2008-06-26

Algorithm for Generating Derivative Structures

Gus L. W. Hart
gus.hart@gmail.com

Rodney W. Forcade

Original Publication Citation

Gus L. W. Hart and Rodney W. Forcade, "Algorithm for generating derivative structures," Phys. Rev. B 77 224115 (26 June 28). The original article may be found here: http://prb.aps.org/abstract/PRB/v77/i22/e224115

BYU ScholarsArchive Citation

Hart, Gus L. W. and Forcade, Rodney W., "Algorithm for Generating Derivative Structures" (2008). *Faculty Publications*. 180.
https://scholarsarchive.byu.edu/facpub/180

Algorithm for generating derivative structures

Gus L. W. Hart[1] and Rodney W. Forcade[2]
[1]*Department of Physics and Astronomy, Brigham Young University, Provo, Utah 84602, USA*
[2]*Department of Mathematics, Brigham Young University, Provo, Utah 84602, USA*

(Received 7 March 2008; published 26 June 2008)

We present an algorithm for generating all derivative superstructures—for arbitrary parent structures and for any number of atom types. This algorithm enumerates superlattices and atomic configurations in a geometry-independent way. The key concept is to use the quotient group associated with each superlattice to determine all unique atomic configurations. The run time of the algorithm scales *linearly* with the number of unique structures found.

DOI: 10.1103/PhysRevB.77.224115 PACS number(s): 61.50.Ah, 61.66.Dk, 61.90.+d, 61.50.Nw

I. WHY DERIVATIVE STRUCTURES?

Derivative superstructures[1] play an important role in different material phenomena such as chemical ordering in alloys, spin ordering in magnets, and vacancy ordering in nonstoichiometric materials. Similarly, derivative *superlattices*[2–4] are important in problems such as twinning. What is a derivative superstructure? A derivative superstructure is a structure whose lattice vectors are multiples of those of a "parent lattice" and whose atomic basis vectors correspond to *lattice points* of the parent lattice. Many structures of intermetallic compounds can be classified as fcc-derived superstructures; an example is shown in Fig. 1. These superstructures have atomic sites that closely correspond to the sites of an fcc lattice but some of the translational symmetry is broken by a periodic arrangement of different kinds of atoms. The structures shown in Fig. 2 comprise the set of all fcc-derived binary superstructures with unit cell sizes of two, three, and four times larger than the parent lattice.

Large sets of derivative superstructures are often used in (practically) exhaustive searches of binary configurations on a lattice to determine ground state properties of intermetallic systems. The approach is not limited to searches of configurational energies, but other physical observables can also be targeted if an appropriate Hamiltonian is available. For example, Kim *et al.*[5] used an empirical pseudopotential Hamiltonian and a large list of derivative superstructures to directly search semiconductor alloys for desirable band gaps and effective masses. The set of derivative superstructures is useful in any situation where the physical observable of interest depends on the atomic configuration.

For the aforementioned reasons, an algorithm for systematically generating all superstructures of a given parent structure is useful. Such an algorithm has been presented in the literature only once[6,7] by Ferreira, Wei, and Zunger (FWZ). The FWZ algorithm generates superlattices using a geometric, "smallest first" approach[8] (see page 44 of Ref. 6) and then generates all unique atomic configurations. Many of the resulting structures are equivalent by rotational and/or translational symmetry. FWZ removes duplicates by calculating interatom correlations (pairs, triplets, etc.) averaged over the structure, eliminating structures with matching correlations. Because of the limited range of correlations used in the original FWZ implementation, some formally inequivalent structures are eliminated, though in practice the list may be sufficient.

Though the original presentation of the FWZ algorithm is restricted to fcc- and bcc-based superstructures and to binary cases only, the superlattice generation can be extended to the general case and it was implemented in several alloy modeling packages. The code of Ref. 9 overcomes the deficiencies of finding duplicates via correlations by using a rigorously complete list of correlations.[10] The ATAT package[11,12] handles the issue by directly comparing the structures geometrically.

The purpose of this paper is to present a general algorithm that generates a formally complete list of two- or three-dimensional superstructures, and that works for any parent lattice and for arbitrary *k*-nary systems (binary, ternary, etc.). This algorithm is conceptually distinct from FWZ and related implementations. Instead of using a geometrical approach, it takes advantage of known group-theoretical properties of integer matrices. The algorithm is orders of magnitude faster than FWZ, more general, and formally complete. A FORTRAN95 implementation of the algorithm is included in this paper as supplementary material.[13]

Mathematically, we can describe the purpose of the algorithm as this: for a given parent lattice, enumerate all possible superlattices and all rotationally and translationally unique "colorings" or labelings of each superlattice. In presenting the algorithm in Sec. II, we shall refer to superlattices and labelings rather than referring to crystal structures or atomic sites.

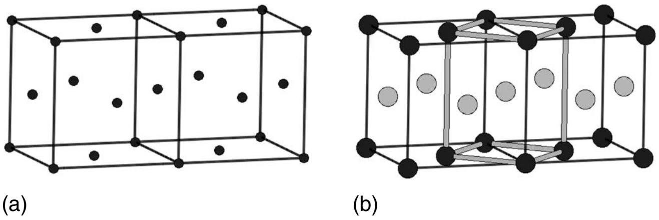


FIG. 1. An example parent lattice (left) and a superstructure (right). The parent lattice is fcc and the superlattice is defined by the (doubled) unit cell outlined in gray. The two interior points of the superlattice are occupied by one black atom and one gray atom. Together the superlattice and atoms constitute a derivative structure. The superstructure of this example is that of CuAu.

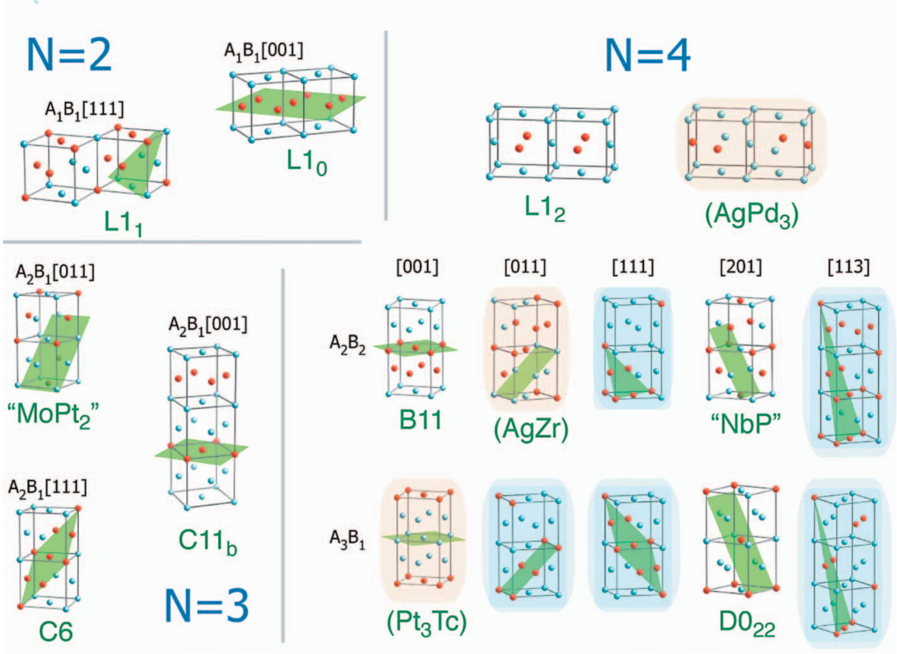


FIG. 2. (Color) The first 17 binary structures derived from the fcc lattice. All have 4 atoms/cell or less. Structures shown with a green plane can be characterized as a stacking of pure A and B atomic layers. For example, the $L1_0$ structure (upper left) is an alternating (A_1B_1) sequence of layers stacked in the [001] direction. All of the 2 and 3 atoms/cell structures have physical manifestations. Of the 4 atoms/cell structures only four have physical manifestations. Three of the others (yellow backgrounds) have been predicted to exist (Ref. 14) but not yet observed. The other five (blue backgrounds) have never been observed or predicted to exist in any system.

II. ENUMERATING ALL DERIVATIVE STRUCTURES

Here is a brief outline of the algorithm.

(1) For each superlattice of size n, generate all Hermite normal form (HNF) matrices.[3,4] (In what follows, we refer to n as the *index* of the superlattice.)

(2) Use the symmetry of the parent lattice to remove rotationally equivalent superlattices, thus shrinking the list of HNF matrices.

(3) For each superlattice index n, find the Smith normal form (SNF) of each HNF in the list.

(a) Generate a list of possible labelings (atomic configurations) for each SNF, essentially a list of all k^n numbers in a base k, n-digit system. For the labels, we use the first k letters of the alphabet, $a,b,\cdots$.

(b) Remove incomplete labelings where each of the k labels $(a,b,\cdots)$ does not appear at least once.[15]

(c) Remove labelings that are equivalent under translation of the parent lattice vectors. This reduces the list of labelings by a factor of $\sim n$.

(d) Remove labelings that are equivalent under an exchange of labels, i.e., $a \rightleftharpoons b$, so that, e.g., the labeling *aabbaa* is removed from the list because it is equivalent to *bbaabb*.

(e) Remove labelings that are superperiodic, i.e., labelings that correspond to a nonprimitive superstructure. This can be done without using the geometry of the superlattice.

(4) For each HNF, remove labelings that are permuted by symmetry operations (of the parent lattice) that leave the superlattice fixed.

An important feature of the algorithm is that the list of possible labelings, generated in step (3)(a), forms a minimal hash table with a perfect hash function. Eliminating *all* duplicate labelings in a list of N can be accomplished in $O(N)$ time. Coupled with the group-theoretical approach, this results in an extremely efficient algorithm that is orders of magnitude faster than FWZ. Enumerating fcc-derived binary structures up to $n=20$ takes five minutes with the present algorithm but more than one day with FWZ. The case of $n=24$ takes less than 2 h but about one month with FWZ. More significantly, the run time of the algorithm scales *linearly*. That is, the time to find N unique structures is proportional to N—the best possible scaling for this type of problem.[16] An illustration for binary superstructures of an fcc parent lattice is shown in Fig. 3.

A. Generating all superlattices

Given a "parent" cell (any lattice), the first step in finding all derivative structures of that cell is to enumerate all derivative superlattices. Consider the transformation $\mathbb{B}=\mathbb{A}\mathbb{H}$,

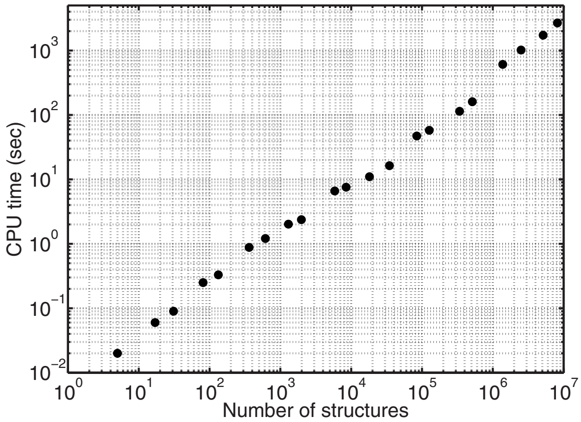


FIG. 3. Required CPU time as a function of the number of unique derivative superstructures found. The scaling is linear, the best possible scaling for this type of problem.

where $\mathbb{A}$ is the basis for the original lattice (the basis vectors listed column-wise), $\mathbb{H}$ is a matrix with all integer elements, and $\mathbb{B}$ is the matrix of the transformed lattice. If the determinant of the transformation matrix $|\mathbb{H}|$ is ± 1, then $\mathbb{H}$ is merely a change in basis that leaves the lattice represented by $\mathbb{A}$ unchanged. Matrices $\mathbb{A}$ and $\mathbb{B}$ are merely two different choices of basis for the same lattice. On the other hand, if the elements of $\mathbb{H}$ are all integers but the determinant of $\mathbb{H}$ is 2, say, then the lattice of $\mathbb{B}$ is a superlattice[4] of (i.e., a subgroup) of the lattice defined by $\mathbb{A}$, but with twice the volume of the original (parent) lattice.

Two different matrices, $\mathbb{H}_1$ and $\mathbb{H}_2$, with the same determinant, will generate different bases for the *same lattice* if and only if $\mathbb{H}_1$ can be reduced to $\mathbb{H}_2$ by elementary integer column operations. The canonical form for such operations is lower-triangular HNF. Thus, if we use only matrices $\mathbb{H}$ which are in HNF, we will produce exactly one representation of each superlattice.[3,4] In three dimensions, the lower-triangular Hermite normal form is

$$\begin{pmatrix} a & 0 & 0 \\ b & c & 0 \\ d & e & f \end{pmatrix}, \quad 0 \le b < c, \quad 0 \le d,e < f. \tag{1}$$

In this form, the product of the integers on the diagonal alone, $a \times c \times f$, fixes the determinant. Again, we refer to the superlattice size, or the determinant, as the index n. Generating all HNF matrices of a given index can be done then by finding each unique triplet, $acf = |\mathbb{H}|$, and then generating all values of b, d, and e that obey the conditions in Eq. (1).

The algorithm for generating all possible HNF matrices of a given index $|\mathbb{H}|$ is rather simple, comprising just two steps. In the first step, find all possible diagonals: find all values a, $1 \le a \le |\mathbb{H}|$, which evenly divide $|\mathbb{H}|$; for each of these values, find all c, $1 \le c \le |\mathbb{H}|/a$, which evenly divide $|\mathbb{H}|/a$. For each value of c, let $f = |\mathbb{H}|/(ac)$. For example, consider the case of $|\mathbb{H}| = 6$. We execute two nested loops over the possible values of a and c; each loop runs over all integers between 1 and the $|\mathbb{H}|$, testing the above conditions at each iteration. The loops run from 1 to 6, and the algorithm finds nine cases that meet the above conditions. They are

a	1	1	1	1	2	2	3	3	6
c	1	2	3	6	1	3	1	2	1
f	6	3	2	1	3	1	2	1	1

The set of acf triplets generated during this first step comprises all possible diagonals of the HNF matrices for the case of $n = |\mathbb{H}| = 6$. The second step, generating each set of values of b, d, and e for each diagonal (set of acf triplets), can be accomplished simply by three nested loops that start at zero and terminate at $b < c$ and $d,e < f$.

As an example of both steps, consider the case where the index is merely double that of the original lattice, i.e., where $|\mathbb{H}| = 2$. The factors of 2 are just the set {1,2}, so the first step finds only three cases: (2,1,1), (1,2,1), and (1,1,2). Then, generating the off-diagonal terms for each of these three cases, we find seven HNF matrices:

$$\text{case 1:} \quad \begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \text{case 2:} \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$\text{case 3:} \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 2 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 2 \end{pmatrix}.$$

For increasing index, $n = |\mathbb{H}| = 1, 2, 3, \cdots$, the number of HNF matrices generates an interesting sequence: 1, 7, 13, 35, 31, 91, $\cdots$. We find that the closed-form expression for nth term in the series is

$$\sum_{d|n} d\sigma(d) = \prod_{i=1}^{k} \left(\frac{(p_i^{e_i+2} - 1)(p_i^{e_i+1} - 1)}{(p_i - 1)^2 (p_i + 1)} \right), \tag{2}$$

where σ is the sum of divisor function and the p_i and e_i are the prime factors and powers of n: $n = p_1^{e1} \cdot p_2^{e2} \cdots p_k^{e_k}$. This expression is the same as that given for Sloane's A001001.[17] The sequence appears in the crystallography literature[18,19] as well as several other contexts.[20–23]

Significantly, because we have an expression for the number of superlattices, the implementation of the HNF-generating algorithm can be rigorously checked. Also note that this step of the algorithm is independent of the choice of parent lattice.

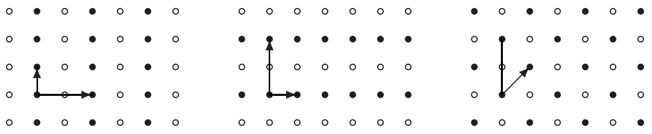
B. Reducing the Hermite normal form list by parent lattice symmetry

The set of HNF matrices defines the set of all derivative superlattices of a parent cell via the transformation mentioned above, $\mathbb{B} = \mathbb{A}\mathbb{H}$. However, not all of the superlattices in this set will be geometrically different. Some distinct lattices will be equivalent under symmetries of the parent lattice, illustrated in the example below.

Such duplicate superstructures must be eliminated by the algorithm. At the end of the algorithm, we want all derivative structures to be unique from a material point of view. So we wish to exclude from the list any superstructures that are related to others already in the list simply by a rotation, reflection, or change in basis.

As an illustration, consider a two-dimensional parent lattice that is square, that is, $\mathbb{A} = \mathbb{I}$ (the 2×2 identity matrix). There are three HNF matrices for which $|\mathbb{H}| = 2$ and three corresponding superlattices, $\mathbb{B} = \mathbb{A}\mathbb{H} = \mathbb{I}\mathbb{H} = \mathbb{H}$

$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 \\ 1 & 2 \end{pmatrix}$$



The parent lattice itself is indicated by the dots (filled and unfilled), while the superlattice is indicated by the filled dots. The vectors defined by the matrices are shown as arrows. The first two lattices are clearly equivalent under a 90° rota-

TABLE I. (Color) An example labeling for the binary case, $k=2$, with four interior points, $n=4$, in the superlattice. There are $k^n=2^4=16$ distinct labelings, but the colored labelings represent incomplete or duplicate superstructures. The yellow labelings are incomplete (not all labels are present), purple are translation duplicates, blue are label-exchange duplicates, and green are superperiodic labelings. Some labelings fall into more than one category (e.g., *baba* is both a translation duplicate and superperiodic).

aaaa	abaa	baaa	bbaa
aaab	abab	baab	bbab
aaba	abba	baba	bbba
aabb	abbb	babb	bbbb

tion, one of the eight symmetry operations of a square lattice.

To enumerate the distinct superlattices of a given index n then, we must check that each new superlattice that is added to the list is not a rotated duplicate of a previous superlattice. More precisely, we must check that each new basis $\mathbb{B}_i$ is not equivalent, under change of basis, to some symmetric image $\mathbb{R}\mathbb{B}_j$ of a basis $\mathbb{B}_j$ already in the list. In other words, we want to avoid the relation $\mathbb{B}_i=\mathbb{R}\mathbb{B}_j\mathbb{H}$, where $\mathbb{B}_i$ is a candidate superlattice, $\mathbb{R}$ is any of the rotations of the parent lattice, $\mathbb{B}_j$ is a superlattice already in the list of distinct superlattices, and $\mathbb{H}$ is any unimodular matrix of integers. (Since $\mathbb{B}_i$ and $\mathbb{B}_j$ have the same determinant, we will only need to check that $\mathbb{B}_j^{-1}\mathbb{R}^{-1}\mathbb{B}_i$ is a matrix of integers.)

For the case of cubic symmetry, the seven superlattices for the $\mathbb{H}=2$ case mentioned above reduce to only two symmetrically distinct superlattices. The corresponding derivative superstructures are $L1_0$ and $L1_1$, both well-known structures in intermetallic compounds. The fact that these are the only two 2 atoms/cell fcc structures is not coincidence or an accident of chemistry; no other 2 atoms/cell structures are *possible geometrically*. The hierarchy of physically observed structures uncovered for fcc and bcc lattices as the index is increased is discussed in Refs. 24 and 25.

C. Find the unique labelings for all superlattices

1. Generate all possible labelings

For each HNF, each superlattice, we start by generating all possible labelings of that superlattice. In other words, given k colors (types of atoms), represented by the labels $a,b,\cdots$, we generate all possible ways of labeling (coloring) the superlattice. Each HNF matrix of determinant size n represents a superlattice with n interior points to be decorated. If the number of colors is k, then the list of all possible labelings is easily represented by the list of all n-digit, base-k numbers. So, from a combinatorial point of view, there are k^n distinct labelings. For example, in the case of a binary system ($k=2$) with four interior points (index $n=4$), there are $2^4=16$ possible labelings (see Table I).

2. Concept of eliminating duplicate labelings

The rest of the algorithm deals with just one conceptual issue—given the k^n labelings (colorings) of the superlattice—eliminate the duplicates. In the FWZ algorithm and its extended implementations, duplicate structures are eliminated by comparing[26] one candidate structure to another, necessitating an expensive $O(N^2)$ search. We eliminate the duplicates via group theory rather than checking the structures themselves. Although this approach is more abstract than the geometric approach, it is much more efficient—eliminating the duplicates in a list becomes a strictly $O(N)$ procedure.

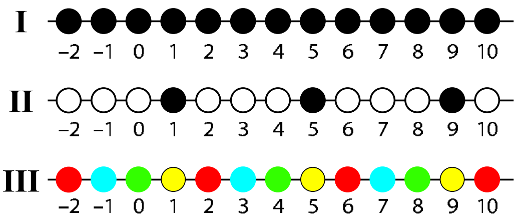


FIG. 4. (Color online) (I) One-dimensional example of a parent lattice, (II) a derivative superlattice (index $n=4$), and (III) one possible superlattice labeling.

a. One-dimensional example. We start with a simple illustration and then discuss the essential group-theoretical concepts in the context of that example. Consider the one-dimensional case of Fig. 4. The first line (I) is a parent lattice, an infinite collection of equally spaced points, identified with the set of integers, denoted **Z**. The second line (II) is a superlattice, a subset of the parent lattice (every fourth point; those colored black). The third line (III) is a *superstructure*, a "labeling" or "coloring" of the parent lattice that has the same periodicity as the superlattice. The points of the lattice play the role of positions in a crystal, and the colors play the role of atoms placed at those positions.

There are labelings that are distinct yet physically equivalent, as shown in Fig. 5. If Fig. 5 note that line (II) is obtained from line (I) by shifting the colors two units to the right, and line (III) is obtained from line (I) by shifting the numbers two units to the left—with the same result. Lines (II) and (III) are the same labeling, obtained in different ways from (I). Both are physically equivalent to (I). The fact that we can obtain such a shifted labeling either by shifting the numbers or by shifting the labels explains why we can do much of our equivalence checking within a finite group, instead of geometrically within an infinite lattice. By this method, we will identify these equivalent labelings and re-

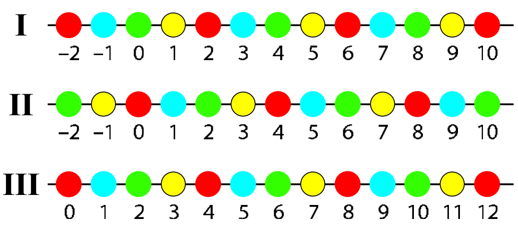


FIG. 5. (Color online) Three labelings of the superlattice of Fig. 4. Lines II and III are identical to line I except the colors (labels) are shifted. In line II the colors have been shifted two units to the right. In line III, the shift has been effected by translating the lattice two units to the left.

TABLE II. Two representations of the Cayley table for the quotient group G (cosets of the subgroup S). On the right, the elements of the group have been denoted (0,1,2,3) for notational convenience.

	$\mathbf{Z}$	$1+\mathbf{Z}$	$2+\mathbf{Z}$	$3+\mathbf{Z}$
$\mathbf{Z}$	$\mathbf{Z}$	$1+\mathbf{Z}$	$2+\mathbf{Z}$	$3+\mathbf{Z}$
$1+\mathbf{Z}$	$1+\mathbf{Z}$	$2+\mathbf{Z}$	$3+\mathbf{Z}$	$\mathbf{Z}$
$2+\mathbf{Z}$	$2+\mathbf{Z}$	$3+\mathbf{Z}$	$\mathbf{Z}$	$1+\mathbf{Z}$
$3+\mathbf{Z}$	$3+\mathbf{Z}$	$\mathbf{Z}$	$1+\mathbf{Z}$	$2+\mathbf{Z}$

	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

move them from the original list of k^n labelings.

We may illustrate the group theory approach using Fig. 4. The parent lattice (I) is the set of integers $\mathbf{Z}$, which is a *group* under the addition operation. We refer to this group as L. The superlattice is the set of multiples of 4, denoted $4\mathbf{Z}$. We refer to this *subgroup* of L as S. We label the parent lattice L in a manner which is periodic with respect to the superlattice S and note that if two points differ by an element of the superlattice, they must receive the same label. We use colors as labels in line (III) of Fig. 4 and note that every fourth point has the same color.

Notice that our superlattice, the green points, are $4\mathbf{Z}$, and the yellow points are a copy of $4\mathbf{Z}$, but *translated one unit to the right*. Thus, we may denote the latter set (the yellow points) by the set $1+4\mathbf{Z}$. Similarly, the red points are the set $2+4\mathbf{Z}$ and the blue points are $3+4\mathbf{Z}$. These four sets, $4\mathbf{Z},1+4\mathbf{Z},2+4\mathbf{Z},3+4\mathbf{Z}$, are mutually disjoint (they do not overlap), and their union is the entire parent lattice L. They are *translations* of S and thus are the *cosets* of the subgroup S. This means we can use them to form a new group, called the *quotient group* $G=L/S$ (see Table II). This new group is *finite*, having only four elements. For notational convenience we shall also refer to these four elements of G as (0,1,2,3). We only need to label the four elements of our quotient group in order to label the *entire parent lattice*.

Suppose we wish to translate a labeling (in order to identify and eliminate equivalent structures). As shown in line (I) of Fig. 5, we have labeled the elements of the quotient group as follows (using g, y, r, and b for the colors):

$$\begin{aligned} 0:&\quad 4\mathbf{Z}\rightarrow g,\\ 1:&\quad 1+4\mathbf{Z}\rightarrow y,\\ 2:&\quad 2+4\mathbf{Z}\rightarrow r,\\ 3:&\quad 3+4\mathbf{Z}\rightarrow b. \end{aligned}$$

In Fig. 5, we see that translating the labels by 2 is the same as simply adding 2 to each coordinate, thus

$$\begin{array}{rcl} 4\mathbf{Z} & \rightarrow & 2+4\mathbf{Z} \\ 1+4\mathbf{Z} & \rightarrow & 3+4\mathbf{Z} \\ 2+4\mathbf{Z} & \rightarrow & 4+4\mathbf{Z}=4\mathbf{Z} \\ 3+4\mathbf{Z} & \rightarrow & 5+4\mathbf{Z}=1+4\mathbf{Z} \end{array} \qquad \begin{array}{rcl} 0 & \rightarrow & 2 \\ 1 & \rightarrow & 3 \\ 2 & \rightarrow & 0 \\ 3 & \rightarrow & 1 \end{array}$$

The effect is the same as if we had assigned the colors differently,

$$\begin{aligned} 0:&\quad 4\mathbf{Z}\rightarrow r,\\ 1:&\quad 1+4\mathbf{Z}\rightarrow b,\\ 2:&\quad 2+4\mathbf{Z}\rightarrow g,\\ 3:&\quad 3+4\mathbf{Z}\rightarrow y. \end{aligned}$$

Translating the lattice by adding +2 to every point (moving the origin by two units) has the same effect on the labeling as if we had merely labeled the four elements of the quotient group, and then added +2 to every element of the group, producing the permutation 0→2, 1→3, 2→0, and 3→1, denoted (2,3,0,1).

Instead of determining that two labelings of the (infinite) lattice are equivalent by translation, we may simply check that the corresponding labelings of our finite quotient group $G=\mathbf{Z}_4$ are equivalent. We do this by just adding a fixed element to every element in the group, effecting a permutation of the cyclic group $\mathbf{Z}_4$. This idea—of labeling the quotient group instead of the lattice elements and checking equivalence within the group instead of by translating the lattice—may seem unduly abstract and unnecessary in one dimension, but it becomes much more efficient and crucial in higher dimensions, as we now show.

b. Application to higher dimensions. In any dimension, we have a parent lattice L and a superlattice S which is a subgroup of L. Labeling L in a manner which is periodic with respect to S is equivalent to merely labeling the elements of the quotient group $G=L/S$. Note that even though L and S are infinite sets, their quotient group is always a finite group with the same number of elements as the superlattice index n. Again, we check for equivalence by doing operations within the group instead of by lattice translation.

The key to this approach is the SNF. The SNF is useful because it provides the quotient group directly as follows. Recall that if $\mathbb{A}$ is a basis for L, then the distinct lattices of index n are uniquely characterized by bases $\mathbb{B}=\mathbb{A}\mathbb{H}$, where $\mathbb{H}$ is a matrix of determinant n in HNF. If S is given by one such basis $\mathbb{B}_1=\mathbb{A}\mathbb{H}_1$, then the quotient group $G=L/S$ can be found by converting the matrix $\mathbb{H}_1$ into SNF (which is a diagonal matrix with certain special properties; see Appendix). In higher dimensions the quotient group may not be purely cyclic, but it is a direct sum of cyclic groups which are given by the diagonal entries in the SNF matrix (see Fig. 6). For example, if the SNF matrix is

$$\begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix},$$

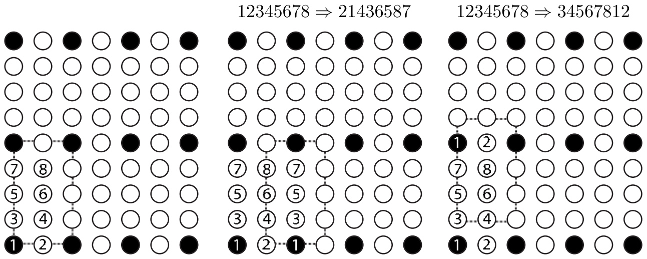


FIG. 6. A superlattice whose SNF is nontrivial (noncyclic). Because the quotient group of the superlattice is noncyclic, translations of the lattice are not cyclic permutations. Instead the sites are permuted in groups of two or four with each translation (rather than permuting as a single group of 8). In the figure, three different pictures of the same superlattice are shown. In each case a different origin is chosen. In the second case, the translation permutes two groups of four sites and in the third, four pairs are permuted.

then the quotient group $G=L/S$ is the direct sum $\mathbf{Z}_2\oplus\mathbf{Z}_4$.

In relation to the algorithm, there are two important facts to note about the SNF. (i) The SNF provides the quotient group directly, which in turn is the key to implementing an $O(N)$ algorithm. (ii) The number of SNFs (and so quotient groups) is small compared to the number of distinct lattices of index n (see Table III). This means that translation duplicates can be removed from the k^n list for hundreds or thousands of *different* superlattices *simultaneously*. (The surprising geometric implications of this are discussed in Appendix.) This reduces the running time by many orders of magnitude.

3. Eliminating translation duplicates

Because of its periodicity, the choice of origin of a superlattice is arbitrary. A change in origin implies a permutation of the labels which nonetheless defines the same superstructure (compare lines I and III in Fig. 5). As stated previously, by examining the quotient group instead of directly comparing the structures, the duplicate labelings can be readily identified. For example, consider the case for $n=4$. Adding each member to the quotient group $\mathbf{Z}_4=(0,1,2,3)$ produces four permutations as follows:

Member	Mapping	Permutation
0	$0\rightarrow 0,\ 1\rightarrow 1,\ 2\rightarrow 2,\ 3\rightarrow 3$	(0,1,2,3)
1	$0\rightarrow 1\rightarrow 2\rightarrow 3\rightarrow 0$	(1,2,3,0)
2	$0\rightarrow 2\rightarrow 0,\ 1\rightarrow 3\rightarrow 1$	(2,3,0,1)
3	$0\rightarrow 3\rightarrow 2\rightarrow 1\rightarrow 0$	(3,0,1,2)

If we take the 14 complete labelings of Table I and the three nontrivial permutations above, we find that ten are duplicates (colored purple in Table I),

	duplicates		
(0,1,2,3)	(1,2,3,0)	(2,3,0,1)	(3,0,1,2)
aaab	aaba	abaa	baaa
aabb	abba	bbaa	baab
abab	baba	abab	baba
abbb	bbba	bbab	babb

Of the original $2^4=16$ labelings, two were discarded immediately because they were incomplete. Of the remaining 14, 10 are translation duplicates, leaving 4 that are translationally inequivalent (left column above).

4. Remove "label-exchange" duplicates

The next step in the algorithm is to remove labelings that are equivalent under exchange of labels. Structurally, there is no difference between a superlattice whose interior points are labeled *aaab* versus *bbba*. Although the energy of an isostructural compound with composition X_3Y_1 is different from one with composition X_1Y_3, we only wish to include one entry in our list of derivative superstructures because the full composition list can always be recovered by making all possible label exchanges (i.e., $a\rightleftharpoons b$). In the example above, four labelings were unique under translations:

aaab,

aabb,

abab,

abbb.

However, the first and the fourth are equivalent by exchanging $a\rightleftharpoons b$ and applying the permutation (1,2,3,0).

5. Remove superperiodic labelings (nonprimitive structures)

At this point of the algorithm, many of the duplicate labelings have been removed from the original k^n list. However, there are still more duplicates to remove. Some of the labelings in the list will represent superstructures that are not primitive. In other words, the labelings will be superperiodic—they will have periods shorter scale than the superlattice.[27]

The superperiodic duplicates are easily identified because they are identical under at least one permutation. The quotient group G dictates a set of permutations under which the labelings are duplicate. One of these permutations will leave the labeling unchanged if it is superperiodic. For example, continuing the example above, three unique labelings are still in the list: *aaab*, *aabb*, and *abab*. One of the permutations of the quotient group $G=\mathbf{Z}_4$ is (2,3,0,1). Under this permutation, the labeling *abab* is unchanged. Thus it is superperiodic, as depicted in Fig. 7. It is a duplicate in the sense that the algorithm would have already enumerated this structure with the index $n=2$ structures.

6. For each Hermite normal form: remove "label-rotation" duplicates

The previous three steps of the algorithm yield a list of distinct labelings for each SNF of index n. Three kinds of

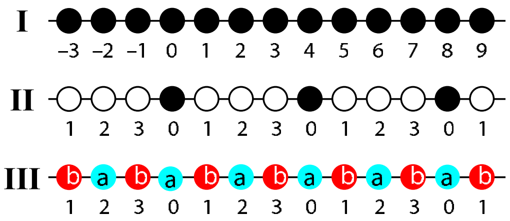


FIG. 7. (Color online) (I) One-dimensional example of a parent lattice, (II) a derivative superlattice (index $n=4$), and (III) a superperiodic (or nonprimitive) labeling. Although the index of the superlattice is $n=4$, the structure can be represented by a superlattice labeling of period 2 instead of 4. The superstructure of line III would have been found as an index $n=2$ derivative structure and is therefore a duplicate.

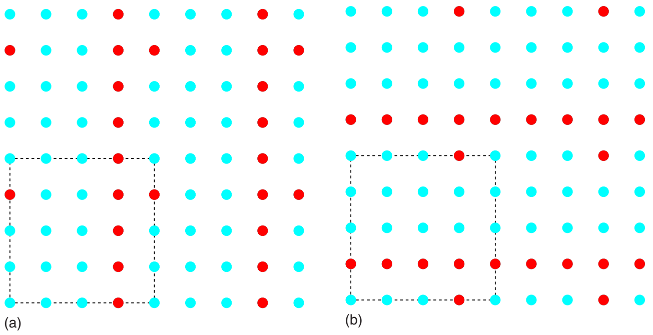


FIG. 8. (Color online) Two identical 4×4 superlattices (dotted lines) with rotationally equivalent labelings (red and blue circles). Although the superlattices themselves are unchanged, a 90° rotation applied to the left labeling yields that shown on the right. Thus the second is a duplicate of the first and should be removed from the list of labelings.

duplicate labelings (translation duplicates, label-exchange duplicates, and superperiodic duplicates) have already been removed. One kind of duplicate remains, however, and these are eliminated in the current step.

This step removes labelings which are permuted by the rotations of the parent lattice. Whereas the preceding steps were applied to generate a list of unique labelings *for each SNF*, the current step must be applied to *each HNF*. In other words, this step must be applied to the surviving labelings *separately* for each superlattice.

Superlattices which are not fixed by rotations of the parent lattice were already eliminated as duplicates in step (2) of the algorithm. However, rotations which leave the superlattice unchanged may still permute the labeling itself. Such permutations are physically equivalent (merely rotated with respect to one another). So any two labelings which are equivalent under rotations that fix the superlattice are duplicate and one must be removed from the list. Figure 8 illustrates the situation in two dimensions.

Here again, the group theory approach and the SNF make the search extremely efficient. Label-rotation duplicates can be identified easily using the properties of the quotient group and the SNF transformation. The row and column operations required to transform the HNF matrix of a superlattice into its SNF can be represented by two integer transform matrices, $\mathbb{L}$ and $\mathbb{R}$, so that $\mathbb{L}\mathbb{H}\mathbb{R}=\mathbb{S}$, where $\mathbb{S}$ is the SNF. This step of the algorithm is implemented using the left transformation matrix $\mathbb{L}$.

Let $\mathbb{G}$ be a $3\times n$ matrix where each member of the quotient group is represented as a column[28] in $\mathbb{G}$ and let $\mathbb{R}$ be one of the rotations that fixes the superlattice. Then the permutation of the labels (which is the same as the permutations of the quotient group) enacted by the rotation $\mathbb{R}$ is given by

$$\mathbb{G}' = \mathbb{L}\mathbb{A}^{-1}\mathbb{R}(\mathbb{L}\mathbb{A}^{-1})^{-1}\mathbb{G} \tag{3}$$

where columns of $\mathbb{A}$ are the lattice vectors of the parent lattice and $\mathbb{L}$ is the left transformation matrix for the SNF.

The power of this expression is that it allows the label-rotation duplicates to be identified by working entirely within the quotient group, without requiring any explicit reference to the geometry of the superlattice. Thus, as in the other steps, duplicate labelings can be eliminated in a time proportional only to the number of labelings in the list.

III. EXAMPLES

In this section, we give several example derivative structure lists enumerated by the algorithm. We discuss the symmetry reduction of the structure lists and then give results for several cases. First, we compare the fcc/bcc binary list to that generated by the FWZ algorithm. We also show the small-unit-cell binary structures for a simple-cubic parent lattice and the small-unit-cell *ternary* structures for an fcc parent lattice.

A. Symmetry reduction of superlattice lists

In step (2) of the algorithm, the complete list of HNF matrices is reduced to those that are unique under the symmetry operations of the parent lattice. Asymptotically, the factor by which the list is reduced is one half the order of the rotation group of the parent lattice. For example, for cubic parent lattices (simple cubic, face-centered cubic, or body-centered cubic), the point group contains 48 rotations (proper

TABLE III. Table showing the number of Hermite normal form (HNF) matrices and Smith normal form (SNF) matrices as a function of index n (determinant size). The number of HNFs is a rapidly increasing function of n [see Eq. (2)], whereas the number of SNFs grows very slowly.

n	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
HNFs	7	13	35	31	91	57	155	130	217	133	455	183	399	403	651
SNFs	1	1	2	1	1	1	3	2	1	1	2	1	1	1	4

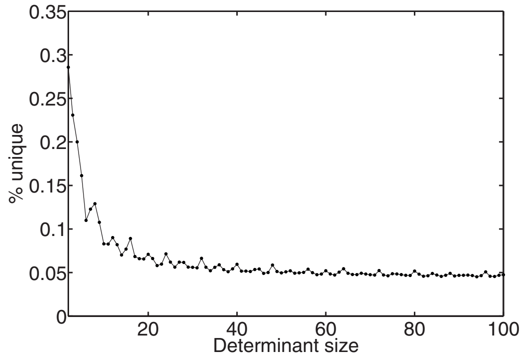


FIG. 9. Symmetrically unique HNF matrices as a function of the index n (determinant size) for an fcc parent lattice. Asymptotically, the fraction of unique superlattices approaches $2/N=1/24\approx5\%$, where $N=48$ is the number of symmetry operations of the fcc parent lattice.

and improper). For superlattices with large index n, the number of HNFs is reduced by a factor of $48/2=24$. Because every lattice is symmetric under inversion, only the proper rotations (i.e., not reflections) need to be considered in the reduction (thus the factor of 1/2). Figure 9 shows the fraction of symmetrically distinct superlattices for determinant sizes of up to 100, while Fig. 10 shows the actual number of fcc-based superlattices compared to the total number of distinct HNF matrices.

For an fcc or bcc parent lattice (the numbers are the same), the number of unique lattices as a function of index n (cell size) appears to be equivalent to the Sloane sequence A045790.[29] For the sequences generated for other parent lattices, which accordingly have a different symmetry group, there are no known number-theoretic connections. Surprisingly, this is even true for the simple-cubic lattice. For the simple-cubic lattice, the sequence is identical to the fcc/bcc one for *odd* values of the index n but larger for the even values (see Table IV).

B. Number of structures of different parent lattices

The number of superstructures increases much faster than the number of superlattices as a function of n. In general,

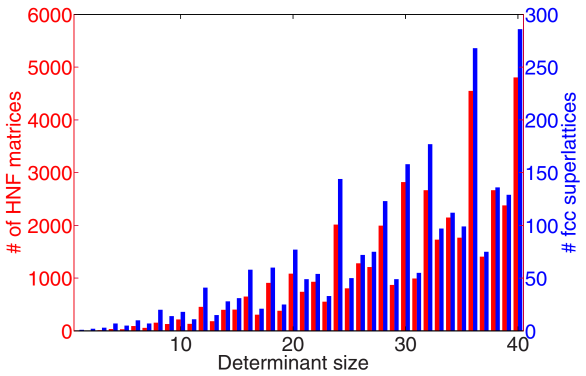


FIG. 10. (Color online) Left axis (red): number of HNF matrices as a function of determinant size. Right axis (blue): number of inequivalent fcc superlattices as a function of volume.

TABLE IV. Number of symmetrically unique superlattices (HNFs) for several different parent lattices as a function of the index size n. Note that fcc/bcc is the same as simple cubic (sc) only for the odd values of n and always smaller for the even values. Hexagonal (hex) and simple tetragonal parent lattices have more unique lattices than the cubic systems because of their lower symmetry.

Index	No. of superlattices			
n	fcc/bcc	sc	hex	tetragonal
2	2	3	3	5
3	3	3	5	5
4	7	9	11	17
5	5	5	7	9
6	10	13	19	29
7	7	7	11	13
8	20	24	34	51
9	14	14	23	28
10	18	23	33	53

TABLE V. Number of unique fcc derivative structures as a function of the index n. The second and fifth columns show the number of unique structures for each n, while the third and sixth columns show the cumulative total.

n	Structures	Cumulative
2	2	2
3	3	5
4	12	17
5	14	31
6	50	81
7	52	133
8	229	362
9	252	614
10	685	1299
11	682	1981
12	3875	5856
13	2624	8480
14	9628	181 08
15	165 84	3469
16	497 64	844 56
17	421 35	126 591
18	212 612	339 203
19	174 104	513 307
20	867 893	138 120 0
21	112 070 8	250 190 8
22	262 818 0	513 008 8
23	304 273 2	817 282 0

TABLE VI. Simple cubic superstructures for $n \leq 4$.

n	HNF	SNF	Superlattices	Labelings
2	7	1	3	3
3	13	1	3	3
4	35	2	9	15
Total	55	4	15	21

each superlattice has many different unique labelings. Table V shows the number of fcc/bcc derivative structures as a function of n. The FWZ begins to undercount (as expected) at $n=15$ but the FWZ count is probably sufficient for applications where it was used. Our algorithm is formally complete and does not undercount.

Table VI lists the number of superlattices and superstructures for the simple-cubic lattice when $n \leq 4$. The corresponding structures are visualized in Fig. 11 (compare this to Fig. 2). There are more simple-cubic derivitive structures than fcc/bcc because there are more superlattices for a simple-cubic parent lattice than for an fcc/bcc parent.

Similar to the fcc case shown in Fig. 2, most of the simple-cubic superstructures can be characterized as stackings of pure A and B planes. The stacking directions are indicated in the figure. In contrast to the fcc case, there are six unique stacking directions. It is interesting to note that the three structures that cannot be characterized as pure stackings are the only ones corresponding to a composite quotient group, namely, $G=\mathbf{Z}_2 \oplus \mathbf{Z}_2$. This is also true for the nonstacked structures in the fcc case (Fig. 2), $L1_2$ and $AgPd_3$. For the "stackable" structures, the quotient group is a single cyclic group, $\mathbf{Z}_4$.

TABLE VII. Number of ternary and quaternary derivative structures for an fcc parent lattice. Compare to the number of binary structures of Table V. As the number of labels k is increased, the number of derivative structures increases rapidly.

n	Ternary		Quaternary	
	Structures	Cumulative	Structures	Cumulative
3	3	3		
4	13	16	7	7
5	23	39	9	16
6	130	169	110	126
7	197	366	211	337
8	1267	1633	2110	2447
9	2322	3955	5471	7918
10	9332	13287	32362	40280

Table VII lists the number of fcc/bcc ternary and *quaternary* derivative structures. A figure displaying the ternary structures for $n \leq 4$ is unnecessary—the ternary structures have the same unit cell as the binary structures, only the labelings are different. For $n=3$ the labeling *aab* is replaced by *abc*. For $n=4$ the labelings *aaab* and *aabb* are replaced by *aabc* and *abac*, and the $AgPd_3$ structure has both labelings, rather than one.

IV. SUMMARY

We developed an algorithm for enumerating derivative structures. The results of such an algorithm are useful in a variety of physics and materials science contexts. The algorithm first generates all unique superlattices by enumerating all Hermite normal form matrices and using the symmetry operations of the parent lattice to eliminate rotationally equivalent superlattices. Next, the algorithm generates all possible atomic configurations (labelings) of each superlat-

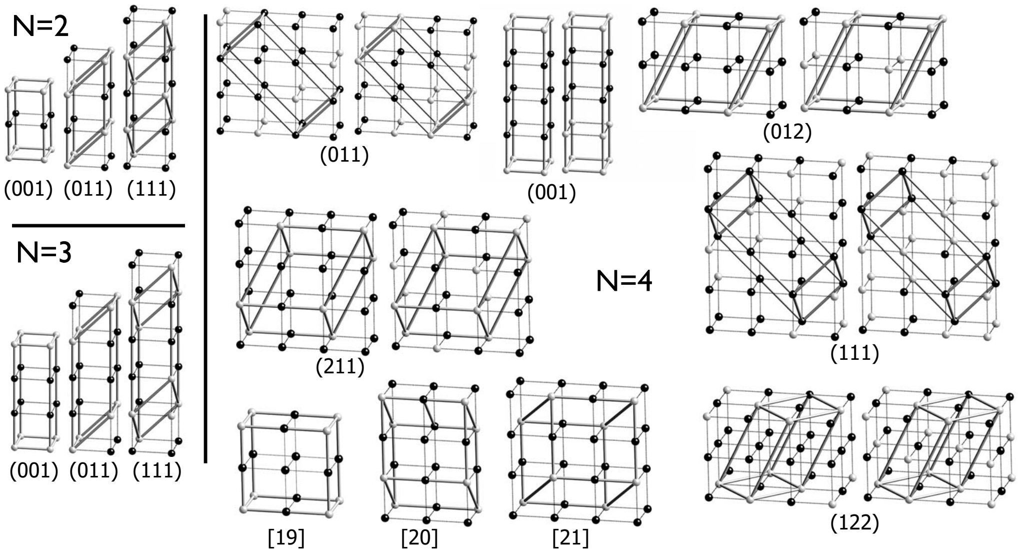


FIG. 11. The first 21 binary structures derived from the simple-cubic lattice. All have 4 atoms/cell or less. Structures marked with a crystallographic direction (hkl) can be characterized as a stacking of pure A and B atomic layers (black and white spheres). For $n=4$ all of the stacked structures occur in pairs, A_3B_1 and A_2B_2. The last three structures, labeled 19, 20, and 21, cannot be characterized as pure stackings. These structures have basis vectors whose corresponding quotient group is $G=\mathbf{Z}_2 \oplus \mathbf{Z}_2$ (rather than $\mathbf{Z}_4$).

tice and eliminates duplicates using a group-theoretical approach rather than geometric analysis.

The algorithm is exceptionally efficient due to the use of (i) perfect, minimal hash tables and (ii) a group-theoretical approach to eliminating duplicate structures. These two features result in a linearly scaling algorithm that is orders of magnitude faster than the previous method. Moreover, the method can be applied to any parent lattice and to arbitrary k-nary systems (binary, ternary, quaternary, etc.). The method is formally complete (does not undercount) and the key parts of the algorithm (and its implementation) can be rigorously checked by number theory results and Burnside's lemma.

ACKNOWLEDGMENTS

G.L.W.H. gratefully acknowledges financial support from the National Science Foundation through Grant No. DMR-0650406. G.L.W.H. thanks Martin L. Searcy and Bronson S. Argyle for useful discussions concerning algorithm development and implementation of hash tables and hash functions. We also wish to thank Axel van de Walle whose input was helpful in testing the code.

APPENDIX

1. Hermite normal form

If L is a lattice, with basis given by the columns of a square matrix $\mathbb{A}$, and S is a superlattice, then S will have basis $\mathbb{AM}$, where $\mathbb{M}$ is a square matrix of integers. Furthermore, all bases of S will have the form $\mathbb{AMN}$, where $\mathbb{N}$ is an integer matrix with determinant ± 1. Thus, to find a canonical basis for S, we may use elementary *integer* column operations on $\mathbb{M}$ to make it lower triangular, with positive entries down the diagonal. Furthermore, we can arrange that the lower-triangular matrix $\mathbb{H}=\mathbb{MN}$ have the property that every off-diagonal element is less than the diagonal element in its row. Such a matrix $\mathbb{H}$ is said to be in Hermite normal form and is unique with respect to the matrix $\mathbb{M}$.

Thus, if the determinant of $\mathbb{M}$ is n, then the number of superlattices S of L with index n is equal to the number of distinct HNF matrices with determinant n. In three dimensions, that number is

$$\sum_{d|n} d\sigma(d) = \prod_{i=1}^{k} \left(\frac{(p_i^{e_i+2}-1)(p_i^{e_i+1}-1)}{(p_i-1)^2(p_i+1)} \right),$$

where $n=\Pi p_i^{e_i}$ is the prime factorization of n.

2. Smith normal form

Using elementary integer row and column operations (adding or subtracting an integer multiple of one row or column to another, multiplying a row or column by ± 1, or exchanging two rows or columns), we may reduce the integer matrix $\mathbb{M}$ to a diagonal matrix $\mathbb{D}$ with the following properties.

(i) Each diagonal entry of $\mathbb{D}$ divides the next one down.

(ii) The product of the diagonal entries of $\mathbb{D}$ is the absolute value of the determinant of $\mathbb{M}$.

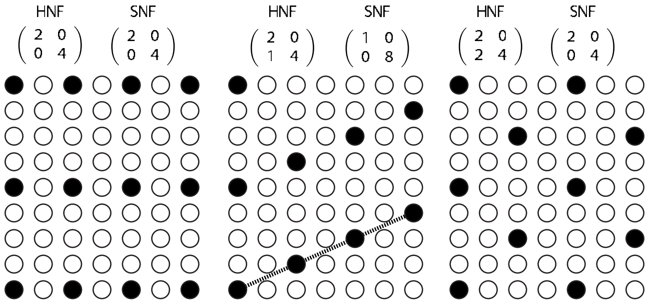

HNF
SNF
$\begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix}$
$\begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix}$
HNF
SNF
$\begin{pmatrix} 2 & 0 \\ 1 & 4 \end{pmatrix}$
$\begin{pmatrix} 1 & 0 \\ 0 & 8 \end{pmatrix}$
HNF
SNF
$\begin{pmatrix} 2 & 0 \\ 2 & 4 \end{pmatrix}$
$\begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix}$

FIG. 12. Two-dimensional superlattices of index $n=8$. The first and third share the same SNF, while the second has the trivial SNF, implying a purely cyclic quotient group. In contrast to the first and third superlattices, the lattice points in layers parallel to the superlattice edge (dotted line) must all have the same label.

This is called the *Smith normal form* of $\mathbb{M}$. In the lattice case, where L is a lattice with basis $\mathbb{A}$ and S is a superlattice (subgroup) with basis $\mathbb{AM}$, then $\mathbb{D}$ describes the quotient group L/S as a direct sum of cyclic groups. The diagonal entries of $\mathbb{D}$ are the orders of the cyclic direct summands of the quotient group (as in the Fundamental Theorem of Finite Abelian Groups). For example, using the notation $\mathbf{Z}_n = \mathbf{Z}/n\mathbf{Z}$, if

$$D = \begin{pmatrix} D_{11} & 0 & 0 \\ 0 & D_{22} & 0 \\ 0 & 0 & D_{33} \end{pmatrix},$$

then

$$L/S \cong G = \mathbf{Z}_{D_{11}} \oplus \mathbf{Z}_{D_{22}} \oplus \mathbf{Z}_{D_{33}}.$$

A simple, two-dimensional example may help us to show how this affects our lattice labeling problem. Consider the three matrices (all in HNF form):

$$\mathbb{H}_0 = \begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix}, \quad \mathbb{H}_1 = \begin{pmatrix} 2 & 0 \\ 1 & 4 \end{pmatrix}, \quad \mathbb{H}_2 = \begin{pmatrix} 2 & 0 \\ 2 & 4 \end{pmatrix}.$$

The matrices $\mathbb{H}_0$ and $\mathbb{H}_2$ both reduce to the SNF matrix

$$\begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix},$$

which corresponds to a quotient group which is Abelian, but not cyclic, but the middle matrix $\mathbb{H}_1$ reduces to SNF matrix

$$\begin{pmatrix} 1 & 0 \\ 0 & 8 \end{pmatrix},$$

corresponding to the cyclic group of order 8. Thus, if we take $\mathbb{A}$ to be the identity matrix, so $L=\mathbf{Z}^2$, and let S_i be the lattice with basis $\mathbb{H}_i$, then L/S_0 and L/S_2 are each isomorphic to the group $\mathbf{Z}_2 \oplus \mathbf{Z}_4$, while L/S_1 is isomorphic to the cyclic group of order 8.

The fact that the latter is cyclic means that we can *layer* the parent lattice in such a way that each parallel layer consists of points which all must get the same label (see Fig. 12). We can arbitrarily label each layer passing through the interior points of the basis parallelogram and label the rest of the lattice cyclically, as if labeling a one-dimensional lattice. The quotient group for S_0 or S_2 is not cyclic but can just as easily be used to determine equivalent labelings.

In general, SNF provides a natural homomorphism from the parent lattice L onto the direct-sum group $G=\mathbf{Z}_{D_{11}} \oplus \mathbf{Z}_{D_{22}} \oplus \mathbf{Z}_{D_{33}}$, with kernel S. By the First Homomorphism Theorem, it effectively gives an isomorphism from L/S to G. Since we do only elementary integer row and column operations, we may write $\mathbb{D}=\mathbb{P}\mathbb{M}\mathbb{Q}$, where the transition matrices $\mathbb{P}$ and $\mathbb{Q}$ are integer matrices with determinant ±1. Note that $\mathbb{A}\mathbb{M}\mathbb{Q}$ is another basis for S, so an element $x \in S$ if and only if $\mathbb{M}\mathbb{Q}z=\mathbb{A}^{-1}x$ for some integer column vector z, which is true if and only if $\mathbb{D}z=\mathbb{P}\mathbb{A}^{-1}x$. So the map

$$x \mapsto \mathbb{P}\mathbb{A}^{-1}x \left[\bmod \begin{pmatrix} D_{11} \\ D_{22} \\ D_{33} \end{pmatrix} \right]$$

(meaning that each row of the resulting column matrix is reduced modulo the corresponding diagonal element of $\mathbb{D}$) maps from L into the direct-sum group G, with its kernel being the superlattice S.

As for computing the SNF of a matrix, there are special algorithms designed to compute it efficiently when $\mathbb{M}$ is very large but the simplest algorithm, effective for small (e.g., 3×3) matrices, is basically an extension of Euclid's algorithm for finding the greatest common divisor of two numbers. One subtracts multiples of elements in the matrix from other elements in the same row or column (using column or row operations, respectively) until the greatest common divisor of all the elements of $\mathbb{M}$ is exposed. That element is then moved to the upper left corner of the matrix and used to zero out all other elements in the first row and in the first column. Then one applies the same algorithm to the 2×2 submatrix in the lower right. Thus, in particular, the upper left entry in $\mathbb{D}$ is always the greatest common divisor of all the entries in $\mathbb{M}$.

Note that the number of 3×3 SNF matrices with determinant n is given by $\Pi_i P_3(e_i)$, where $n=\Pi_i p_i^{e_i}$ (the prime factorization) and $P_3(k)$ is the number of partitions of an integer k using *at most three summands*.[30]

[1] M. J. Buerger, J. Chem. Phys. **15**, 1 (1947).

[2] In the mathematical literature, and in some of the crystallography literature, these derivative lattices are referred to as sublattices. Although this nomenclature is more correct from a mathematical and/or group theory point of view, we follow the nomenclature typically seen in the physics literature where a lattice (or a structure) whose volume is *larger* than that of the parent is referred as a superlattice (or a superstructure).

[3] A. Santoro and A. D. Mighell, Acta Crystallogr., Sect. A: Cryst. Phys., Diffr., Theor. Gen. Crystallogr. **29**, 169 (1973).

[4] A. Santoro and A. D. Mighell, Acta Crystallogr., Sect. A: Cryst. Phys., Diffr., Theor. Gen. Crystallogr. **28**, 284 (1972).

[5] P. A. Graf, K. Kim, W. B. Jones, and G. L. W. Hart, Appl. Phys. Lett. **87**, 243111 (2005).

[6] L. G. Ferreira, S.-H. Wei, and A. Zunger, Int. J. Supercomput. Appl. **5**, 34 (1991).

[7] We recently found a body of work by Hosoya (Ref. 31) that is related to our enumeration problem. Hosoya's algorithm enumerates all possible crystal structures of any parent lattice of which derivative structures are a subset. Though this approach is not useful for the normal applications of derivative structures, interested readers may wish to consult Ref. 31 and its preceding works.

[8] The general idea of the superlattice enumeration step in the FWZ algorithm is to find the most compact form for a unit cell of given volume and then generate all possible triads of unit vectors subject to this constraint. The original FWZ paper described it in terms specific to fcc/bcc lattices. Axel van de Walle's generalization can be described as follows. (1) Note that there is unique way to write a unit cell so that the superlattice vectors are as short as possible; determine the upper bound on the length of the longest lattice vector in this representation for a given superlattice volume. (2) Determine the upper bound for the length of the second longest lattice vector. (3) The third lattice vector is now considerably constrained. Generate the set of all vectors subject to the above constraints. (4) Generate all possible triads of vectors from the set found in the previous step.

[9] N. A. Zarkevich, T. L. Tan, and D. D. Johnson, Phys. Rev. B **75**, 104203 (2007).

[10] D. Johnson (private communication).

[11] A. van de Walle and G. Ceder, J. Phase Equilib. **23**, 348 (2002).

[12] A. van de Walle, M. Asta, and G. Ceder, CALPHAD: Comput. Coupling Phase Diagrams Thermochem. **26**, 539 (2002).

[13] See EPAPS Document No. E-PRBMDO-77-001822 to download the code. For more information on EPAPS, see http://www.aip.org/pubservs/epaps.html.

[14] S. Curtarolo, D. Morgan, and G. Ceder, CALPHAD: Comput. Coupling Phase Diagrams Thermochem. **29**, 163 (2005).

[15] In some applications it is useful to include the incomplete labelings. For example, in a ternary alloy, the incomplete labelings represent the binary structures on the three edges of the phase diagram. Removal of the incomplete labelings can be disabled in the included implementation of the algorithm (see Ref. 13).

[16] Of course, for a single superlattice, N itself scales as k^n, where k is the number of labels (colors) and n is the index of the superlattice. So even with a linear scaling algorithm enumerating all superstructures for modestly large n is impractical.

[17] N. J. A. Sloane, The online encyclopedia of integer sequences (www.research.att.com/ njas/sequences/).

[18] Y. Billiet and M. R. Le Coz, Acta Crystallogr., Sect. A: Cryst. Phys., Diffr., Theor. Gen. Crystallogr. **36**, 242 (1980).

[19] J. S. Rutherford, Acta Crystallogr., Sect. A: Cryst. Phys., Diffr., Theor. Gen. Crystallogr. **48**, 500 (1992).

[20] V. A. Liskovets and A. Mednykh, Commun. Algebra **28**, 1717 (2000).

[21] M. Baake, *Mathematics of Long-Range Aperiodic Order* (Kluwer, Dordrecht, 1997), pp. 9–44.

[22] R. P. Stanley, *Enumerative Combinatorics* (Cambridge Univer-

sity Press, Cambridge, 1999), Vol. 2.

[23] N. J. A. Sloane, Sublattices of index n in generic three-dimensional lattice (http://www.research.att.com/ njas/sequences/A001001).

[24] G. L. W. Hart, Nat. Mater. **6**, 941 (2007).

[25] G. L. W. Hart (unpublished).

[26] In the FWZ algorithm and the implementation of Ref. 9 correlations between sites in each structure are computed and then these correlations are compared, still requiring an $O(N^2)$ search in addition to the overhead of computing the correlations themselves. In the original implementation of FZW, the set of correlations is restricted resulting in undercounting—some formally distinct structures are identified as duplicates.

[27] An analytical check on the counting for these kinds of duplicates can be found in Rutherford's work (Ref. 32).

[28] In three dimensions, each member of the quotient group formally has three components, even though in many cases only one is nontrivial. For example, if the diagonal elements of the SNF of a superlattice are (1, 1, 4), then the four members of the image group are (0,0,0), (0,0,1), (0,0,2), and (0,0,3). On the other hand, if the diagonal elements of the SNF are (1,2,2), then the four members of the image group are (0,0,0), (0,0,1), (0,1,0), and (0,1,1).

[29] J.-O. Moussafir, Three-dimensional simplices of determinant n (http://www.research.att.com/ njas/sequences/A045790).

[30] The partition idea was introduced into the crystallographic literature in Ref. 33.

[31] M. Hosoya, Bull. of the College of Science, University of the Ryukyus **44**, 11 (1987).

[32] J. S. Rutherford, Acta Crystallogr., Sect. A: Cryst. Phys., Diffr., Theor. Gen. Crystallogr. **51**, 672 (1995).

[33] M. Kucab, Acta Crystallogr., Sect. A: Cryst. Phys., Diffr., Theor. Gen. Crystallogr. **37**, 17 (1981).