Mater.* **1**, 15–21 (1953).
29. Bilby, B. A., Bullough, R. & Smith, E. Continuous distributions of dislocations: a new application of the methods of non-riemannian geometry. *Proc. Roy. Soc. A-Math. Phy.* **231**, 263–273 (1955).
30. Wolf, D. A broken-bond model for grain boundaries in face-centered cubic metals. *J. Appl. Phys.* **68**, 3221–3236 (1990).
31. Wolf, D. Correlation between structure, energy, and ideal cleavage fracture for symmetrical grain boundaries in fcc metals. *J. Mater. Res.* **5**, 1708–1730 (1990).
32. Yang, J. B., Nagai, Y. & Hasegawa, M. Use of the Frank–Bilby equation for calculating misfit dislocation arrays in interfaces. *Scripta Mater.* **62**, 458–461 (2010).
33. Olmsted, D. L., Holm, E. A. & Foiles, S. M. Survey of computed grain boundary properties in face-centered cubic metals-II: Grain boundary mobility. *Acta Mater.* **57**, 3704–3713 (2009).
34. Homer, E. R., Foiles, S. M., Holm, E. A. & Olmsted, D. L. Phenomenology of shear-coupled grain boundary motion in symmetric tilt and general grain boundaries. *Acta Mater.* **61**, 1048–1060 (2013).
35. Homer, E. R., Holm, E. A., Foiles, S. M. & Olmsted, D. L. Trends in grain boundary mobility: survey of motion mechanisms. *JOM* **66**, 114–120 (2014).
36. Bulatov, V. V., Reed, B. W. & Kumar, M. Grain boundary energy function for fcc metals. *Acta. Mater.* **65**, 161–175 (2014).

37. Homer, E. R., Patala, S. & Priedeman, J. L. Grain boundary plane orientation fundamental zones and structure-property relationships. *Sci. Rep* **5**, 15476 (2015).
38. Kiyohara, S., Miyata, T. & Mizoguchi, T. Prediction of grain boundary structure and energy by machine learning arXiv:1512.03502 (2015).
39. Bartók, A. P., Payne, M. C., Kondor, R. & Csányi, G. Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons. *Phys. Rev. Lett.* **104**, 136403 (2010).
40. Bartók, A. P., Kondor, R. & Csányi, G. On representing chemical environments. *Phys. Rev. B* **87**, 184115 (2013).
41. Foiles, S. M. & Hoyt, J. J. Computation of grain boundary stiffness and mobility from boundary fluctuations. *Acta Mater.* **54**, 3351–3357 (2006).
42. Tadmor, E. B. & Miller, R. E. *Modeling Materials: Continuum, Atomistic and Multiscale Techniques* (Cambridge University Press, 2011).
43. Cahn, J. W., Mishin, Y. & Suzuki, A. Coupling grain boundary motion to shear deformation. *Acta Mater.* **54**, 4953–4975 (2006).
44. Han, H., Wang, W. -Y. & Mao, B. -H. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *Proceedings of the 2005 International Conference on Advances in Intelligent Computing - Volume Part I*, ICIC'05, 878–887 (Springer-Verlag, Berlin, Heidelberg, 2005).
45. Nguyen, H. M., Cooper, E. W. & Kamei, K. Borderline over-sampling for imbalanced data classification. *Int. J. Knowl. Eng. Soft Data Paradigm* **3**, 4–21 (2011).
46. Lemaître, G., Nogueira, F. & Aridas, C. K. Imbalanced-learn: a python toolbox to tackle the curse of imbalanced datasets in machine learning. *CoRR* abs/1609.06570 (2016).
47. Kelchner, C. L., Plimpton, S. J. & Hamilton, J. C. Dislocation nucleation and defect structure during surface indentation. *Phys. Rev. B* **58**, 11085–11088 (1998).

CHAPTER 6

Software Packages

The works presented in the previous chapters drove development of several software packages, some of which are useful is multiple contexts.

6.1 MACHINE LEARNING FOR GRAIN BOUNDARIES

https://github.com/rosenbrockc/gblearn

The analysis presented in Chapter 5 was refactored and combined into a high-level API that allows the full analysis presented in the paper to be completed in five lines of code and about 15 minutes. We are encouraging a new paradigm in computational science where all results are accompanied by docker containers that have all the necessary code and dependencies to reproduce the results in an afternoon. The high-level API produced for the GB machine learning project is a step toward this paradigm.

We are currently working on a methods paper that includes the additional details that couldn't fit into the original paper and which shows how to reproduce the results using very few lines of code. Upon publication, the source code will be available at the above URL.

6.2 NUMERICAL ALGORITHM FOR PÓLYA ENUMERATION

https://github.com/rosenbrockc/polya

As discussed in Chapter 2, the algorithm to count the number of symmetrically unique colorings of a lattice is available for both `python` and `fortran`.

6.3 AUTO-COMPLETE AND UNIT TESTING FOR FORTRAN

https://github.com/rosenbrockc/fortpy

One of the most important skills a computational scientist can develop is to adopt industry-standard development protocols for the code they write. This includes each of the following:

- Committing early and often to a source control repository so that multiple researchers can work together on the same project and so that errors can easily be corrected by rolling code back to an earlier version.
- Requiring full unit test coverage of any code produced; this means that there is at least one test for every line of code. A test is simply an input/output pair where the output has been verified by the coder to be the correct answer. As long as the code produces the correct output for each input, it is doing what it was designed to do.
- Implementing a continuous integration (CI) server. Whenever a code change is being proposed for the project, the CI server takes the current code and the proposed changes and merges them offline so that unit tests can be run against the proposed code base. This ensures that the changes don't break existing functionality.
- Protecting the master branch so that code changes *cannot* be merged into the master branch if test coverage drops or any of the unit tests fail. This ensures consistent results across versions.
- Having internally documented source code so that documentation is automatically kept up to date as the code changes.

Since a lot of our development takes place in `fortran`, I developed a package called `fortpy` that provides:

- Automated unit testing for `fortran` subroutines and functions. Only input and output need to be specified and the framework will generate the program files, compile the code and run the tests. Profiling, timing and other conveniences are also handled automatically.
- Context-dependent code completion for emacs; this reduces development time by providing hints about signatures for functions or subroutines and members of user-defined types.
- Code checking for bugs that are otherwise hard to find; for example user-defined types that have uninitialized pointers being passed as output parameters.

`fortpy` is currently being used by twelve groups worldwide (outside of BYU). We have used it internally for development and unit testing of several codes, including:

1. The Pólya code discussed in Chapter 2.
2. The enumeration and symmetry codes referenced in Chapter 2.
3. A scattering transform code for machine learning alloy potentials [72, 73].
4. Bayesian Compressive Sensing solver discussed below in Section 6.5.

6.4 API FOR SEARCHING AFLOWLIB

https://github.com/rosenbrockc/aflow

As discussed above, recent advances in computation have enabled the creation of large, materials databases using Density Functional Theory [5, 6]. `aflowlib` [74, 75] is one of the largest with more than 1.7M material compounds (as of November 2017, http://aflowlib.org/). Recently, the AFLUX search API [76] was introduced to provide improved access to the materials' data in a uniform request format via REST [77]. Because the API is based on REST, it allows access to the data from a variety of programming languages through standard libraries.

Unfortunately, the data for material properties and calculation parameters are not stored in a standard format. While the custom serialization format is documented, each property must be parsed individually to access standard formats (such as `numpy` [78] arrays for python). Thus, a researcher attempting to access `aflowlib` data for the first time must 1) read and understand the AFLUX request format; 2) lookup the documentation for the properties of interest and 3) deserialize them appropriately. `aflowlib` fields are stored as strings of values that may be comma-separated, colon-separated or have a more complex structure (such as for the `kpoints` property). Deserialization refers to the transformation of these strings into high-level objects such as dictionaries or arrays. Even though such tasks are well within the abilities of a computational scientist, they are not tasks that leverage scientific expertise.

I developed a high-level python API that abstracts the request and deserialization tasks so that there is virtually no access barrier for newcomers to the `aflowlib` database. An introduction and instructions are published [79] and the code is available open source at the above URL and via the python package index using `pip install aflow`.

This package is an essential component of the alloy potential automation project that was briefly mentioned in the concluding remarks of

the introductory chapter.

6.5 BAYESIAN COMPRESSIVE SENSING SOLVER

https://github.com/rosenbrockc/bcs

A difficult problem in Cluster Expansion was solved by Nelson et al. using Bayesian Compressive Sensing (BCS) [61, 62]. BCS allows underdetermined linear systems to be solved by minimizing the ℓ_2 error in the approximating function subject to minimization of the ℓ_1 norm. Much of the work in developing the algorithm was done by Lance Nelson. I refactored the core code to be stand-alone so that it could easily be applied to other problems outside of the Cluster Expansion context.

The code has been used by collaborators for doing lattice dynamics with compressive sensing [80] and was used extensively for the robustness analysis discussed in Chapter 3.

6.6 AUTOMATED CLUSTER EXPANSION

Before we realized that Cluster Expansion (CE) would be replaced by alloy potentials that can handle lattice dynamics, I refactored some scripts provided by Lance Nelson and expanded the library to create a package called `ancle` that fully automates the various steps of CE from creation of the independent and identically distributed structures for DFT, through fitting, verification and finally searching for stable ground states to create the convex hull. It was designed to be hands-off so that all aspects of the workflow would be handled by scheduled tasks on the server.

We had planned to let `ancle` loose to create CE models for every possible binary alloy system, a project we called CE-flash, but which was waiting for the k-point problem to be solved before execution. Although it wasn't used for its intended purpose, the components have been re-used many times in various contexts. The alloy potential automation project supersedes this.

6.7 AUTOMATED COMPUTATIONAL RESEARCH NOTEBOOK

https://github.com/rosenbrockc/acorn

In the spirit of scientific reproducibility, we set out to create an auto-documenting research notebook for computational science. Since `ipython` notebooks have become the defacto standard for computational science, it seemed to be a logical place to inject the automation.

`acorn` is an open-source project that provides the following features:

- Automatic "smart" logging of all code executed in an `ipython` notebook. By "smart", we mean that the logging system is aware of variables being passed to functions and methods and internally documents and gathers statistics on the nature of the variables at execution time, rather than just logging the text that was typed into the cell.
- Logging of markdown typed between code cells.
- Logging of plot thumbnails as they are produced by plotting functions.
- Customizable filters to decide which code calls are logged and which are ignored.

`acorn` works by wrapping all function calls with new versions that provide the variable-sensitive logging functionality before and after executing the original call. While simple to state, the internals represent some of the most complex python coding I have ever done.

6.8 MACHINE-LEARNED ALLOY POTENTIAL AUTOMATION

https://github.com/rosenbrockc/matdb

One of the shortcomings of historical approaches to creating interatomic potentials is scientific reproducibility. Almost all of the published potentials *only* include details as to how the training databases were created, but do not provide access to those databases or include the necessary code and parameters to recreate them from scratch. This makes it difficult to reproduce the published results or to improve upon existing potentials. For example, creating a machine-learned alloy potential for comparison with a published Embedded Atom Method potential is difficult in most cases because the training databases are not available and cannot easily be recreated using only the publication as a reference.

Inasmuch as the fitting of alloy potentials using machine learning is essentially an automated process, the bottleneck to successful potential creation is now the training databases. Recent successes [81, 82] required 3 years of hand-tuning by students and postdocs to generate such excellent potentials. The `matdb` project provides scientifically rigorous methods to generate alloy databases using a single file that contains all parameters needed to create the training database from scratch and to train the machine-learned potential. Once the final potential has been generated by trial and error, the project provides a `finalize` function that packages the database, potential and all the code needed to recreate them into a single archive that can be referenced in a publication.

Abstracting the creation and fitting functionality behind a single file that contains only free parameters has several advantages:

1. Scientists can make high-level decisions about the information content of the training databases and the flexibility of the representation *without* worrying about the implementation details.

2. In a high-throughput situation, the scripts for automating the calculations only have to configure the high-level specification file because the details are taken care of.

3. As bug-fixes and enhancements roll out, existing databases and specification files still work because of the abstraction layer.

While `matdb` is being used actively for our alloy automation project, it is still private while we work through stability issues, etc. We plan to release it publicly for academic use once we have proved it.

APPENDIX A

Equating Cluster Expansion with the SOAP Kernel

As discussed above, an exciting part of the continuing research into machine learning the quantum physics of alloys involves equating the Cluster Expansion basis and Smooth Overlap of Atomic Positions kernel to extract the space-dependent species interaction terms. We have already invested several years into this project and it is nearing completion. As soon as we demonstrate machine-learned alloy potentials that are phonon-accurate, we will be able to equate the two approaches using the mathematics presented here.

A.1 GAUSSIAN PROCESS REGRESSION FOR SITE ENERGIES

The total energy E is assumed to be the sum of site energies given by a universal site energy function ε that is a function of the neighbour environment, which has compact support and is described by a "descriptor" vector d. Define the list of atomic positions and species of the atoms as

$$|\vec{\rho}\,\rangle = |\vec{r}_1^{\,(s_1)}, \vec{r}_2^{\,(s_2)}, \ldots, \vec{r}_{N_{\text{sites}}}^{\,(s_{N_{\text{sites}}})}\rangle. \tag{A.1}$$

Then we can write the total energy for any crystal (a single point in the space $|\vec{\rho}\,\rangle$) as

$$\begin{aligned} \langle \vec{\rho}\,|E\rangle &= \sum_i^{N_{\text{sites}}} \varepsilon(\langle \vec{\rho}\,|d_i(\vec{r})\rangle) \\ &= \sum_i \varepsilon(\langle \vec{\rho}\,|d_i\rangle). \end{aligned} \tag{A.2}$$

As an example, d_i might be the charge density in the neighborhood of site i expanded as a sum of Gaussians (see Eq. (A.4)). $\langle \vec{\rho}\,|d_i\rangle$ defines the parameter space over which d is defined; for each value of $\vec{r}$ within a crystal (parameterized by $\vec{\rho}$), d returns a scalar. It could thus also be written equivalently as $d_i(\vec{r}, \vec{r}_1^{\,(s_1)}, \vec{r}_2^{\,(s_2)}, \ldots, \vec{r}_{N_{\text{sites}}}^{\,(s_{N_{\text{sites}}})})$. However, we retain the notation for clarity later on.

In Gaussian process regression, the site energy is given by a linear combination of basis functions (kernels), each centered on a data point in descriptor space,

$$\varepsilon(\langle \vec{\rho}\,|d\rangle) = \sum_i^{\text{database}} \alpha_i K(\langle \vec{\rho}\,|d_i\rangle, \langle \vec{\rho}\,|d\rangle) \tag{A.3}$$

where α are the coefficients obtained from the regularized fit. The database for the regularization is described in detail by Szlachta et al. [82]. Essentially, for a given material with total energy calculated using DFT, the site energies are extracted analytically with the Hellmann-Feynmann theorem; thus a crystal with 5 atoms in the basis will produce 5 entries in the database. Since many local environments are similar, the database does not necessarily include all atomic descriptors from every material in the database.

An additional note for clarity: the $|\vec{\rho}\,\rangle$ above describes the space of all possible materials that can be formed using a list of atomic positions with corresponding species information at each one.

Mathematically, it is a direct sum of N_{species} copies of $\mathbb{R}^3$, one for each atom, with corresponding species labels. The descriptor contraction $\langle\vec{\rho}\,|d_i\rangle$ in Eq. (A.3) shows that the *descriptor* d_i used in the kernel regression was evaluated for some specific member of the space $|\vec{\rho}\,\rangle$.

A.2 SOAP KERNEL INTRODUCTION

The kernel compares environments, and needs to take the value 1 only if the two environments are the same up to symmetries (global rotations and permutations of similar atoms). We define the neighbour density of atom i as

$$\begin{aligned}\langle\vec{\rho}\,|\rho_{is}\rangle &= \sum_{\vec{r}_{ij}\in|\vec{\rho}\,\rangle} e^{-(\vec{r}_{ij}-\vec{r})^2/2\sigma_{\text{atom}}^2} f_{\text{cut}}(|\vec{r}_{ij}|)\delta_{ss_j}\\ \langle\vec{\rho}\,|\rho_i\rangle &= \bigoplus_{s=1}^{n_{\text{species}}} \langle\vec{\rho}\,|\rho_{is}\rangle,\end{aligned} \tag{A.4}$$

where f_{cut} is a smooth cutoff function that ensure compact support, and s_j is the species of the atom at site j.

The overlap of two different site environments is defined as

$$\begin{aligned}S(\langle\vec{\rho}\,|\rho_i\rangle,\langle\vec{\rho}\,|\rho_k\rangle) &= \int d^3r \langle\vec{\rho}\,|\rho_i\rangle^\dagger\langle\vec{\rho}\,|\rho_k\rangle\\ &= \sum_{ss'}^{\text{species}} \int \langle\vec{\rho}\,|\rho_{is}\rangle^\dagger\langle\vec{\rho}\,|\rho_{ks'}\rangle\langle\vec{\rho}\,|\varsigma_{ss'}(\vec{r})\rangle\, dr.\end{aligned} \tag{A.5}$$

with $\langle\vec{\rho}\,|\varsigma_{ss'}(\vec{r})\rangle$ defining the "overlap of species" s and s'; its value is derived explicitly later. Here we see the first deviation from the original SOAP formalism. The atomic environments are expanded for each species independently; thus, in the absence of a way to describe the overlap function $\varsigma_{ss'}(\vec{r})$, it defaults to $\delta_{ss'}$. In this limit, the formalism only makes sense for pure elements and is *not* concentration dependent. If we examine the mathematical space $|\vec{\rho}\,\rangle$, we find that the species-*independent* SOAP basis is defined over a subspace of $|\vec{\rho}\,\rangle$, the subspace of all pure elements, no matter how the spatial orientation of their atoms changes.

The overlap in Eq. (A.5) is permutationally invariant (because of the sum in ρ_{is}), but not yet rotationally invariant. In order to make it so, we integrate it over all rotations of one of its arguments.

$$\tilde{K}(\langle\vec{\rho}\,|\rho_i\rangle,\langle\vec{\rho}\,|\rho_k\rangle) = \int d\hat{R}\,|S(\langle\vec{\rho}\,|\rho_i\rangle,\langle\vec{\rho}\,|\hat{R}\rho_k\rangle)|^p, \tag{A.6}$$

where $\hat{R}$ is a 3D rotation operator (element of SO(3)), and p is a small integer, e.g. 2. Finally the normalised SOAP kernel is

$$K(\langle\vec{\rho}\,|\rho_i\rangle,\langle\vec{\rho}\,|\rho_k\rangle) = \frac{\tilde{K}(\langle\vec{\rho}\,|\rho_i\rangle,\langle\vec{\rho}\,|\rho_k\rangle)}{\sqrt{\tilde{K}(\langle\vec{\rho}\,|\rho_i\rangle,\langle\vec{\rho}\,|\rho_i\rangle)\tilde{K}(\langle\vec{\rho}\,|\rho_k\rangle,\langle\vec{\rho}\,|\rho_k\rangle)}}. \tag{A.7}$$

Now we derive the efficient formula to evaluate the SOAP kernel, in which the species dependent term will be explicitly seen. For clarity, we show this only for a single overlap between

ρ_{is} and $\rho_{ks'}$. Since the integrals are linear, these steps apply equally well to each term in the sum over ss' in Eq. (A.5).

We start by expanding the neighbour density for each species in an orthonormal basis,

$$\langle \vec{\rho} \,|\rho_{is}(\vec{r})\rangle = \sum_{nlm} \langle \vec{\rho} \,|c_{i,snlm}\rangle \, g_n(r) Y_{lm}(\hat{r}), \tag{A.8}$$

where g are an orthonormal radial basis, Y_{lm} are spherical harmonics, and $\langle \vec{\rho} \,|c_{i,snlm}\rangle$ are the expansion coefficients. The neighbour density corresponding to each species is expanded independently. The effect of the rotation operator acting on ρ_{ks} is written in terms of Wigner matrices,

$$\langle \vec{\rho} \,|\hat{R}\rho_{ks'}(\vec{r})\rangle = \sum_{nlmm'} D^l_{mm'}(\hat{R}) \langle \vec{\rho} \,|c_{k,s'nlm'}\rangle \, g_n(r) Y_{lm}(\hat{r}). \tag{A.9}$$

So the overlap S is given by

$$S(\langle \vec{\rho} \,|\rho_i\rangle, \langle \vec{\rho} \,|\hat{R}\rho_k\rangle) = \sum_{ss'} \sum_{nlmm'} \langle \vec{\rho} \,|\varsigma_{ss'nlmm'}\rangle \langle \vec{\rho} \,|c_{i,snlm}\rangle^\dagger D^l_{mm'}(\hat{R}) \langle \vec{\rho} \,|c_{k,s'nlm'}\rangle. \tag{A.10}$$

where the constants $\langle \vec{\rho} \,|\varsigma_{ss'nlmm'}\rangle$ have been introduced to represent the effect of the species overlap function $\varsigma_{ss'}(\vec{r})$ within each "shell" of the basis $g_n(r)Y_{lm}(\hat{r})$ for an element of $|\vec{\rho}\,\rangle$. We now need to square this expression and integrate over all rotations. This is aided by the formula

$$\int d\hat{R} \, D^l_{mm'}(\hat{R}) D^\lambda_{\mu\mu'}(\hat{R}) = \delta_{l\lambda} \, \delta_{m\mu} \, \delta_{m'\mu'} \frac{1}{\sqrt{2l+1}} \tag{A.11}$$

We get the unnormalised kernel $\tilde{K}(\langle \vec{\rho} \,|\rho_i\rangle, \langle \vec{\rho} \,|\rho_k\rangle) =$

$$\sum_{\substack{ss' \\ s''s'''}} \sum_{\substack{nn' \\ lmm'}} \frac{\langle \vec{\rho} \,|\varsigma_{ss'nlmm'}\rangle \langle \vec{\rho} \,|c_{isnlm}\rangle^\dagger \langle \vec{\rho} \,|c_{ks'n'lm'}\rangle \langle \vec{\rho} \,|\varsigma_{s''s'''n'lmm'}\rangle \langle \vec{\rho} \,|c_{is''n'lm}\rangle^\dagger \langle \vec{\rho} \,|c_{ks'''n'lm'}\rangle}{\sqrt{2l+1}} \tag{A.12}$$

where s and s'' run over species in the neighbourhood of atom i and s' and s''' run over species in the neighbourhood of atom k.

We can write this kernel as a dot product of rotationally invariant descriptors of the two environments which we call the "power spectrum". In order to simplify the notation, we also introduce the set of indices $\mathbf{x} \equiv nl$ and $\mathbf{x}' \equiv n'l$; thus the expression $\sum_{\mathbf{x}} \equiv \sum_{nl}$, and $\sum_{\mathbf{x}'} \equiv \sum_{n'l}$ In doing so, we explicitly retain the indices s and s' to show how the species dependence carries through the derivation. Also, we change the order of the "dummy" indices $\mathbf{x}$ and ss' so that we can refer to them as "pixies" basis functions $|p_{i,\mathbf{x}ss'}\rangle$.

$$\langle \vec{\rho} \,|p_{i,\mathbf{x}ss'}\rangle = \sum_{m=-l}^{l} \langle \vec{\rho} \,|c_{i,snlm}\rangle^\dagger \langle \vec{\rho} \,|c_{i,s'n'lm}\rangle, \tag{A.13}$$

which shows that in the absence of contraction with $|\vec{\rho}\,\rangle$, the $|p_{i,\mathbf{x}ss'}\rangle$ can be thought of as a basis function. We have retained the explicit references to $|\vec{\rho}\,\rangle$ up to this point to emphasize that a specific crystal (a single point in the space $|\vec{\rho}\,\rangle$) can be expanded in a basis of $|p_{i,\mathbf{x}ss'}\rangle$. As long as

we can describe the species overlap function $\varsigma_{ss'}(\vec{r})$ sufficiently well, the $|p_{i,\mathbf{x}ss'}\rangle$ basis *spans the entire materials space* $|\vec{\rho}\rangle$. We can now write the full, un-normalized kernel as

$$\tilde{K}(\langle\vec{\rho}\,|\rho_i\rangle,\langle\vec{\rho}\,|\rho_k\rangle) = \sum_{\substack{\mathbf{x}ss'\\ \mathbf{x}'s''s'''}} \langle\vec{\rho}\,|p_{i,\mathbf{x}ss'}\rangle\,\langle\vec{\rho}\,|p_{k,\mathbf{x}'s''s'''}\rangle\langle\vec{\rho}\,|\kappa_{\mathbf{x}ss'\mathbf{x}'s''s'''}\rangle \tag{A.14}$$

with $\kappa_{\mathbf{x}ss'\mathbf{x}'s''s'''} = \varsigma_{\mathbf{x}ss'}\varsigma_{\mathbf{x}'s''s'''}$. Then, we rewrite the site energy as (using the *normalized* kernel $K(d_i,d_k)$ again):

$$\begin{aligned}
\varepsilon(\langle\vec{\rho}\,|d_k\rangle) &= \sum_{i}^{\text{database}} \alpha_i K(\langle\vec{\rho}\,|d_i\rangle,\langle\vec{\rho}\,|d_k\rangle)\\
&= \sum_{i}^{\text{database}} \alpha_i \sum_{\substack{\mathbf{x}ss'\\ \mathbf{x}'s''s'''}} \langle\vec{\rho}\,|p_{i,\mathbf{x}'s''s'''}\rangle\,\langle\vec{\rho}\,|p_{k,\mathbf{x}ss'}\rangle\langle\vec{\rho}\,|\kappa_{\mathbf{x}ss'\mathbf{x}'s''s'''}\rangle\\
&= \sum_{\mathbf{x}ss'}\langle\vec{\rho}\,|p_{k,\mathbf{x}ss'}\rangle \sum_{\mathbf{x}'s''s'''}\langle\vec{\rho}\,|\kappa_{\mathbf{x}ss'\mathbf{x}'s''s'''}\rangle \left(\sum_{i}^{\text{database}} \alpha_i\langle\vec{\rho}\,|p_{i,\mathbf{x}'s''s'''}\rangle\right)\\
&= \sum_{\mathbf{x}ss'}\langle\vec{\rho}\,|p_{k,\mathbf{x}ss'}\rangle\langle\vec{\rho}\,|B_{\mathbf{x}ss'}\rangle,
\end{aligned} \tag{A.15}$$

where we have defined

$$\langle\vec{\rho}\,|B_{\mathbf{x}ss'}\rangle = \sum_{\mathbf{x}'s''s'''}\langle\vec{\rho}\,|\kappa_{\mathbf{x}ss'\mathbf{x}'s''s'''}\rangle \left(\sum_{i}^{\text{database}} \alpha_i\langle\vec{\rho}\,|p_{i,\mathbf{x}'s''s'''}\rangle\right). \tag{A.16}$$

Now that the site energy expansion is linear in $|p_{k,\mathbf{x}ss'}\rangle$, we next derive an expansion for the cluster basis functions in this same $|p_{i,\mathbf{x}ss'}\rangle$ basis. But first, a quick side not on the use and meaning of indices.

A.2.1 A Note on Indices

Unfortunately, it is not easy to describe a general atomic environment simply. The set of indices we use to describe the $|p_{i,\mathbf{x}ss'}\rangle$ do have important physical meaning. Here we describe why it is essential to keep the various single, double, and triple primed indices on s and n in the context of maintaining physical invariances (rotational and permutational specifically).

The local atomic environments expanded in $|p_{i,\mathbf{x}ss'}\rangle$ are defined *for each site* in the material's unit cell. Thus, for a cell with multiple atoms in its basis, we have a set of environment expansions. When we dot any two of these $|p_{i,\mathbf{x}ss'}\rangle$ together (which may also be from different materials), we are looking for the similarity between them; it should be 1 only if they are identical. Now, imagine the environment ρ_i is rotated slightly relative to ρ_k. The rotational information about atom i's surroundings is encoded in the n and l indices; thus, we expect changes in the values of the $|p_{i,\mathbf{x}ss'}\rangle$ as the environment rotates.

Now, imagine that atom k shared the *same* indices n and l, instead of having its own n' and l'. Its $|p_{k,\mathbf{x}ss'}\rangle$ values would **not** allow atom i to rotate *independently*; we would lose angular

resolution in the atomic representation and its ability to distinguish between environments would be reduced. By keeping separate indices, we allow the environments to maintain independent angular resolution with respect to each other.

The same logic applies to the use of indices s and s' etc. to track species occupations. If we don't use the full range of primes, it would be possible to permute the atoms in one environment without changing the similarity between it and the one we are comparing it to.

A.3 CLUSTER BASIS FUNCTION EXPANSION IN $|P_{I,\mathbf{X}SS'}\rangle$

The SOAP basis $|p_{i,\mathbf{x}ss'}\rangle$ is continous, differentiable and has all the necessary physical invariances to represent any material. Since we are interested in equating the two bases, we wish to expand the CE basis functions $\Theta_{\mathbf{a}t}$ in the basis of $|p_{i,\mathbf{x}ss'}\rangle$, since these have validity everywhere in space and composition. We will show later that such an expansion leads to a smoothly varying formulation that matches CE *on* the lattice, but also provides a good approximation (extrapolation) for deviations from the ideal lattice positions.

The basis of configurational functions used in CE is defined by symmetrically unique permutations of atomic sites in a crystal[1]. In this derivation we index them by $\mathbf{a}$, which lists the lattice sites defining the cluster. The label $\mathbf{a}$ can thus be directly correlated to the vector $\vec{\rho}$ of the atomic positions in the crystal. In order for the basis to be complete, we also need to adjust each cluster for the number of unique species in the system. For example, in a binary expansion we need at least two functions (called point functions) defined for each lattice site. If the system is ternary we need three. We introduce the label t to define which of these point functions is being used to define the cluster basis function.

Next, we consider the *site* energy ε obtained *via cluster expansion* of unrelaxed crystal structures:

$$\varepsilon(\langle\vec{\sigma}|\rho_i\rangle) = \sum_{\mathbf{a}t} j_{\mathbf{a}t}\langle\vec{\sigma}|\Theta_{i\mathbf{a}t}\rangle. \tag{A.17}$$

where ρ_i is understood to implicitly contain the configuration information $\vec{\sigma}$, and $\langle\vec{\sigma}|\Theta_{i\mathbf{a}t}\rangle$ is the value of cluster function $|\Theta_{\mathbf{a}t}\rangle$ evaluated with its center on atom i in $|\vec{\sigma}\rangle$. The vector space $|\vec{\sigma}\rangle$ is the space of all possible configurations of any material with a regular lattice. For a specific lattice type, say FCC, the number of configurations is countable. The materials space $|\vec{\rho}\rangle$, on the other hand, has an uncountably infinite number of values because it varies smoothly in space. As an example, the material space $|\vec{\rho}\rangle$ can be likened to the real line (which has an uncountably infinite number of values); the set of integers on the real line are a subspace of the reals with a countably infinite set of values. Similarly, the space of all configurations on regular lattices forms a subspace of all possible materials.

By expanding $|\Theta_{i\mathbf{a}t}\rangle$ in $|p_{i,\mathbf{x}ss'}\rangle$, we are using a basis that has validity everywhere in materials space, and we are evaluating it only at points in the subspace. Any points that lie outside of that subspace become interpolations or extrapolations. Returning to the example of the 1D line, it can be likened to interpolating a function over the reals by evaluating it only at integer points, and

[1] By crystal, we mean any material with definite translational and rotational symmetries.

then using a specific basis set (e.g. polynomials) to predict between points. Our interpolating functions are the $|p_{i,\mathbf{x}ss'}\rangle$ and we are evaluating them for specific values in the cluster subspace.

Mathematically then, $|\vec{\sigma}\rangle \in |\vec{\rho}\rangle$, and

$$|\vec{\sigma}\rangle = |\bar{r}_1^{(s_1)}, \bar{r}_2^{(s_2)}, \ldots, \bar{r}_{N_{\text{sites}}}^{(s_{N_{\text{sites}}})}\rangle, \tag{A.18}$$

with $\bar{r}_i$ indicating that the vector position of the atom i is tied to an ideal lattice point.

A.3.1 Motivation for Expandability

Next, we need to motivate the idea that a function of configuration for an entire crystal with n_{species} can be expanded in a set of basis functions $|p_{i,\mathbf{x}ss'}\rangle$ that individually depend only on two species s and s' at a time. The discussion above regarding $|\vec{\sigma}\rangle$ as a subspace of $|\vec{\rho}\rangle$ shows validity in the vector spaces. Our goal here is to show that it also makes sense for *functions* over the composition in those spaces $|\vec{\sigma}\rangle$.

1. Given a specific configuration in $|\vec{\sigma}\rangle$ on the lattice, we can rewrite it in terms of an ordered set of N^2 two-point configuration terms $\{\sigma_{i,j}\}_{i,j=1}^{N}$. Thus, a function of the configuration such as the energy $\langle\vec{\sigma}|E\rangle$ could also, in theory, be written as a many-body function that takes the set of two-point configuration terms as its parameters/inputs.

2. The SOAP power spectrum basis $|p_{i,\mathbf{x}ss'}\rangle$ is evaluated for every combination of s and s' that can be formed with the set of n_{species} unique species. Note also that the $|p_{i,\mathbf{x}ss'}\rangle$ have validity through all materials space $|\vec{\rho}\rangle$.

3. The $\langle\vec{\sigma}|\Theta_{i\mathbf{a}t}\rangle$ cluster functions are first symmetrized by summing over all rotationally equivalent cluster positions.

4. $\langle\vec{\rho}\,|p_{i,\mathbf{x}ss'}\rangle$ are formed by summing index m in the expansion coefficients $\langle\vec{\rho}\,|c_{i,snlm}\rangle$. Because of the form of the spherical harmonics, Y_{lm}, this sum over m produces a spherically symmetric expansion while preserving angular information.[2].

Comparing point 1 to point 2, and point 3 to point 4, we see that functions of the global configuration can formally be expanded in the symmetrized basis of the $|p_{i,ss'\mathbf{x}}\rangle$; thus, our cluster functions $\langle\vec{\sigma}|\Theta_{i\mathbf{a}t}\rangle$ can also be expanded in this basis.

We use the geometric information in the cluster descriptor $\mathbf{a}$ to expand the cluster function vertices (with a Gaussian on each vertex) in the $g_n(r)Y_{lm}(\hat{r})$ basis. We do this once for each unique species s in the *cluster*. Thus, for each set of cluster vertices, we expand only those atomic positions in $|\vec{\sigma}\rangle$ occupied by species s,

$$\langle\vec{\sigma}|\mathbf{a}_{is}t\rangle = \sum_{\vec{r}_j\in\mathbf{a}_s} e^{-(\vec{r}_j-\vec{r})^2/2\sigma_s^2} = \sum_{nlm}\langle\vec{\sigma}|c_{i\mathbf{a}t,snlm}\rangle\, g_n(r)Y_{lm}(\hat{r}), \tag{A.19}$$

[2]Think, for example, of the sum of p-type orbitals p_x, p_y and p_z in the Hydrogen atom; the sum of all three orbitals produces a spherically symmetric shell. It can be shown to be general for all l

with $\mathbf{a}_{is}$ representing the set of vectors pointing from atom i to the vertices specified by the cluster index $\mathbf{a}$ when it is centered on i. Using the $\langle\vec{\sigma}|c_{i\mathbf{a}t,snlm}\rangle$ obtained from this expansion, we can define the cluster power spectrum around site i, $\langle\vec{\sigma}|p_{i\mathbf{a}t,ss'\mathbf{x}}\rangle$, as before:

$$\langle\vec{\sigma}|p_{i\mathbf{a}t,ss'\mathbf{x}}\rangle = \sum_{m=-l}^{l} \langle\vec{\sigma}|c_{i\mathbf{a}t,snlm}\rangle^{\dagger}\langle\vec{\sigma}|c_{i\mathbf{a}t,s'n'lm}\rangle. \tag{A.20}$$

Notice that in all these definitions, the same *geometric* cluster $|\Theta_{i\mathbf{a}t}\rangle$ is evaluated on a specific configuration of the relevant lattice points *for each point function set* t. Hence, we retain the indexing over both $\mathbf{a}$ and t,

$$\langle\vec{\sigma}|\Theta_{i\mathbf{a}t}\rangle = \sum_{\mathbf{x}ss'} \beta_{i\mathbf{a}t,\mathbf{x}ss'}\langle\vec{\sigma}|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle. \tag{A.21}$$

For each cluster function $|\Theta_{i\mathbf{a}t}\rangle$ we have a whole *set of functions* expanded in $|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle$, one for each unique configuration $|\vec{\sigma}\rangle$ that can exist on the lattice. If we want to use a single set of $\beta_{i\mathbf{a}t,\mathbf{x}ss'}$ for all possible configurations, we would need to solve a system of equations that has all possible configurations well represented. Obviously there are a very large number of possible configurations on a lattice (M^N for M species and N lattice points) and it quickly becomes computationally impossible to compute the entire set. Thus, we resolve instead to stochastically sample the configuration space in a way that represents all possible values of $|\vec{\sigma}\rangle$ "well". For example, we could imagine using the Pólya enumeration theorem to count the number of symmetrically unique arrangements on the lattice and weight the stochastic sampling by each concentration's contribution to the total number of unique arrangements for the entire system.

With this expansion, we can rewrite the site energy as

$$\varepsilon(\langle\vec{\sigma}|\rho_i\rangle) = \sum_{\mathbf{a}t} j_{\mathbf{a}t} \sum_{\mathbf{x}ss'} \beta_{i\mathbf{a}t,\mathbf{x}ss'}\langle\vec{\sigma}|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle. \tag{A.22}$$

A.3.2 Calculation of $\beta_{i\mathbf{a}t,\mathbf{x}ss'}$ Coefficients

We now turn to the calculation of the $\beta_{i\mathbf{a}t,\mathbf{x}ss'}$ coefficients. For a given order n of spherical harmonic, the number of basis functions $|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle$ can be quite large. Because of the enormous number of possible configurations $\vec{\sigma}$ on the lattice, it will always be possible to match the number of $\langle\vec{\sigma}|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle$ contractions with data points through the stochastic configuration sampling discussed in the previous section. Thus, the calculation of the $\beta_{i\mathbf{a}t,\mathbf{x}ss'}$ values will come by inversion of a large ($10^4 \times 10^4$) matrix of values. One such linear solution will be calculated *for each cluster function* $|\Theta_{i\mathbf{a}t}\rangle$.

Once a sparse CE has been trained for the unrelaxed alloy system, the number of cluster functions describing the alloy energetics should be relatively small. For each of these, we generate the possible combinations of $\mathbf{a}$ and t indices for each $|\Theta_{i\mathbf{a}t}\rangle$ and evalute the cluster functions stochastically over the set of all possible configurations until we have enough equations to have an invertible linear system. For a cluster with k vertices in a system of M species, there are k^M variations of $|\Theta_{i\mathbf{a}t}\rangle$ to evaluate. A typical CE solution has about $\approx$40 relevant clusters; the

number of each type of cluster (2-body, 3-body, etc.) decreases rapidly with increasing number of vertices. In such a CE, the total number of unique $|\Theta_{i\mathbf{a}t}\rangle$ cluster functions is about 500.

Because of the generality employed up to this point, we could imagine solving this problem for *all* Bravais lattices simultaneously (or specific combinations of Bravais lattices). We would merely add additional equations to the system for each of the subsequent lattices' cluster functions. The set of $\beta_{i\mathbf{a}t,\mathbf{x}ss'}$ found in solving the composite system would represent an interpolation between cluster functions on different underlying lattices. For a subset of similar Bravais lattices, the interpolation may be better than for a solution that interpolates all the Bravais lattices; so, it may still be worthwhile to expand combinations of them separately. The reason we can do this in one, foul swoop is because in all cases we are expanding the $|\Theta_{i\mathbf{a}t}\rangle$ in the *same* basis. However, we should exercise caution in doing so: we need to stochastically sample the configuration space for each Bravais lattice well. As we add additional lattices to the solution, we also need to expand the number of $|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle$ by increasing the order n of the spherical harmonics. Matrix inverses scale at best as $n^{\approx 2.3}$, which sets a hard limit computationally on the number of lattices that can be done concurrently.

One other important point is that once a cluster function has been expanded in $|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle$, those values are valid forever and can be tabulated. This allows any cluster expansion to be transformed into the $|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle$ basis,

$$\varepsilon(\langle\vec{\sigma}|\rho_i\rangle) \;=\; \sum_{\mathbf{x}ss'}\left(\sum_{\mathbf{a}t} j_{\mathbf{a}t}\langle\vec{\sigma}|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle\beta_{i\mathbf{a}t,\mathbf{x}ss'}\right). \tag{A.23}$$

A.4 TYING THE FORMALISMS TOGETHER

In the previous two sections, we described 1) the linearization of the SOAP kernel expansion of total energy in the $|p_{i,\mathbf{x}ss'}\rangle$ basis; 2) the transformation of the $|\Theta_{i\mathbf{a}t}\rangle$ cluster basis functions into the $|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle$ basis. In this section we discuss the possible ways in which these two formalisms can be combined to provide a *species-aware* mathematical description of materials suitable for machine learning.

A.4.1 Determination of $\langle\vec{\rho}\,|\kappa_{\mathbf{x}ss'\mathbf{x}'s''s'''}\rangle$

For a given material in $|\vec{\rho}\,\rangle$, $\langle\vec{\rho}\,|\kappa_{\mathbf{x}ss'\mathbf{x}'s''s'''}\rangle$ is the non-linear, global species overlap term between two atomic environments. Combining Eq. (A.15) with Eq. (A.23), we obtain a connection between the two expansions. The important point here is that the $|p_{i,\mathbf{x}ss'}\rangle$ are a linearly independent set. Thus each term (indexed by $\mathbf{x}$, s and s') in the separate expansions must be equal. This gives

$$\sum_{\mathbf{a}t} j_{\mathbf{a}t}\langle\vec{\sigma}|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle\beta_{i\mathbf{a}t,\mathbf{x}ss'} = \langle\vec{\sigma}|p_{i,\mathbf{x}ss'}\rangle\langle\vec{\sigma}|B_{\mathbf{x}ss'}\rangle. \tag{A.24}$$

where we have switched the "dummy" indices i and k on the right-hand side of Eq. (A.15) to match the CE expansion on the left. Taken as an entire set (i.e. by summing over $\mathbf{a}t$), the $|\Theta_{i\mathbf{a}t}\rangle$, expanded as $|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle$ provide a connection between the subspace $|\vec{\sigma}\,\rangle$ and the global materials

space $|\vec{\rho}\,\rangle$. Inserting the definition of $B_{\mathbf{x}ss'}$, we see the relationship between the expansion coefficients $\beta_{i\mathbf{a}t,\mathbf{x}ss'}$ and the non-linear species overlap coefficients $\langle\vec{\rho}\,|\kappa_{\mathbf{x}ss'\mathbf{x}'s''s'''}\rangle$,

$$\sum_{\mathbf{a}t} j_{\mathbf{a}t}\langle\vec{\sigma}|p_{i\mathbf{a}t,\mathbf{x}ss'}\rangle\beta_{i\mathbf{a}t,\mathbf{x}ss'} = \langle\vec{\sigma}|p_{i,\mathbf{x}ss'}\rangle\sum_{\mathbf{x}'s''s'''}\langle\vec{\rho}\,|\kappa_{\mathbf{x}ss'\mathbf{x}'s''s'''}\rangle\left(\sum_{k}^{\text{database}}\alpha_k\langle\vec{\rho}\,|p_{k,\mathbf{x}'s''s'''}\rangle\right). \quad \text{(A.25)}$$

For each atomic environment i and tuple $\mathbf{x}ss'$, we can generate the LHS by cluster expanding the site energy and then transforming it to $|p_{i\mathbf{a}t,\mathbf{x}ss'}\,\rangle$. This represents the "pinning down" of the basis to specific points in the subspace $|\vec{\sigma}\,\rangle$ for which CE is defined.

Similarly, we can perform Gaussian Process Regression as described earlier on a database of local atomic environments to produce the RHS of Eq. (A.25). This pins the global $|p_{i,\mathbf{x}ss'}\,\rangle$ basis to the pure elemental materials for which the current SOAP implementation is valid.

As we accumulate additional environments for each i and each tuple $\mathbf{x}ss'$, we form a system of equations that can be solved or approximated using standard numerical methods.

A.4.2 Determination of $\langle\vec{\rho}\,|\varsigma_{ss'nlmm'}\rangle$

Another alternative is to access the linear species overlap functions between only *two* species s and s' directly. This reduces the problem to finding the values of $\langle\vec{\rho}\,|\varsigma_{ss'nlmm'}\rangle$. Because of the inherent non-linearity in $c_{i,snlm}$, it is not possible to access $\langle\vec{\rho}\,|\varsigma_{ss'nlmm'}\rangle$ by equating CE and SOAP on total energy. Instead, we choose to equate the CE and SOAP expansions on electrostatic energy only. In doing so, we assume that the overlap $\langle\vec{\sigma}|\varsigma_{ss'nlmm'}\rangle$ derived from a CE for only electrostatic energy will be a good representative *even for quantum contributions to the material properties*. If we confine ourselves to energetic calculations, this is not far-fetched (since the quantum contributions are also directionally dependent). As long as the other properties are extracted from higher-order methods based on the forces or potential energies, the assumption should be satisfactory.

First, we derive an expression for the electrostatic energy $\langle\vec{\rho}\,|\tilde{E}\rangle$ that is linear in the charge density. Formally, if we restrict ourselves to a finite volume, this energy (for an individual atomic site with cutoff radius r_{cut}) is defined as:

$$\begin{aligned}\tilde{\varepsilon}(\langle\vec{\rho}\,|\rho_i\rangle) &= \frac{1}{2}\int_0^{r_{\text{cut}}}\int_0^{\pi}\int_0^{2\pi}\langle\vec{\rho}\,|\rho_i(\vec{r})\rangle\langle\vec{\rho}\,|\phi_i(\vec{r})\rangle d^3r \quad \text{(A.26)}\\ &= \frac{1}{2}\int_0^{r_{\text{cut}}}\int_0^{\pi}\int_0^{2\pi}\sum_{nlm}\langle\vec{\rho}\,|c_{i,snlm}\rangle\, g_n(r)Y_{lm}(\hat{r})\langle\vec{\rho}\,|\phi_i(\vec{r})\rangle d^3r.\end{aligned}$$

The trick whereby we avoid the non-linearity in ρ_i is to take the electrostatic potential ϕ_i as given. Since DFT codes have the ability to output a representation of ϕ_i (in a different basis than $|p_{i,\mathbf{x}ss'}\,\rangle$), we can take it as a known function and just expand the charge density as shown in Eq. (A.26). In doing so, we use the spherically averaged electrostatic potential; without this approximation the integrals in the derivation below would be a nightmare.[3]

[3]Software already exists for extracting the spherically averaged potential $\phi(r)$, within a sphere of finite volume using the LOCPOT file in VASP. See "Electronic Chemical Potentials of Porous MetalâĂŞOrganic Frameworks", Keith T. Butler, Christopher H. Hendon, and Aron Walsh; J. Am. Chem. Soc., 2014, 136 (7), pp 2703âĂŞ2706. See also https://github.com/WMD-group/MacroDensity

Following the same derivation in Eq. (A.19), we expand the $|\Theta_{i\mathbf{a}t}\rangle$, but this time we restrict ourselves to the coefficients $c_{i\mathbf{a}t,snlm}$ instead of continuing to the power spectrum. Then,

$$\tilde{\varepsilon}(\langle\vec{\sigma}|\rho_i\rangle) = \sum_{\mathbf{a}t} j_{\mathbf{a}t} \sum_{snlm} \langle\vec{\sigma}|c_{i\mathbf{a}t,snlm}\rangle\, g_n(r) Y_{lm}(\hat{r}) \tag{A.27}$$

is the cluster expanded electrostatic *site* energy in $g_n(r)Y_{lm}(\hat{r})$. We can now equate these two expressions, once again using the orthogonality of g_n and Y_{lm} and re-arranging the order of summation and integration in Eq. (A.26).

$$\begin{aligned}\tilde{\varepsilon}(\langle\vec{\sigma}|\rho_i\rangle) &= \sum_{\mathbf{a}t} j_{\mathbf{a}t} \langle\vec{\sigma}|c_{i\mathbf{a}t,snlm}\rangle\, g_n(r) Y_{lm}(\hat{r}) \\ &= \frac{1}{2}\int_0^{r_{\text{cut}}}\int_0^{\pi}\int_0^{2\pi} \langle\vec{\rho}\,|c_{i,snlm}\rangle\, g_n(r) Y_{lm}(\hat{r}) \langle\vec{\rho}\,|\tilde{\phi}_i(\vec{r})\rangle d^3r,\end{aligned} \tag{A.28}$$

$\tilde{\phi}_i$ being the spherically averaged potential within the cutoff sphere around atom i, which varies only with r. We now exploit the orthogonality of our g_nY_{lm} basis,

$$\int_0^{2\pi}\int_0^{\pi} Y^*_{\lambda\mu}(\hat{r}) Y_{lm}(\hat{r}) \sin\theta d\theta d\phi = \delta_{l\lambda}\delta_{m\mu}, \tag{A.29}$$

first for the cluster expanded energy.

$$\begin{aligned}4\pi\int_0^{\infty} \tilde{\varepsilon}^2(\langle\vec{\sigma}|\rho_i\rangle) e^{-r} d^3r &= \int_0^{\infty} \sum_{\mathbf{a}t} j^*_{\mathbf{a}t}\, j_{\mathbf{a}t} \langle\vec{\sigma}|c_{i\mathbf{a}t,snlm}\rangle^{\dagger} \langle\vec{\sigma}|c_{i\mathbf{a}t,snlm}\rangle\, g^*_n(r) g_n(r) e^{-r} r^2 dr \\ &= \left(\sum_{\mathbf{a}t}\sum_{\eta\tau}\sum_{ss'} j^*_{\mathbf{a}t}\, j_{\eta\tau} \langle\vec{\sigma}|c_{i\mathbf{a}t,snlm}\rangle^{\dagger} \langle\vec{\sigma}|c_{i\eta\tau,s'nlm}\rangle\right) \int_0^{\infty} g^*_n(r) g_n(r) e^{-r} r^2 dr \\ &= \sum_{ss'} \langle\vec{\sigma}|\zeta_{i,snlm}\rangle^{\dagger} \langle\vec{\sigma}|\zeta_{i,s'nlm}\rangle \int_0^{\infty} g^*_n(r) g_n(r) e^{-r} r^2 dr,\end{aligned} \tag{A.30}$$

where we have added e^{-r} so that the integral over the constant value $\tilde{\varepsilon}^2$ converges, and defined

$$\langle\vec{\sigma}|\zeta_{i,snlm}\rangle = \sum_{\mathbf{a}t} j_{\mathbf{a}t} \langle\vec{\sigma}|c_{i\mathbf{a}t,snlm}\rangle. \tag{A.31}$$

And again for the SOAP expanded electrostatic energy:

$$\begin{aligned}\int_0^{\infty} \tilde{\varepsilon}^2(\langle\vec{\rho}\,|\rho_i\rangle) e^{-r} dr &= \frac{1}{4}\sum_{ss'} \langle\vec{\rho}\,|\varsigma_{ss'nlm}\rangle \langle\vec{\rho}\,|c_{i,snlm}\rangle^{\dagger} \langle\vec{\rho}\,|c_{i,s'nlm}\rangle \int_0^{\infty} g^*_n(r) g_n(r) \tilde{\phi}^2_{r_{\text{cut}}}(r) e^{-r} r^2 d^3r \\ &= \sum_{ss'} \langle\vec{\rho}\,|\varsigma_{ss'nlm}\rangle \langle\vec{\rho}\,|c_{i,snlm}\rangle^{\dagger} \langle\vec{\rho}\,|c_{i,s'nlm}\rangle\, \bar{\phi}^2,\end{aligned} \tag{A.32}$$

where we define the integral

$$\bar{\phi^2} = \frac{1}{4}\int_0^{\infty} g^*_n(r) g_n(r) \tilde{\phi}^2_{r_{\text{cut}}}(r) e^{-r} r^2 dr \tag{A.33}$$

for clarity. Since $\bar{\phi}^2$ is strictly positive, there won't be any unfortunate cancellation in potential-neutral environments to interfere with the coefficients. We can now equate the CE version of the electrostatic energy with the one we derived from the charge density,

$$\langle\vec{\sigma}|\zeta_{i,snlm}\rangle^{\dagger}\langle\vec{\sigma}|\zeta_{i,s'nlm}\rangle\int_0^{\infty} g_n^*(r)g_n(r)e^{-r}r^2dr = \langle\vec{\rho}\,|\varsigma_{ss'nlm}\rangle\langle\vec{\rho}\,|c_{i,snlm}\rangle^{\dagger}\langle\vec{\rho}\,|c_{i,s'nlm}\rangle\,\bar{\phi}^2. \quad \text{(A.34)}$$

This can be simplified to obtain the value of $\langle\vec{\rho}\,|\varsigma_{ss'nlm}\rangle$,

$$\langle\vec{\rho}\,|\varsigma_{ss'nlm}\rangle = \frac{\langle\vec{\sigma}|\zeta_{i,snlm}\rangle^{\dagger}\langle\vec{\sigma}|\zeta_{i,s'nlm}\rangle}{\langle\vec{\rho}\,|c_{i,snlm}\rangle^{\dagger}\langle\vec{\rho}\,|c_{i,s'nlm}\rangle}\frac{\int_0^{\infty} g_n^*(r)g_n(r)e^{-r}r^2dr}{\int_0^{\infty} g_n^*(r)g_n(r)\tilde{\phi}^2_{r_{\text{cut}}}(r)e^{-r}r^2dr} \quad \text{(A.35)}$$

As long as the integrals converge and the value for $\bar{\phi}^2 \neq 0$, the value will be defined. A good course of action would be to calculate the global, non-linear overlap coefficients $\langle\vec{\rho}\,|\kappa_{\mathbf{x}ss'\mathbf{x}'s''s'''}\rangle$ and the local, linear coefficients $\langle\vec{\rho}\,|\varsigma_{ss'nlm}\rangle$ and compare them to gain possible physical insights.

A.5 OFF-LATTICE CLUSTER EXPANSION

Before we embark on the complicated journey described in the previous section, we would like to get a feel for how easily the cluster functions can be transformed into the $|p_{i\mathbf{a}\boldsymbol{t},\mathbf{x}ss'}\rangle$ basis.

As a first approximation to an off-lattice CE, we use a regular CE trained on a group of systems in which only the lattice parameter is allowed to relax. All internal degrees of freedom remain fixed so that the CE basis is complete. In that instance, we have an expansion similar to Eq. (A.8) for each structure in $|\vec{\sigma}\rangle$,

$$\begin{aligned}\langle\vec{\sigma}|\rho_i(\vec{r})\rangle &= \sum_{snlm}\langle\vec{\rho}\,|c_{i,snlm}\rangle\, g_n(r)Y_{lm}(\hat{r}) \\ &= \sum_{\mathbf{x}ss'}\langle\vec{\sigma}|p_{i,\mathbf{x}ss'}\rangle. \end{aligned} \quad \text{(A.36)}$$

This allows us to expand the CE energy as

$$\varepsilon(\langle\vec{\sigma}|\rho_i\rangle) = \sum_{\mathbf{x}ss'}\langle\vec{\sigma}|p_{i,\mathbf{x}ss'}\rangle\chi_{i,\mathbf{x}ss'}, \quad \text{(A.37)}$$

where the $\chi_{\mathbf{x}ss'}$ are the expansion parameters. Since the number of structures and $|p_{i,\mathbf{x}ss'}\rangle$ basis functions may not always be equal, we imagine a rectangular matrix $\mathbf{P}_i = \langle\vec{\sigma}|p_{i,\mathbf{x}ss'}\rangle$ and compute its pseudo-inverse using Singular Value Decomposition (SVD).[4] Using the pseudo-inverse, we can obtain the $\chi_{i,\mathbf{x}ss'}$ values using $\vec{\chi}_i = \mathbf{V}\mathbf{S}^{+}\mathbf{U}^{*}\varepsilon(\langle\vec{\sigma}|\rho_i\rangle)$ where $\text{SVD}(\mathbf{P}_i) = \mathbf{U}\mathbf{S}\mathbf{V}^{*}$ and $\mathbf{S}^{+}$ is the pseudoinverse of matrix $\mathbf{S}$.

Once the value for $\chi_{i,\mathbf{x}ss'}$ has been calculated from the *on-lattice* cluster expansion energies and structures, it can be treated as constant. Then for a new structure (including one with small relaxations from the ideal lattice positions), we can calculate a new expansion in $|p_{i,\mathbf{x}ss'}\rangle$ and use

[4]There are many solvers for underdetermined linear systems based on various norms (e.g. ℓ_1 norm solvers with compressing sensing).

the on-lattice $\vec{\chi}_i$ to approximate the off-lattice energy. For small deviations we expect the energies to be in good agreement.

To gain intuition for the range of applicability, we return to the idea of pinning the full materials space basis functions using values calculated in a subspace. The $\chi_{i,\mathbf{x}ss'}$ values calculated allow the $|p_{i,\mathbf{x}ss'}\rangle$ basis functions to interpolate between the known values produced by the CE. When we move off-lattice, the predictions will be in good agreement until we "fall off the manifold" for which the interpolations are defined. Inasmuch as the material space is high-dimensional and complicated, it is difficult to quantify the range of applicability without numerical experiments.

This first attempt at an off-lattic CE is a useful starting point to prove the applicability of the derivation up to this point. If we can expand the on-lattice CE values in the $|p_{i,\mathbf{x}ss'}\rangle$ basis with high accuracy, it gives us confidence to pursue the more involved calculation of the $\beta_{i\mathbf{a}t,ss'\mathbf{x}}$ or $\langle\vec{p}\,|\varsigma_{ss'nl}\rangle$ described earlier.

A.5.1 Cluster Expansion Transformation to $|p_{i,\mathbf{x}ss'}\rangle$

Using the method described, we set out to see if the $|p_{i,\mathbf{x}ss'}\rangle$ basis could reproduce CE results for *on-lattice* structures. Since those structures are on the manifold of the training data, they should be reproducible without much difficulty *if the* $|p_{i,\mathbf{x}ss'}\rangle$ *can accurately represent the cluster functions*. Keep in mind that here we are not expanding *individual* cluster functions, but rather summing over all the cluster functions (index $\mathbf{a}t$) from the CE and summing over all the $|p_{i,\mathbf{x}ss'}\rangle$ (index $\mathbf{x}ss'$), which merely shows that when taken as a *set*, the two approaches are equivalent.

Using a CE of Cu-Pt with 41 terms, we performed the pseudoinverse described below Eq. (A.37). We trained on 3710 data points from both unrelaxed and relaxed systems, leaving 850 points for validation. RMS error on the test set was 4.5 meV/atom. Considering that the error on the CE itself was on the order of 7.1 meV/atom, this is suitable proof that the CE basis functions, at least *when taken together for the whole model*, can be expanded in the pixies basis.

We also expanded the individual contributions of each cluster basis function in the pixies basis and performed the pseudoinverse to predict on-lattice cluster values within the SOAP basis. Across the 3-body and higher cluster functions, we obtained an average RMS error of 0.15 meV/atom on the test sets. This shows that the cluster basis functions can also be expanded individually and that the mathematical formalism is sufficient within the assumptions and constraints that we have specified.

Bibliography

[1] K.-W. Ang, K.-J. Chui, V. Bliznetsov, C.-H. Tung, A. Du, N. Balasubramanian, G. Samudra, M. F. Li, and Y.-C. Yeo, "Lattice strain analysis of transistor structures with silicon-germanium and silicon-carbon source/drain stressors," Applied Physics Letters **86,** 093102 (2005).

[2] F. Faggin and T. Klein, "Silicon gate technology," Solid-State Electronics **13,** 1125 – 1144 (1970).

[3] A. Andreini, C. Contiero, and P. Galbiati, "A new integrated silicon gate technology combining bipolar linear, CMOS logic, and DMOS power parts," IEEE Transactions on Electron Devices **33,** 2025–2030 (1986).

[4] L. L. Vadasz, A. S. Grove, T. A. Rowe, and G. E. Moore, "Silicon-gate technology," IEEE Spectrum **6,** 28–35 (1969).

[5] W. Kohn and L. J. Sham, "Self-Consistent Equations Including Exchange and Correlation Effects," Phys. Rev. **140,** A1133–A1138 (1965).

[6] P. Hohenberg and W. Kohn, "Inhomogeneous Electron Gas," Phys. Rev. **136,** B864–B871 (1964).

[7] J. Hong *et al.*, "Exploring atomic defects in molybdenum disulphide monolayers," **6,** 6293 EP .

[8] K. Burke, "Perspective on density functional theory," The Journal of Chemical Physics **136,** 150901 (2012).

[9] D. Miracle and O. Senkov, "A critical review of high entropy alloys and related concepts," Acta Materialia **122,** 448 – 511 (2017).

[10] C. T. Koch, Ph.D. thesis, Arizona State University, 2002.

[11] A. K. Fletcher and S. Rangan, "Inference in Deep Networks in High Dimensions," CoRR abs/1706.06549 (2017).

[12] L. Gao, J. Song, X. Liu, J. Shao, J. Liu, and J. Shao, "Learning in High-Dimensional Multimedia Data: The State of the Art," CoRR abs/1707.02683 (2017).

[13] C. Beck, W. E, and A. Jentzen, "Machine learning approximation algorithms for high-dimensional fully nonlinear partial differential equations and second-order backward stochastic differential equations," ArXiv e-prints (2017).

[14] X. Zhu, C. Vondrick, C. C. Fowlkes, and D. Ramanan, "Do We Need More Training Data?," CoRR abs/1503.01508 (2015).

[15] A. Halevy, P. Norvig, and F. Pereira, "The Unreasonable Effectiveness of Data," IEEE Intelligent Systems **24,** 8–12 (2009).

[16] S. F. Dodge and L. J. Karam, "A Study and Comparison of Human and Deep Learning Recognition Performance Under Visual Distortions," CoRR abs/1705.02498 (2017).

[17] R. Geirhos, D. H. J. Janssen, H. H. Schütt, J. Rauber, M. Bethge, and F. A. Wichmann, "Comparing deep neural networks against humans: object recognition when the signal gets weaker," CoRR abs/1706.06969 (2017).

[18] X. Zhao, Y. Wu, G. Song, Z. Li, Y. Zhang, and Y. Fan, "A deep learning model integrating FCNNs and CRFs for brain tumor segmentation," CoRR abs/1702.04528 (2017).

[19] S. Liu, H. Zheng, Y. Feng, and W. Li, "Prostate Cancer Diagnosis using Deep Learning with 3D Multiparametric MRI," CoRR abs/1703.04078 (2017).

[20] Y. Liu *et al.*, "Detecting Cancer Metastases on Gigapixel Pathology Images," CoRR abs/1703.02442 (2017).

[21] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Achieving Human Parity in Conversational Speech Recognition," CoRR abs/1610.05256 (2016).

[22] B. Athiwaratkun and K. Kang, "Feature Representation in Convolutional Neural Networks," CoRR abs/1507.02313 (2015).

[23] H. F. Guo, Y. L. Gong, and J. X. Guo, "Feature Extraction Based on the Multiple Harmonic Analysis of 2-Dimensional Pattern of Partial Discharge," In *Sensors, Measurement and Intelligent Materials*, Applied Mechanics and Materials **303,** 478–481 (Trans Tech Publications, 2013).

[24] H. Erişti, A. Uçar, and Y. Demir, "Wavelet-based feature extraction and selection for classification of power system disturbances using support vector machines," Electric Power Systems Research **80,** 743 – 752 (2010).

[25] M. Uyar, S. Yildirim, and M. T. Gencoglu, "An effective wavelet-based feature extraction method for classification of power quality disturbance signals," Electric Power Systems Research **78,** 1747 – 1755 (2008).

[26] J. L. Terry, A. Crampton, and C. J. Talbot, "Passive sonar harmonic detection using feature extraction and clustering analysis," In *Proceedings of OCEANS 2005 MTS/IEEE*, pp. 2760–2766 Vol. 3 (2005).

[27] S. K. Dmitry Storcheus, Afshin Rostamizadeh, "A Survey of Modern Questions and Challenges in Feature Extraction," JMLR: Workshop and Conference Proceedings **44,** 1–18 (2015).

[28] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, "Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning," Phys. Rev. Lett. **108,** 058301 (2012).

[29] J. Behler and M. Parrinello, "Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces," Phys. Rev. Lett. **98,** 146401 (2007).

[30] J. P. Janet and H. J. Kulik, "Predicting electronic structure properties of transition metal complexes with neural networks," Chem. Sci. **8,** 5137–5152 (2017).

[31] B. Huang and O. A. von Lilienfeld, "Communication: Understanding molecular representations in machine learning: The role of uniqueness and target similarity," The Journal of Chemical Physics **145,** 161102 (2016).

[32] K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O. A. von Lilienfeld, A. Tkatchenko, and K.-R. Müller, "Assessment and Validation of Machine Learning Methods for Predicting Molecular Atomization Energies," Journal of Chemical Theory and Computation **9,** 3404–3419 (2013), pMID: 26584096.

[33] K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O. A. von Lilienfeld, A. Tkatchenko, and K.-R. Müller, "Assessment and Validation of Machine Learning Methods for Predicting Molecular Atomization Energies," Journal of Chemical Theory and Computation **9,** 3404–3419 (2013).

[34] K. Toyoura, D. Hirano, A. Seko, M. Shiga, A. Kuwabara, M. Karasuyama, K. Shitara, and I. Takeuchi, "Machine-learning-based selective sampling procedure for identifying the low-energy region in a potential energy surface: A case study on proton conduction in oxides," Phys. Rev. B **93,** 054112 (2016).

[35] A. P. Bartók, R. Kondor, and G. Csányi, "On representing chemical environments," Phys. Rev. B **87,** 184115 (2013).

[36] A. P. Bartok, S. De, C. Poelking, N. Bernstein, J. Kermode, G. Csanyi, and M. Ceriotti, "Machine Learning Unifies the Modelling of Materials and Molecules," ArXiv e-prints (2017).

[37] R. Dingreville, R. A. Karnesky, G. Puel, and J.-H. Schmitt, "Review of the synergies between computational modeling and experimental characterization of materials across length scales," Journal of Materials Science **51,** 1178–1203 (2015).

[38] Q. Gao, S. Yao, J. Schneider, and M. Widom, "Machine Learning methods for interatomic potentials: application to boron carbide," ArXiv e-prints (2015).

[39] A. V. Shapeev, "Moment Tensor Potentials," Multiscale Model. Simul. **14,** 1153–1173 (2016).

[40] A. Shapeev, "Accurate representation of formation energies of crystalline alloys with many components," Computational Materials Science **139,** 26 – 30 (2017).

[41] S. Biswas and S. Narasimhan, "A simple descriptor and predictor for the stable structures of two-dimensional surface alloys," ArXiv e-prints (2017).

[42] H. Huo and M. Rupp, "Unified Representation for Machine Learning of Molecules and Crystals," ArXiv e-prints (2017).

[43] J. M. Sanchez, F. Ducastelle, and D. Gratias, "Generalized cluster description of multicomponent systems," Physica A: Statistical and Theoretical Physics **128,** 334–350 (1984).

[44] J. M. Sanchez, "Cluster expansion and the configurational theory of alloys," Physical Review B **81,** 224202 (2010).

[45] A. van de Walle, "Methods for First-Principles Alloy Thermodynamics," JOM **65,** 1523–1532 (2013).

[46] D. Lerch, O. Wieckhorst, G. L. W. Hart, R. W. Forcade, and S. Müller, "UNCLE: a code for constructing cluster expansions for arbitrary lattices with minimal user-input," Modelling and Simulation in Materials Science and Engineering **17,** 055003 (2009).

[47] R. Tanaka, K. Takeuchi, and K. Yuge, "Application of Grid Increment Cluster Expansion to Modeling Potential Energy Surface of Cu-Based Alloys," Materials Transactions **56,** 1077–1080 (2015).

[48] W. Chen, G. Xu, I. Martin-Bragado, and Y. Cui, "Non-empirical phase equilibria in the Cr-Mo system: A combination of first-principles calculations, cluster expansion and Monte Carlo simulations," Solid State Sciences **41,** 19–24 (2015).

[49] X. Zhang and M. H. F. Sluiter, "Cluster Expansions for Thermodynamics and Kinetics of Multicomponent Alloys," Journal of Phase Equilibria and Diffusion (2015).

[50] A. Seko, K. Shitara, and I. Tanaka, "Efficient determination of alloy ground-state structures," Physical Review B **90,** 174104 (2014).

[51] B. Alling, a. V. Ruban, A. Karimi, L. Hultman, and I. a. Abrikosov, "Unified cluster expansion method applied to the configurational thermodynamics of cubic Ti1-xAlxN," Physical Review B - Condensed Matter and Materials Physics **83,** 1–8 (2011).

[52] I. Tanaka, A. Seko, A. Togo, Y. Koyama, and F. Oba, "Phase relationships and structures of inorganic crystals by a combination of the cluster expansion method and first principles calculations.," Journal of physics. Condensed matter : an Institute of Physics journal **22,** 384207 (2010).

[53] M. Asato, H. Takahashi, T. Inagaki, N. Fujima, R. Tamura, and T. Hoshino, "Cluster Expansion Approach for Relative Stability among Different Atomic Structures in Alloys: an Approach from a Dilute Limit," MATERIALS TRANSACTIONS **48,** 1711–1716 (2007).

[54] S. V. Barabash, V. Blum, S. Müller, and A. Zunger, "Prediction of unusual stable ordered structures of Au-Pd alloys via a first-principles cluster expansion," Physical Review B **74,** 035108 (2006).

[55] A. van de Walle and G. Ceder, "Automating first-principles phase diagram calculations," Journal of Phase Equilibria **23,** 348–359 (2002).

[56] D. B. Laks, L. G. Ferreira, S. Froyen, and A. Zunger, "Efficient cluster expansion for substitutional systems," Physical Review B **46,** 12587–12605 (1992).

[57] A. Seko, Y. Koyama, and I. Tanaka, "Cluster expansion method for multicomponent systems based on optimal selection of structures for density-functional theory calculations," Physical Review B **80,** 165122 (2009).

[58] A. H. Nguyen, C. W. Rosenbrock, C. S. Reese, and G. L. W. Hart, "Robustness of the cluster expansion: Assessing the roles of relaxation and numerical error," Phys. Rev. B **96,** 014107 (2017).

[59] G. L. W. Hart and R. W. Forcade, "Generating derivative structures from multilattices: Algorithm and application to hcp alloys," Physical Review B - Condensed Matter and Materials Physics **80,** 1–8 (2009).

[60] C. W. Rosenbrock, W. S. Morgan, G. L. W. Hart, S. Curtarolo, and R. W. Forcade, "Numerical Algorithm for Pólya Enumeration Theorem," J. Exp. Algorithmics **21,** 1.11:1–1.11:17 (2016).

[61] L. J. Nelson, G. L. W. Hart, F. Zhou, and V. Ozoliņš, "Compressive sensing as a paradigm for building physics models," Physical Review B **87,** 035125 (2013).

[62] L. J. Nelson, V. Ozoliņš, C. S. Reese, F. Zhou, and G. L. W. Hart, "Cluster expansion made easy with Bayesian compressive sensing," Physical Review B **88,** 155105 (2013).

[63] B. J. Campbell, H. T. Stokes, D. E. Tanner, and D. M. Hatch, "*ISODISPLACE*: a web-based tool for exploring structural distortions," Journal of Applied Crystallography **39,** 607–614 (2006).

[64] L. B. Pártay, A. P. Barták, and G. Csányi, "Efficient Sampling of Atomic Configurational Spaces," The Journal of Physical Chemistry B **114,** 10502–10512 (2010), pMID: 20701382.

[65] L. B. Pártay, A. P. Bartók, and G. Csányi, "Nested sampling for materials: The case of hard spheres," Phys. Rev. E **89,** 022302 (2014).

[66] R. J. N. Baldock, L. B. Pártay, A. P. Bartók, M. C. Payne, and G. Csányi, "Determining pressure-temperature phase diagrams of materials," Phys. Rev. B **93,** 174108 (2016).

[67] W. Z. Han, M. J. Demkowicz, E. G. Fu, Y. Q. Wang, and A. Misra, "Effect of grain boundary character on sink efficiency," Acta Materialia **60,** 6341–6351 (2012).

[68] A. King, G. Johnson, D. Engelberg, W. Ludwig, and J. Marrow, "Observations of intergranular stress corrosion cracking in a grain-mapped polycrystal," Science **321,** 382–385 (2008).

[69] C. W. Rosenbrock, E. R. Homer, G. Csányi, and G. L. W. Hart, "Discovering the building blocks of atomic systems using machine learning: application to grain boundaries," npj Computational Materials **3,** 747 (2017).

[70] L. Huber, B. Grabowski, M. Militzer, J. Neugebauer, and J. Rottler, "Ab initio modelling of solute segregation energies to a general grain boundary," Acta Materialia **132,** 138 – 148 (2017).

[71] H. Lambert, A. Fekete, J. R. Kermode, and A. De Vita, "Imeall: A Computational Framework for the Calculation of the Atomistic Properties of Grain Boundaries," ArXiv e-prints (2017).

[72] M. Hirn, N. Poilvert, and S. Mallat, "Quantum Energy Regression using Scattering Transforms," CoRR abs/1502.02077 (2015).

[73] M. Hirn, S. Mallat, and N. Poilvert, "Wavelet Scattering Regression of Quantum Chemical Energies," Multiscale Modeling & Simulation **15,** 827–863 (2017).

[74] S. Curtarolo *et al.*, "AFLOW: An automatic framework for high-throughput materials discovery," Computational Materials Science **58,** 218 – 226 (2012).

[75] S. Curtarolo *et al.*, "AFLOWLIB.ORG: A distributed materials properties repository from high-throughput ab initio calculations," Computational Materials Science **58,** 227 – 235 (2012).

[76] F. Rose, C. Toher, E. Gossett, C. Oses, M. B. Nardelli, M. Fornari, and S. Curtarolo, "AFLUX: The LUX materials search API for the AFLOW data repositories," Computational Materials Science **137,** 362 – 370 (2017).

[77] R. H. Taylor, F. Rose, C. Toher, O. Levy, K. Yang, M. B. Nardelli, and S. Curtarolo, "A RESTful API for exchanging materials data in the AFLOWLIB.org consortium," Computational Materials Science **93,** 178 – 192 (2014).

[78] S. van der Walt, S. C. Colbert, and G. Varoquaux, "The NumPy Array: A Structure for Efficient Numerical Computation," Computing in Science Engineering **13,** 22–30 (2011).

[79] C. W. Rosenbrock, "A Practical Python API for Querying AFLOWLIB," CoRR abs/1710.00813 (2017).

[80] F. Zhou, W. Nielson, Y. Xia, and V. Ozoliņš, "Lattice Anharmonicity and Thermal Conductivity from Compressive Sensing of First-Principles Calculations," Phys. Rev. Lett. **113,** 185501 (2014).

[81] S. De, A. P. Bartok, G. Csanyi, and M. Ceriotti, "Comparing molecules and solids across structural and alchemical space," Phys. Chem. Chem. Phys. **18,** 13754–13769 (2016).

[82] W. J. Szlachta, A. P. Bartók, and G. Csányi, "Accuracy and transferability of Gaussian approximation potential models for tungsten," Physical Review B **90,** 104108 (2014).