

ARTICLE OPEN



Comparing crystal structures with symmetry and geometry

John C. Thomas^{1,2}, Anirudh Raju Natarajan^{1,2} and Anton Van der Ven¹

Measuring the similarity between two arbitrary crystal structures is a common challenge in crystallography and materials science. Although there are an infinite number of ways to mathematically relate two crystal structures, only a few are physically meaningful. Here we introduce both a geometry-based and a symmetry-adapted similarity metric to compare crystal structures. Using crystal symmetry and combinatorial optimization we describe an algorithm to arrive at the structural relationship that minimizes these similarity metrics across all possible maps between any pair of crystal structures. The approach makes it possible to (i) identify pairs of crystal structures that are identical, (ii) quantitatively measure the similarity between crystal structures, and (iii) find and rank structural transformation pathways between any pair of crystal structures. We discuss the advantages of using the symmetry-adapted cost metric over the geometric cost. Finally, we show that all known structural transformation pathways between common crystal structures are recovered with the mapping algorithm. The methodology presented in this study will be of value to efforts that seek to catalogue crystal structures, identify structural transformation pathways or prune large first-principles datasets used to parameterize on-lattice Hamiltonians.

npj Computational Materials (2021)7:164; <https://doi.org/10.1038/s41524-021-00627-0>

INTRODUCTION

The crystal structure of a material determines much of its functional behavior, including its electronic, ionic, and mechanical properties. The number of possible crystal structures is infinite, and a large number of crystal structures have already been characterized¹. This number will undoubtedly continue to grow as researchers expand into the vast, high-dimensional composition spaces of multi-principle element alloys and functional compounds.

Although great effort has been expended to cataloging^{2–4} and categorizing unique crystal structures and their chemical manifestations, challenges remain to realizing a universal classification scheme. Such a scheme is becoming increasingly desirable with the growth of first-principles databases, such as the Materials Project⁵, AFLOW⁶, OQMD⁷, NOMAD⁸, and the Materials Cloud⁹. Essential to a successful classification scheme is a robust method with which to measure the structural similarity of two crystals. Unfortunately, the comparison of two crystal structures is made challenging due to their periodic nature, which allows for an infinite number of mathematical representations. Symmetry can be used to help classify crystal structures. However, while there are only 230 crystallographic space groups, two crystal structures belonging to the same space group can nevertheless be very different from the point of view of local connectivity, unit cell size and number of distinct sites within the unit cell.

Pathways that connect different crystal structures are also of great importance. Many crystalline materials undergo structural transformations as a function of temperature and pressure^{10–18}. Only a limited number of continuous pathways connecting common crystal structures have been identified^{19–27}. Knowledge of these pathways, however, is often invaluable to understand and manipulate phase transformations in shape-memory alloys and actuator applications. The identification of pathways requires an enumeration of possible mappings of one crystal onto another²⁴ and a robust definition of a distance metric between crystal structures to enable a ranking of such mappings.

Several methods and accompanying software packages have been developed to compare and classify crystal structures. These approaches tend to rely on structure standardization, space group and Wyckoff position identification and comparisons of crystal invariants^{28–31}. Other methods rely on comparisons of pair distribution functions³². Some of the well known or recently released software packages include STRUCTURE-TIDY²⁸, CRY-COM³⁰, CMPZ³³, XTAL-COMP³⁴, COMPSTRU³¹, SPAP³⁵, the pymatgen Structure Matcher³⁶, and AFLOW-XtalFinder³⁷. While they all have their merits, none determine the similarity between crystal structures using combinatorial mapping approaches with a rigorous quantification of strain and displacement costs that account for symmetry. Furthermore, there are no software tools generally available with which to systematically enumerate and rank mappings between pairs of crystal structures.

In this paper, we describe an algorithm implemented in the CASM software package^{15,38,39} to enumerate mappings between pairs of crystal structures. As part of the mapping approach, we introduce a geometric and symmetry-breaking distance metric that measures the similarity between two crystal structures. The tools allow users to answer three questions. Given two input crystal structures:

- Are they identical up to a symmetry-preserving scaling factor?
- If the two crystal structures are not identical, how similar are they?
- What are ranked pathways between the two crystal structures and what is the orientation relationship between these crystal structures for a particular path?

Answering these questions requires (i) a robust way to map one crystal onto another and (ii) metrics of similarity between any pair of crystal structures. In the next sections we describe the mathematical underpinning of the crystal mapping tool implemented in the CASM software package.

¹Materials Department, University of California, Santa Barbara, CA, USA. ²These authors contributed equally. John C Thomas, Anirudh Raju Natarajan. ✉email: johnct@umich.edu, anirudh@ucsb.edu; avdv@ucsb.edu

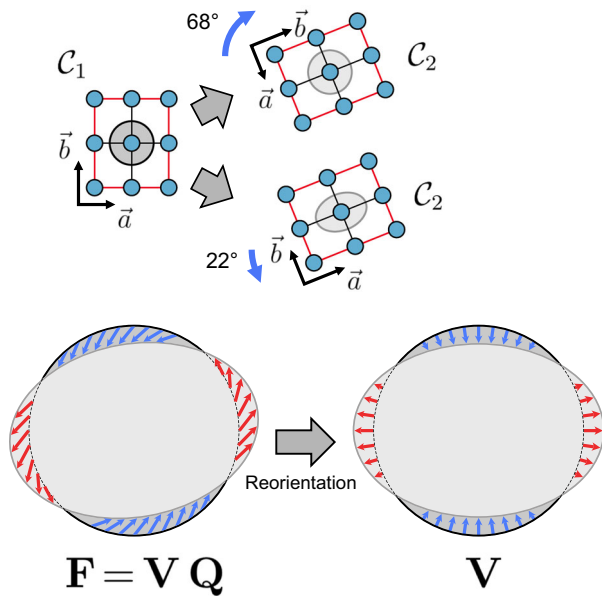


Fig. 1 Multiplicity of lattice mappings. Depiction of two possible orientation relationships between C_1 and C_2 . In one case, C_2 is rotated by -68° without an accompanying strain, in the other case, C_2 is rotated by 22° and also strained. Bottom illustration depicts the definition of the strain cost both before and after removal of rigid rotation. The strain cost is minimized at a unique value when the rigid rotation is removed.

RESULTS

The objective of this study is to obtain a relationship between the spatial coordinates of atoms in two crystals, C_1 and C_2 . The approach relies on finding a distortion of crystal C_2 such that, after deformation, the atoms of C_2 coincide exactly with sites of C_1 . In this context, we refer to C_1 as the parent crystal and C_2 as the child crystal. There are many possible ways to deform C_2 such that it maps onto C_1 . We denote by \mathcal{M} the set of all possible ways to relate C_2 to C_1 and refer to a particular element \mathcal{M}_m as a mapping of C_2 onto C_1 . Different mappings \mathcal{M}_m will require different degrees of deformation.

Crystal structures are represented by lattice vectors and a list of coordinates that correspond to the sites in the unit cell. We may compactly specify the lattice vectors as columns of a 3×3 matrix $\mathbf{L} = [l_1, l_2, l_3]$. The fractional (f) and cartesian (\vec{r}) coordinates of an arbitrary point in space are related to each other as $\vec{r} = \mathbf{L}f$.

A particular mapping of C_2 onto C_1 comprises two separate operations. First, in the lattice-mapping step, C_2 is rigidly rotated and homogeneously deformed until a shared lattice is obtained that can be used to define both structures simultaneously. Second, in the atomic mapping step, each atom of C_2 is assigned a companion in C_1 and displaced so that it is coincident with its companion. In the next section we describe how the strain and displacement fields that relate these crystal structures may be obtained.

Lattice mapping

We will first consider the mapping of the lattice of C_2 , denoted \mathbf{L}_2 , onto the lattice of C_1 , denoted \mathbf{L}_1 . Our lattice-mapping approach is similar to that of Trinkle et al.²⁴. In the simplest case, the two sets of lattice vectors are related via matrix multiplication according to

$$\mathbf{L}_1 = \mathbf{F} \mathbf{L}_2 \quad (1)$$

$$= \mathbf{V} \mathbf{Q} \mathbf{L}_2. \quad (2)$$

The deformation gradient tensor, \mathbf{F} , which relates the two lattices, can be uniquely decomposed as the product of a reorientation matrix, \mathbf{Q} , and the left stretch tensor, \mathbf{V} , which is symmetric and has positive determinant. We define a strain tensor in terms of the left stretch tensor as

$$\mathbf{B} = \mathbf{V} - \mathbf{I}, \quad (3)$$

where \mathbf{I} is the 3×3 identity matrix. The strain tensor \mathbf{B} (sometimes called the Biot strain) is also symmetric. At small deformation, the left stretch tensor is close in value to the identity matrix, and \mathbf{B} approaches zero. We note that the application of a rotation followed by a left stretch tensor to the child lattice until it maps onto the parent lattice is equivalent to an application of a right stretch tensor followed by a rotation to the parent lattice.

The matrix \mathbf{L}_1 is one of many matrices that generates the same lattice of crystal sites in the parent crystal. Any other matrix $\mathbf{L}_1^{(\mathbf{N})} = \mathbf{L}_1 \mathbf{N}$ generates the same lattice as \mathbf{L}_1 if \mathbf{N} is an integer matrix having determinant ± 1 . Such matrices are referred to as unimodular matrices. A more general equation relating \mathbf{L}_2 to \mathbf{L}_1 is thus

$$\mathbf{L}_1 \mathbf{N} = \mathbf{V}^{(\mathbf{N})} \mathbf{Q}^{(\mathbf{N})} \mathbf{L}_2. \quad (4)$$

Given \mathbf{L}_1 and \mathbf{L}_2 , there are many possible lattice mappings, corresponding to different choices of \mathbf{N} , and each results in a distinct left stretch tensor and reorientation matrix. Thus, $\mathbf{B}^{(\mathbf{N})}$ and $\mathbf{Q}^{(\mathbf{N})}$ can both be considered as functions of the integer matrix \mathbf{N} . This is illustrated in Fig. 1 in two dimensions where the parent C_1 has a rectangular unit cell and where the child crystal C_2 is simply a rotated version of C_1 . Two possible mappings are shown. In the first mapping, the short axis of C_2 is mapped onto the short axis of C_1 . In this case, the mapping involves only a rigid rotation with a zero strain cost. A unit sphere in C_1 remains a unit sphere in C_2 . In the second mapping, the long axis of C_2 is mapped onto the short axis of C_1 . This mapping requires not only a rotation, but also a deformation, as illustrated by the transformation of the circle in C_1 into an ellipsoid in C_2 .

Right-multiplying \mathbf{L}_1 by a different integer matrix, \mathbf{T}_1 , whose determinant has a magnitude greater than one would specify a supercell of C_1 . For example, a determinant-three matrix would specify a supercell having a volume three times that of the primitive cell, \mathbf{L}_1 , and contain three times as many basis atoms. By considering supercells of C_1 with lattice vectors $\mathbf{S}_1 = \mathbf{L}_1 \mathbf{T}_1$, it becomes possible to identify mappings in cases when the number of basis atoms in the primitive cell of C_2 is an integer multiple of the number of basis atoms in the primitive cell of C_1 . The integer matrix \mathbf{T}_1 with determinant greater than one is referred to as the supercell transformation matrix. The lattice-mapping relationship of Eq. (4) can be generalized by allowing for mappings onto supercells of C_1 according to

$$\mathbf{S}_1 \mathbf{N} = \mathbf{L}_1 \mathbf{T}_1 \mathbf{N} = \mathbf{V}^{(\mathbf{N})} \mathbf{Q}^{(\mathbf{N})} \mathbf{L}_2. \quad (5)$$

The integer volume of the supercell, defined as the ratio of the volume of the supercell \mathbf{S}_1 to the volume of \mathbf{L}_1 is equal to $\det(\mathbf{T}_1)$.

The problem of finding lattice deformations that link two periodic crystals can thus be reduced by Eq. (5) to counting over all possible integer and unimodular transformation matrices and recording the deformation and rotation tensors that link the two structures. This process, while computationally challenging, will provide an exhaustive list of deformations connecting the two periodic unit cells. However, as detailed in the next section, the symmetry of the underlying structures can be exploited to greatly simplify this process.

Accounting for symmetry during lattice mappings

The point group symmetry operations that leave the parent or child structures unchanged result in mappings between their lattices that are equivalent. In this section we describe how

symmetrically equivalent maps can be avoided while enumerating all possible lattice deformations that relate two structures.

First, we only count over symmetrically distinct supercell lattices of the parent crystal at each integer supercell volume. For a particular supercell, \mathbf{S}_1 , there are many possible representations of its lattice vectors, each of which is generated by a particular unimodular matrix \mathbf{N} , as is the case for the primitive cell lattice vectors. To enumerate distinct supercells, we employ a widely used approach based on Hermite Normal Forms of integer matrices⁴⁰.

Once the symmetrically unique supercells of the parent lattice have been identified, we can proceed with Eq. (5) by counting over all unimodular matrices and recording the deformation tensor that links the parent and child lattices:

$$\mathbf{S}_1 \mathbf{N} = \mathbf{F}^{(N)} \mathbf{L}_2 \quad (6)$$

We can reduce the number of unimodular matrices \mathbf{N} that need to be considered by only using those that generate symmetrically distinct transformations. To determine the symmetrically unique unimodular matrices, we use a convenient identity relationship between different representations of the same symmetry operation. Let us denote the crystal point group symmetry operations that leave the supercell and child lattices unchanged as $\mathcal{G}^{(1)}$ and $\mathcal{G}^{(2)}$, respectively. These symmetry operations may be represented in their cartesian form, which we denote as \mathcal{G} , or in their fractional form, which we denote as \mathcal{J} . The cartesian and fractional representations are related according to:

$$\mathcal{J} = \mathbf{L}^{-1} \mathcal{G} \mathbf{L} \quad (7)$$

Cartesian symmetry operations are applied to the lattice by left multiplication while the fractional symmetry operations are multiplied to the right of lattice vectors. The cartesian symmetry representation is unitary such that its inverse is equal to its transpose (i.e., $\mathcal{G}^{-1} = \mathcal{G}^T$). The fractional symmetry representation and its inverse are both unimodular integer matrices. Simultaneously applying the cartesian symmetry operation and the corresponding inverse fractional operation leaves the underlying lattice unchanged. For example,

$$\mathbf{S}_1 = \mathcal{G}^{(1)} \mathbf{S}_1 (\mathcal{J}^{(1)})^{-1}. \quad (8)$$

Invariance relationships of this type for \mathbf{S}_1 and \mathbf{L}_2 can be inserted into Eq. (6) to yield

$$\mathcal{G}^{(1)} \mathbf{S}_1 (\mathcal{J}^{(1)})^{-1} \mathbf{N} = \mathbf{F}^{(N)} \mathcal{G}^{(2)} \mathbf{L}_2 (\mathcal{J}^{(2)})^{-1}. \quad (9)$$

Upon rearranging, this simplifies to

$$\mathbf{S}_1 (\mathcal{J}^{(1)})^{-1} \mathbf{N} \mathcal{J}^{(2)} = (\mathcal{G}^{(1)})^{-1} \mathbf{F}^{(N)} \mathcal{G}^{(2)} \mathbf{L}_2 \quad (10)$$

$$\mathbf{S}_1 \mathbf{M} = \mathbf{F}^{(M)} \mathbf{L}_2 \quad (11)$$

where $\mathbf{M} = (\mathcal{J}^{(1)})^{-1} \mathbf{N} \mathcal{J}^{(2)}$ and $\mathbf{F}^{(M)} = (\mathcal{G}^{(1)})^{-1} \mathbf{F}^{(N)} \mathcal{G}^{(2)}$ are symmetrically equivalent unimodular and deformation matrices. Counting over such deformation tensors that are related under the symmetries of the parent or child structures may be avoided by only using one of the symmetrically related unimodular matrices. Thus for each distinct supercell of \mathcal{C}_1 we only count over symmetrically unique unimodular matrices when recording the set of deformation tensors that relate \mathcal{C}_1 and \mathcal{C}_2 .

Atomic maps

If the primitive cell of \mathcal{C}_1 or \mathcal{C}_2 contains more than one basis atom, the atoms in \mathcal{C}_2 may need to be repositioned inhomogeneously to coincide with sites of \mathcal{C}_1 . This deformation of the atomic positions can be represented by a displacement vector for each atom in the unit cell of \mathcal{C}_2 . We denote the site positions in the unit cell of \mathcal{C}_1 as

$\{\vec{r}_1^{(1)}, \dots, \vec{r}_{n_A}^{(1)}\}$ and the atom positions in the unit cell of \mathcal{C}_2 as $\{\vec{r}_1^{(2)}, \dots, \vec{r}_{n_A}^{(2)}\}$. All positions are in Cartesian coordinates. By convention, we measure displacements by first deforming \mathcal{C}_2 via a particular lattice mapping, making its lattice sites coincident with lattice sites of \mathcal{C}_1 . Then, we rigidly translate the basis of \mathcal{C}_2 by a vector \vec{t} to bring the basis atoms of \mathcal{C}_2 into closer registration with the sites of \mathcal{C}_1 . An atomic displacement is then the vector that translates a particular basis atom j of \mathcal{C}_2 to the position of atom i of \mathcal{C}_1 . Given a particular rigid translation \vec{t} , the displacement vector that translates atom j of \mathcal{C}_2 onto site i of \mathcal{C}_1 is

$$\vec{d}_{ij}^{(1)}(\vec{t}) = \lfloor \vec{r}_i^{(1)} - \mathbf{V}^{(\mathbf{N}, \mathbf{T}_1)} \mathbf{Q}^{(\mathbf{N}, \mathbf{T}_1)} \vec{r}_j^{(2)} - \vec{t} \rfloor_{\mathbf{S}_1}, \quad (12)$$

where the operation $\lfloor \cdot \rfloor_{\mathbf{S}_1}$ selects the shortest possible displacement vector with respect to periodic boundary conditions by considering all possible lattice translations of the lattice \mathbf{S}_1 . Equation (12) demonstrates how the position of an atom in \mathcal{C}_2 must first be reoriented and deformed, according to the lattice mapping, before relating it to an atom in \mathcal{C}_1 via translation. The superscript of $\vec{d}_{ij}^{(1)}$ indicates that it is the displacement measured in the strain state of \mathcal{C}_1 .

There are $n_A!$ ways to form (i, j) pairs between sites of \mathcal{C}_1 and \mathcal{C}_2 . We express a particular one-to-one assignment of atoms in \mathcal{C}_2 to sites in \mathcal{C}_1 as the permutation \hat{p} , such that $(i, \hat{p}[i])$ is an assignment of atom $\hat{p}[i]$ in \mathcal{C}_2 to site i in \mathcal{C}_1 .

Geometric mapping cost

As shown in the previous sections, two arbitrary crystal structures can be related to each other geometrically in a multitude of ways. The set of strains relating the lattices of the two structures are infinite, and for each possible strain map there are several ways to displace the atoms to bring the two structures into perfect coincidence. In this section and the next section we describe how geometry and symmetry may be used to define useful “cost metrics” that can capture the complexity of every enumerated transformation pathway.

Equations (3) and (5) describe the relationship between two lattices through a symmetric left stretch tensor (\mathbf{V}) and the Biot strain tensor (\mathbf{B}). These tensors describe changes in both the volume and shape of crystal \mathcal{C}_2 . Volume change alone is a poor measure of crystal structure similarity since volume depends strongly on chemistry, temperature, and other considerations. The volumetric dependence of the left stretch tensor is given simply by $\det \mathbf{V} = \mathcal{V}_{L_2} / \mathcal{V}_{L_1}$. Consequently, we can renormalize the left stretch tensor to make it independent of volume changes, yielding the volume-normalized stretch tensor

$$\tilde{\mathbf{V}} = \frac{1}{\det(\mathbf{V})^{1/3}} \mathbf{V} \quad (13)$$

and the corresponding volume-normalized strain tensor

$$\tilde{\mathbf{B}} = \tilde{\mathbf{V}} - \mathbf{I}. \quad (14)$$

We use the volume-normalized strain to define cost functions of lattice deformation in order to isolate shape-change effects from volume-change effects.

We now introduce a lattice-deformation cost function that measures the magnitude of lattice strain. In general, the Biot strain tensor in Eq. (14) depends on the orientation of the coordinate system used to define the lattice vectors of \mathcal{C}_1 and \mathcal{C}_2 . However, the lattice-deformation cost function should be invariant to changes in coordinate system. A common metric of tensor magnitude that is invariant to coordinate system is the Frobenius

norm:

$$\begin{aligned} \mathbf{B} : \mathbf{B} &= \sum_{ij} B_{ij}^2, \\ &= \text{tr}(\mathbf{B}^2). \end{aligned} \quad (15)$$

To demonstrate that the Frobenius norm of \mathbf{B} is a physically meaningful metric of deformation, we consider a uniform sphere of radius ρ as it is deformed by the volume-normalized stretch tensor $\tilde{\mathbf{V}}$. The displacement of a point on the sphere is

$$\delta \vec{\rho} = \tilde{\mathbf{V}} \vec{\rho} - \vec{\rho} = \tilde{\mathbf{B}} \vec{\rho}, \quad (16)$$

and the mean-squared displacement over the surface of the sphere is the integral

$$\text{avg}(|\delta \vec{\rho}|^2) = \frac{1}{4\pi\rho^2} \int_S \rho^2 d\omega (\vec{\rho}^\top \tilde{\mathbf{B}}^2 \vec{\rho}), \quad (17)$$

taken over the domain of solid angle in three dimensions. Expanding the integrand term by term and integrating yields

$$\text{avg}(|\delta \vec{\rho}|^2) = \frac{1}{3} \rho^2 \text{tr}(\tilde{\mathbf{B}}^2). \quad (18)$$

The spherical radius ρ can be set to a physically meaningful parameter, such as the Wigner-Seitz radius of the child crystal,

$$\rho_2^{(WS)} = \left(\frac{3}{4}\Omega_2\right)^{1/3}, \quad (19)$$

to obtain a per-atom metric of lattice deformation. However, Eq. (18) can also be normalized by ρ^2 to obtain a scale-invariant metric of deformation, which is how we shall specify the lattice-deformation cost function:

$$c_L(\mathcal{M}_m) = \frac{1}{3} \text{tr}(\tilde{\mathbf{B}}^2). \quad (20)$$

Measuring the degree of atomic distortion is simpler than measuring the lattice strain. We measure the quality of a particular atomic assignment \hat{p} with the mean-squared atomic displacement:

$$c_A(\mathcal{M}_m) = \frac{1}{\rho_1^2} \left(\frac{1}{n_A} \sum_i \left| \vec{d}_{i,\hat{p}[i]}^{(1)}(\vec{t}) \right|^2 \right) \quad (21)$$

As in the case of the lattice-deformation cost function, we normalize the atomic-deformation cost by the squared Wigner-Seitz radius $\rho_1 = (3\Omega_1/4)^{1/3}$ to obtain a scale-invariant cost. The cost function Eq. (21) is a function of both the rigid translation \vec{t} and atomic assignment \hat{p} , and we will show that it can be minimized with respect to these two parameters. It is also implicitly, via Eq. (12), a function of the integer matrices \mathbf{N} and \mathbf{T}_1 , which determine the lattice mapping.

The lattice-deformation cost function $c_L(\mathcal{M}_m)$ and the atomic-deformation cost function $c_A(\mathcal{M}_m)$ measure qualitatively distinct aspects of crystal deformation. We define mappings to be “optimal” if they simultaneously minimize both the lattice and atomic-deformation costs. Specifically, we define the following scalar total cost function:

$$c(\mathcal{M}_m, w_L) = w_L c_L(\mathcal{M}_m) + (1 - w_L) c_A(\mathcal{M}_m), \quad (22)$$

which is a weighted average of the atomic and lattice-deformation cost functions specified by a lattice weight parameter w_L . The presence of two cost functions (lattice and atomic costs) leads to the well-known problem in multi-objective optimization where there is usually no unambiguous ‘optimal’ mapping (The exception to this is when \mathcal{C}_1 and \mathcal{C}_2 are structurally identical, in which case some mappings have both zero lattice deformation and zero atomic deformation). By introducing the total cost function that is related to both these costs, the minima is well-defined. Because the atom- and lattice-deformation cost functions defined in Eqs. (20),(21) are both scaled to remove volumetric

dependence and are normalized per atom, we anticipate that a 50/50 weighted average, corresponding to $w_L = 0.5$ should be appropriate for many applications. Values of $w_L > 0.5$ increase the relative penalty of lattice deformation (thus allowing atomic displacements to be relatively larger) while values of $w_L < 0.5$ decrease the relative penalty of lattice deformation (thus biasing atomic displacements to remain small).

Symmetry-adapted mapping cost

Crystal structures that differ by more than just their volumes can still be very similar. For instance, the hexagonal close-packed (hcp) structure may be deformed to have any $\frac{c}{a}$ ratio without qualitatively changing the crystal. Hcp structures that differ with respect to their $\frac{c}{a}$ ratio will have a nonzero geometric cost despite being crystallographically identical. To enable the easy comparison of such structures we define a “symmetry-adapted” mapping cost that is invariant under deformations that preserve the symmetry of the parent crystal.

The symmetry-adapted cost is obtained by removing any symmetry-preserving deformations or displacements from the mapping field. The symmetry-preserving Biot strain tensor, \mathbf{B}_{sym} , can be obtained by averaging the effect of each point group operation on the calculated stretch tensor obtained from Eq. (5):

$$\mathbf{B}_{\text{sym}} = \frac{1}{N_{\mathcal{G}_2}} \sum_{\mathcal{G}_2} \mathcal{G}_2 \mathbf{B} \mathcal{G}_2^\top \quad (23)$$

where \mathcal{G}_2 is the point group of the child structure (\mathcal{C}_2) and $N_{\mathcal{G}_2}$ is the number of operations in the point group. Formally this corresponds to applying the Reynolds operator of the point group to the Biot strain tensor. Obtaining the average over all possible symmetrically equivalent Biot strain tensors leaves us with only that part of the deformation that is invariant under symmetry. Examples of symmetry-preserving deformations include simple hydrostatic deformations of the unit cells and changes to the $\frac{c}{a}$ ratio in hcp.

The symmetry-adapted mapping score is computed using the symmetry-breaking strain tensor defined as

$$\mathbf{B}_{\text{sym-break}} = \mathbf{B} - \mathbf{B}_{\text{sym}} \quad (24)$$

The symmetry-adapted lattice cost is then calculated as:

$$c_L(\mathcal{M}_m) = \frac{1}{3} \text{tr}(\mathbf{B}_{\text{sym-break}}^2) \quad (25)$$

The symmetry preserving score relative to the parent may be calculated by utilizing the stretch tensor relative to the parent and the point group operations of the parent.

The symmetry-adapted atomic cost can be obtained in a manner similar to that described for the lattice-deformation cost. First, we obtain the collective atomic displacements that are invariant under the factor group of the parent structure by again applying the Reynolds operator. Subsequently, the displacements Eq. (12) that map the atoms of the child onto the atoms of the parent can be projected on to the symmetry-invariant collective displacements of the parent and removed to estimate the symmetry-breaking displacements. To perform these steps it is convenient to collect atomic displacements relative to the parent crystal atomic positions as a vector containing $3n$ entries

$$\vec{D} = \begin{bmatrix} d_1^x \\ d_1^y \\ d_1^z \\ \vdots \\ d_n^x \\ d_n^y \\ d_n^z \end{bmatrix} \quad (26)$$

These atomic shuffles can be transformed by symmetry operations in the factor group of the parent crystal according to

$$\vec{D} = \mathcal{G}_1 \vec{D} \quad (27)$$

where \mathcal{G}_1 is now a $3n \times 3n$ matrix representation of a symmetry operation in the factor group of the parent structure that acts on the unrolled displacement vector \vec{D} . The $3n \times 3n$ matrix representations can be constructed from the 3×3 Cartesian representation and the permutation representation of the factor group operations.

Symmetry-invariant modes can be constructed by applying the Reynolds operator to the set of basis vectors (denoted \mathbf{I} , the $3n \times 3n$ identity matrix) that span the full $3n \times 3n$ space^{15,41}:

$$\langle \mathbf{D} \rangle = \frac{1}{N_S} \sum_S \mathcal{S}_i \mathbf{I} \quad (28)$$

where $\langle \mathbf{D} \rangle$ is the symmetry-averaged basis vectors. Each column of the $\langle \mathbf{D} \rangle$ matrix corresponds to a collective displacement field that is invariant to the symmetry of the parent crystal. Not all columns are independent from each other and a set of spanning vectors may be identified by performing singular value decomposition (SVD) of the matrix

$$\langle \mathbf{D} \rangle = \mathbf{W} \mathbf{\Sigma} \mathbf{X}^T \quad (29)$$

The dimensionality (r) of the symmetry-invariant space is equal to the number of nonzero singular values (or diagonal elements) in $\mathbf{\Sigma}$, while a set of orthogonal vectors that span this space can be derived from the r column vectors of \mathbf{W} corresponding to the nonzero singular values. This space of symmetry-preserving distortions can be expanded into larger supercells, and their symmetry-breaking deformation fields can then be obtained as:

$$\vec{D}_{\hat{p}, \text{sym-break}}(\vec{t}) = \vec{D}_{\hat{p}}(\vec{t}) - \sum_i (\vec{D}_{\hat{p}}(\vec{t})^T \vec{D}_{\text{sym}}^i) \vec{D}_{\text{sym}}^i \quad (30)$$

where \vec{D}_{sym}^i is a matrix of symmetry-invariant collective atomic modes calculated from Eq. (29) and expanded so that it accounts for the total number of atoms in the supercell of \mathcal{C}_1 that is being considered. \vec{D}_{sym}^i refers to the i th collective displacement mode in this matrix.

The symmetry-breaking cost is then obtained as:

$$c_A(\mathcal{M}_m) = \frac{1}{\rho_1^2} \left(\frac{1}{n_A} \left| \vec{D}_{\hat{p}, \text{sym-break}}(\vec{t}) \right|^2 \right) \quad (31)$$

A summary of the mapping algorithm

The mapping algorithm described above is schematically illustrated in Fig. 2. The figure shows the steps involved in mapping a two dimensional honeycomb arrangement of atoms (labeled \mathcal{C}_2) on to a parent square lattice (labeled \mathcal{C}_1). First the algorithm counts over all symmetrically distinct supercells of the parent square lattice that contains two atoms. The first lattice map, labeled \mathbf{F}_1 , involves a large shear and tetragonal deformation of the child lattice. The second lattice map (\mathbf{F}_2) relates the child structure to an equivalent but geometrically different supercell. The lattice undergoes a tetragonal deformation and a smaller shear distortion. Finally, \mathbf{F}_3 relates the child structure to a supercell of the square lattice that is different from those depicted in the first two deformations. Having enumerated some of the strain deformations, the algorithm then deforms the child structure and rigidly translates the atoms such that at least one of the atoms is in perfect coincidence with the lattice points of the parent structure. Figure 2 illustrates the effect of applying the homogeneous strain distortion and rigid translation to the child structure across all three lattice maps. The other atom then needs to be displaced along the highlighted vector for each of the three maps to bring it into perfect coincidence with the parent

structure. Having recovered the parent crystal structure from the honeycomb child structure, the algorithm estimates the scalar costs associated with each deformation and displacement through Eqs. (20),(21),(31),(25).

Finding optimal maps

In the previous sections we outlined how a set of mappings \mathcal{M} linking any two arbitrary crystals can be generated. First a set of strains linking the crystals is generated by counting over all symmetrically distinct lattices. For each strain map, a set of $n!$ atomic maps can be generated by assigning and displacing every atom in the child so that it comes into perfect coincidence with its counterpart in the parent. We also defined two ways of quantifying the extent of distortion in each of these maps through cost functions. The first, which we refer to as the geometric cost calculates a scalar quantity for the strain and atomic deformations in each map \mathcal{M}_m . The second, called symmetry-adapted cost, uses the symmetry of the underlying parent crystal structure to calculate these costs over only the symmetry-breaking parts of the strain and displacement fields. In both cases we define a total cost function that combines the individual contributions from the strain and atomic distortions to calculate a single scalar mapping cost for every mapping in \mathcal{M} . In this section we discuss optimization algorithms that can be leveraged to quickly and efficiently obtain the map with the lowest total mapping cost and an algorithm to systematically traverse the multitude of mappings in increasing order of their mapping costs

The lattice-deformation cost function for a particular supercell transformation \mathbf{T}_1 can be optimized simply by counting over unimodular matrices \mathbf{N} in Eq. (5). Some of the simplest examples of \mathbf{N} matrices describe operations such as a swap or permutation of lattice vectors, or the replacement of a lattice vector by its negative direction. Although \mathbf{N} must always have a determinant of ± 1 , there is no limit on the magnitude of its coefficients. However, if \mathbf{N} includes coefficients of very large magnitude, the lattice vectors $\mathbf{S}'_i = \mathbf{S}_i \mathbf{N}$ that define the lattice become increasingly elongated, and favorable mappings become unlikely. To minimize the range of values of \mathbf{N} that must be considered, we first obtain the reduced cells of \mathbf{S}_1 and \mathbf{L}_2 using, for example, the Lenstra–Lenstra–Lovász algorithm⁴². The reduced cells are maximally compact, such that their lattice vectors have the minimal possible lengths and the angles between lattice vectors are as close to 90° as possible. The reduced cell is not unique, although conventions do exist, such as the Niggli reduced cell, that provide a unique definition, to within a rigid rotation. If both $\mathbf{L}^{(2)}$ and $\mathbf{S}^{(1)}$ are in reduced form, we need only consider lattice-equivalent transforms that have small entries, e.g., $\{-1, 0, 1\}$, or $\{-2, \dots, 2\}$. Once a range for the elements of \mathbf{N} has been decided upon, it is straightforward to count first over distinct supercells $\{\mathbf{T}_1\}$ and then over unimodular matrices $\{\mathbf{N}\}$ within the specified range and record the lattice-deformation cost for each $(\mathbf{T}_1, \mathbf{N})$. We find it useful for many cases to simply retain all such mapping relationships, sorted by their weighted lattice-deformation cost or to find the best or K -best lattice-mapping relationships. The algorithm remains unchanged when using the symmetry-adapted lattice costs as defined by eq. (25).

A total mapping of \mathcal{C}_2 onto \mathcal{C}_1 comprises the ordered tuple $\mathcal{M}_m = (\mathbf{T}_1, \mathbf{N}, \vec{t}, \hat{p})$, where \mathbf{T}_1 and \mathbf{N} are integer matrices specifying the lattice mapping, \vec{t} is a rigid translation of crystal \mathcal{C}_2 relative to \mathcal{C}_1 , and \hat{p} is a permutation specifying the atomic mapping. The optimal atomic mapping for a given lattice mapping (i.e., \mathbf{N} and \mathbf{T}_1) is one that minimizes Eq. (21) with respect to both \vec{t} and \hat{p} . The rigid translation that minimizes Eq. (21) for a given permutation operation, \hat{p} , is simply the translation that yields zero average displacement of the atoms in \mathcal{C}_2

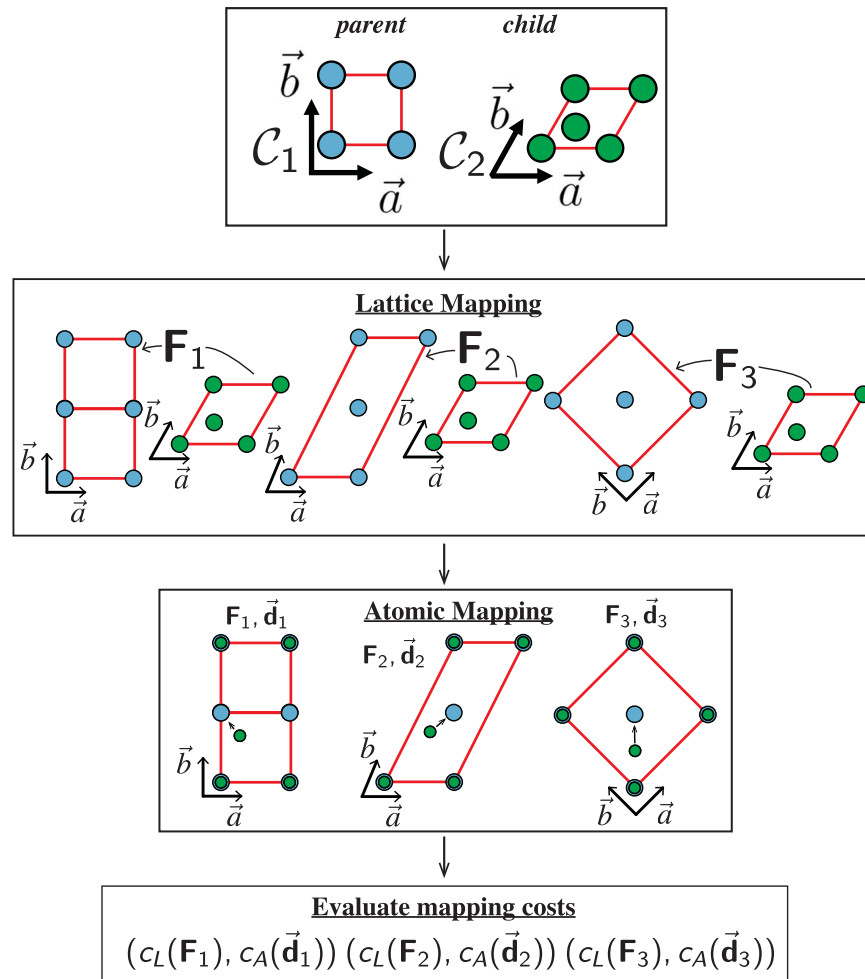


Fig. 2 Schematic of the mapping algorithm. A summary of the steps involved in generating a map between a parent structure with a square lattice and a child structure with the atoms arranged in a honeycomb pattern. Three lattice and atomic maps that relate these structures are illustrated, with the final step being the evaluation of the lattice and atomic cost functions.

(if nonzero, the average displacement amounts to an overall ‘drift’ of C_2 relative to C_1). The permutation operation \hat{p} , in contrast, is a discrete variable—the minimization of Eq. (21) with respect to \hat{p} for a particular choice of \vec{t} is a discrete combinatorial optimization problem. Fortunately, since Eq. (21) is a simple linear combination of contributions from $(i, \hat{p}[i])$ pairs, this optimization is classified as a linear assignment problem, for which well-known solution methods exist⁴³.

Because the atomic-deformation cost function in Eq. (21) depends on both the discrete assignment \hat{p} and the continuous translation vector \vec{t} , we propose an iterative procedure to optimize the atomic-deformation cost. In the first step, we identify potential rigid translations of C_2 that are likely to be near the optimal translation. These are translations that bring an atom in C_2 into coincidence with a chemically compatible site of C_1 . It is preferable at this step to pick an atom from among the minority species. For example, if C_1 and C_2 each have two Cu atoms and four Au atoms per unit cell, we would pick as candidate translation vectors \vec{t}_κ those that map one Cu atom of C_2 onto each of the two Cu atoms in C_1 . In general, we denote the number of minority species in C_2 as m_s .

Next, we find the optimal atomic assignment for each attempted translation \vec{t}_κ , $\kappa = 1, \dots, n_{ms}$ where n_{ms} is the number of minority species. To do so, we construct the atomic assignment cost matrix, $\mathbf{C}(\vec{t}_\kappa)$, for each rigid translation. The (i, j) element of the cost matrix is the pairwise cost contribution of assigning site j

in C_2 to site i in C_1 for the rigid translation \vec{t}_κ . The elements of the cost matrix are

$$C_{ij}(\vec{t}_\kappa) = \begin{cases} \frac{1}{n_A} \left| \vec{d}_{ij}^{(1)}(\vec{t}_\kappa) \right|^2 & \text{if } s_i = s_j \\ \infty & \text{if } s_i \neq s_j \end{cases} \quad (32)$$

where an element of the cost matrix is infinite (or at least very large) when the species at sites i and j , labeled s_i and s_j , respectively, are chemically incompatible. This ensures that chemically incompatible assignments are excluded from the optimal assignment. Once the cost matrix is constructed according to eq. (32), the optimal assignment \hat{p}_κ^* is the solution to

$$\hat{p}_\kappa^* = \arg \min_{\hat{p}} \sum_i C_{i, \hat{p}[i]}(\vec{t}_\kappa). \quad (33)$$

The minimization of Eq. (33) can be solved by the Hungarian algorithm⁴⁴ (used in this work) or via linear programming methods⁴³.

Once the optimal assignment is found for an attempted translation, \vec{t}_κ , we update the translation by subtracting the average displacement (i.e., drift) due to that assignment \hat{p}_κ^* according to

$$\vec{t}_\kappa = \vec{t}_\kappa - \frac{1}{n_A} \sum_i \vec{d}_{i, \hat{p}_\kappa^*[i]}^{(1)}(\vec{t}_\kappa). \quad (34)$$

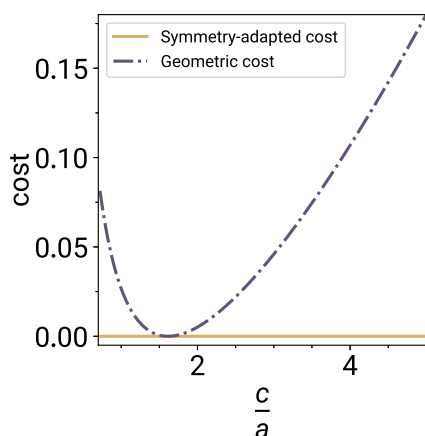


Fig. 3 Geometric cost versus symmetry-adapted cost. A comparison of the symmetry-adapted cost with the geometric cost when mapping an hcp crystal structure with varying c/a ratio onto a reference hcp crystal with a c/a ratio of ≈ 1.6 .

Next, we calculate the atomic-deformation cost, eq. (21), for the assignment/translation pair $(\hat{p}_\kappa^*, \vec{t}_\kappa)$ and record the result. This must be performed for each attempted translation $\kappa = 1, \dots, n_{ms}$. The optimal atomic assignment \hat{p}_κ^* for the imposed lattice map is the one with the lowest atomic-deformation cost.

In certain extreme cases, it may be necessary to perform additional optimization steps. When calculating the revised atomic displacements due to the augmented translation \vec{t}_κ , it should be verified that the average total displacement (i.e., drift) is indeed equal to zero, i.e.,

$$\frac{1}{n_A} \sum_i \vec{d}_{i, \hat{p}_\kappa^*[i]}^{(1)}(\vec{t}_\kappa) = 0 \quad (35)$$

Violations of Eq. (35) may occur due to the periodic boundary considerations imposed in Eq. (12). If an outlier atom has a very large atomic displacement under translation \vec{t}_κ , it may indicate that atom $\hat{p}[i]$ in C_2 is nearly midway between two periodic images of site i in C_1 . In such a case, a small change in rigid translation may affect a large change in the displacement of the atom (e.g., the revised translation swaps the first- and second-nearest periodic image of atom i). If a violation of Eq. (35) is encountered, it indicates that the optimal atomic assignment may be unstable relative to small rigid translations in the vicinity of \vec{t}_κ . As such, the augmented translation \vec{t}_κ may be recalculated iteratively until Eq. (35) is satisfied, with the augmented translation at each iteration added to the list of plausible translations to be considered for solution of the linear assignment problem.

Minimizing the symmetry-adapted atomic cost requires additional steps. Unlike the geometric-cost function of eq. (21), the symmetry-adapted atomic cost can only be calculated once a particular assignment of atoms has been chosen. As a result the Hungarian algorithm cannot be directly applied to obtain the minimum cost assignment. The algorithm can, however, be adapted to systematically search over suboptimal maps to arrive at the assignment of atoms that minimizes the symmetry-adapted cost. To do this, we adopt Murty's algorithm⁴⁵ to generate a queue of suboptimal maps that are arranged in increasing magnitude of their geometric cost (Eq. (21)). We then search this queue up to some maximum depth that is user specified and recalculate the symmetry-adapted cost for each of these elements. The element with the minimal symmetry-adapted cost is then recorded as the optimal map. This algorithm is guaranteed to provide the optimal map if all $n!$ assignments are enumerated. However in practice we have found that the assignment that minimizes the symmetry-adapted cost is often found by searching over the first 5–10 suboptimal assignments. A detailed algorithm outlining a

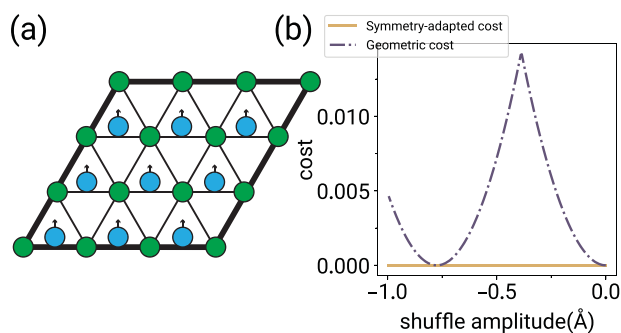


Fig. 4 Geometric cost versus symmetry-adapted cost. A comparison of the symmetry-adapted costs against the geometric costs when mapping the O' structure with varying shuffle amplitude onto a reference O' structure. **a** A schematic of the O' crystal structure. The arrows signify a shuffle mode of the triangular layer of blue atoms between rigid triangular layers of green atoms. **b** Comparison between the symmetry-adapted and geometric costs.

methodology to systematically enumerate all possible suboptimal maps is outlined in the supplementary information.

Case studies

We illustrate the mapping algorithm by first comparing the geometric-cost functions against the symmetry-adapted cost function for two parent structure prototypes. Next we demonstrate how well-known structural transformation pathways can be identified by systematically analyzing all possible mappings and their associated cost function values.

The hexagonal close-packed (hcp) crystal structure is usually characterized by two lattice parameters that are conventionally denoted a and c . The distance between the atoms in a particular triangular layer of hcp is equal to a while the spacing between the layers is half the c lattice parameter. The underlying crystal structure remains qualitatively unchanged across all values of a and c . It is desirable to use a mapping function that is insensitive to such changes in the crystal structure. Figure 3 compares the difference between the geometric and symmetry-adapted lattice costs for a child structure with a wide-range of $\frac{c}{a}$ ratios that is mapped on to a parent crystal with a $\frac{c}{a} \approx 1.6$. The geometric cost rises rapidly at values that deviate from the parents' lattice parameters. On the other hand the symmetry-adapted cost remains exactly zero across the full range of values.

Figure 4a shows the crystal structure of an orthorhombic phase, denoted O' , that is frequently formed in titanium-rich alloys⁴⁶. Similar to hcp, the O' crystal structure consists of triangular layers; however, they do not adopt the ideal stacking as in hcp as is illustrated in Fig. 4a. Furthermore, the unit cell of O' is slightly strained relative to that of hcp. As a result, the O' structure has a lower symmetry than hcp. We ignore the lattice strain for the purpose of demonstrating our algorithm. Figure 4b compares the geometric cost against the symmetry-adapted cost when mapping O' onto a reference O' structure as a function of the shuffle amplitude of alternating close-packed layers of O' (i.e., along the arrows in Fig. 4a). The geometric cost rapidly increases with the shuffle amplitude until it reaches a maximum before it starts decreasing. The decrease in geometric cost is due to the periodicity of the parent crystal structure. In contrast to the periodically varying geometric cost, the symmetry-adapted cost remains identically zero. The shuffle of alternating close-packed planes as shown in Fig. 4 do not result in further symmetry breaking of O' and thus do not incur a cost penalty. We note that as the layers are translated relative to each other in O' , a particular shuffle amplitude will result in the formation of perfect hcp. It is not possible to easily identify shuffles that increase the symmetry

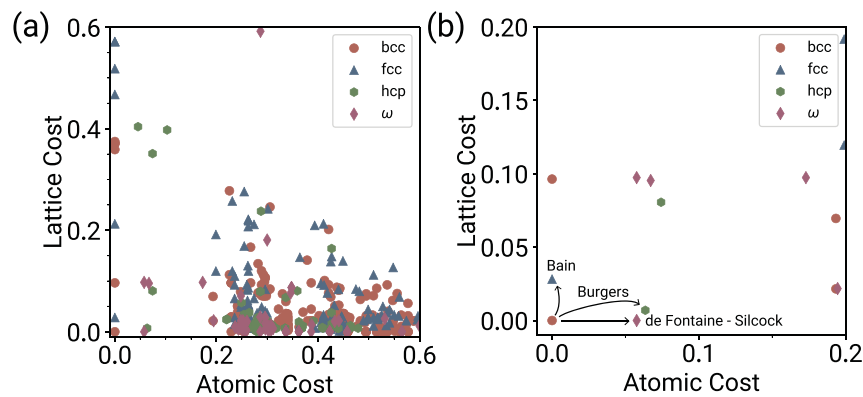


Fig. 5 Mapping costs of common structures onto bcc. The lattice and atomic costs when mapping bcc, fcc, hcp, and ω onto a reference bcc crystal structure. **a** Most mappings have large mapping costs. **b** Several well-known transformations, such as the Bain, Burgers, and de Fontaine-Silcock path, have smaller mapping costs.

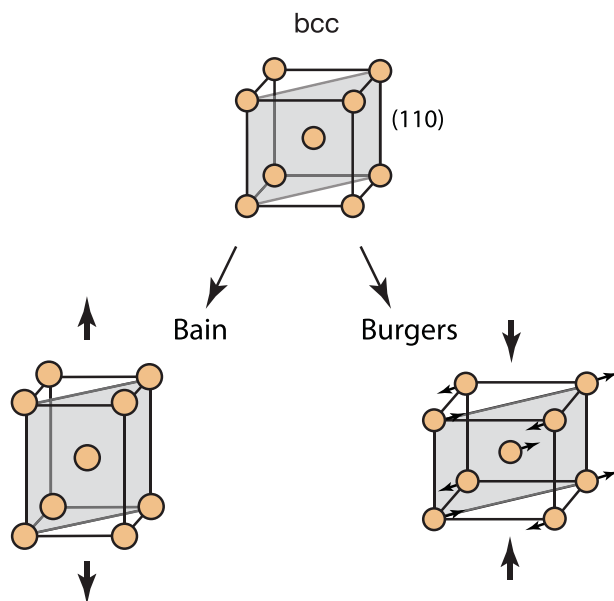


Fig. 6 The Bain and Burgers paths. The bcc crystal can be converted to fcc through a tetragonal strain known as the Bain path¹⁹ or it can be converted to hcp along the Burgers path, which combines a compressive tetragonal strain with an atomic shuffle²⁰.

of the underlying crystal with just the symmetry operations of the low-symmetry structure.

There is a large number of structural relationships between any pair of crystals. As an illustration, we explore mapping relationships between well-known crystal structures such as bcc, fcc, hcp, and ω . The ω phase can be described as alternating triangular and honeycomb layers, with the atoms of the triangular layers residing above and below the centers of the hexagons of the honeycomb layers. Figure 5 plots the lattice and atomic displacement costs for different mappings of fcc, hcp, and ω onto bcc. Three pathways connecting bcc with hcp, fcc, and ω are highlighted in Fig. 5b. Also included are different mappings of bcc onto bcc, which are denoted with a circle.

Several of the mappings with small mapping scores in Fig. 5 correspond to the end points of well-known pathways that convert one crystal structure into another. Among them are the Bain path that connects bcc to fcc^{19,26,27}, the Burgers path that connects bcc to hcp^{20,26,27} and a path that converts bcc into ω ^{21,22,27}. Each pathway is linked to a particular mapping relationship. The Bain path, illustrated in Fig. 6, converts bcc into fcc through a tetragonal tensile strain of the bcc unit cell vectors

along one of its cubic axes. The mapping corresponding to this path only has a lattice cost, but no atomic displacement cost. The resulting fcc crystal can be converted to another bcc crystal through a Bain path along a different fcc cubic axis. This leads to the mapping relationship between two bcc crystals denoted by the circle with a lattice cost of ≈ 0.1 in Fig. 5.

The Burgers path, which connects bcc to hcp, connects end states with a mapping that has both a lattice cost and an atomic displacement cost. As illustrated in Fig. 6, the Burgers path couples a compressive tetragonal distortion of bcc along one of its cubic axes, followed by a shuffle of (110) planes as referenced to the original bcc lattice. While the lattice cost of the Burgers path is smaller than that of the Bain path, it does have a sizable atomic displacement cost.

A third well-known path, due to Silcock²¹ and de Fontaine^{22,47}, connects bcc to ω . Early descriptions of the bcc to ω path invoked a transverse shuffle of (11 $\bar{2}$) planes along the (111) direction^{21,48}. Later, de Fontaine pointed out a more intuitive description in terms of longitudinal shuffles of (111) planes²². This is illustrated in Fig. 7, where every two out of three (111) planes of bcc collapse to form a honeycomb layer, thereby forming the ω crystal structure consisting of alternating honeycomb layers and triangular layers. The supercell of bcc must have an integer volume of three and has a hexagonal unit cell, with its \vec{A} and \vec{B} vectors spanning one family of (111) planes and its \vec{C} axis perpendicular to the same family of (111) planes of bcc (i.e., \vec{A} is parallel to $[\bar{1}01]$, \vec{B} is parallel to $[0\bar{1}1]$ and \vec{C} is parallel to $[111]$). As is evident in Fig. 5, the mapping between bcc and ω corresponding to the Silcock-de Fontaine path has a very small lattice cost but a large atomic displacement cost due to the collapse of adjacent (111) planes.

Figure 8 explores the structural relationship between hcp and common crystal structure prototypes such as bcc, fcc, ω , and a family of topologically close-packed phases. The Burgers transformation relating bcc and hcp requires a large atomic shuffle and a small lattice strain. The transformation pathway to the fcc structure with an atomic cost of ≈ 0.07 can be viewed as successively applying the Burgers and Bain transformation to hcp.

Structural relationships between hcp and the ω crystal structure have been extensively explored by Trinkle et al.^{23,24}. Our mapping algorithm shows that two of these transformation pathways, labeled TAO-1 and Silcock, have the smallest cost. The Silcock pathway primarily involves atomic shuffles within the basal plane such that the (0001) planes of hcp are transformed into the (11 $\bar{2}$ 0) planes of ω . The TAO-1 transformation pathway, which is predicted to have the lowest barrier in pure titanium²³, transforms the basal plane of hcp to (0 $\bar{1}$ 11) planes of ω . Both transformation pathways require significant shuffles but are accompanied by small strain distortions.

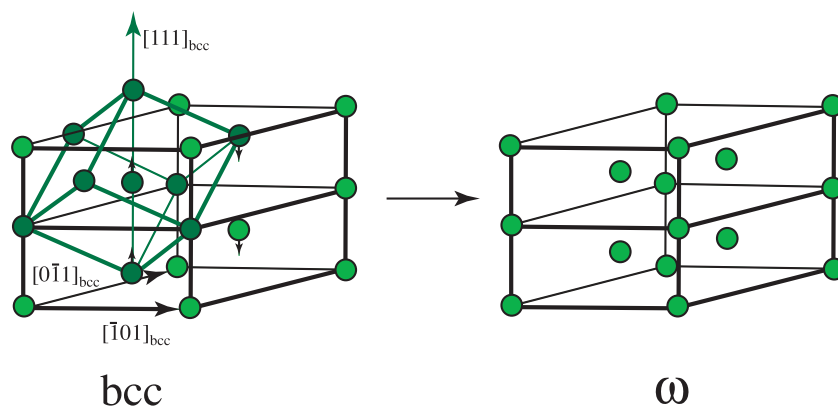


Fig. 7 The de Fontaine path between bcc and omega. The ω crystal structure can be formed from bcc by collapsing two out of every three bcc (111) planes²².

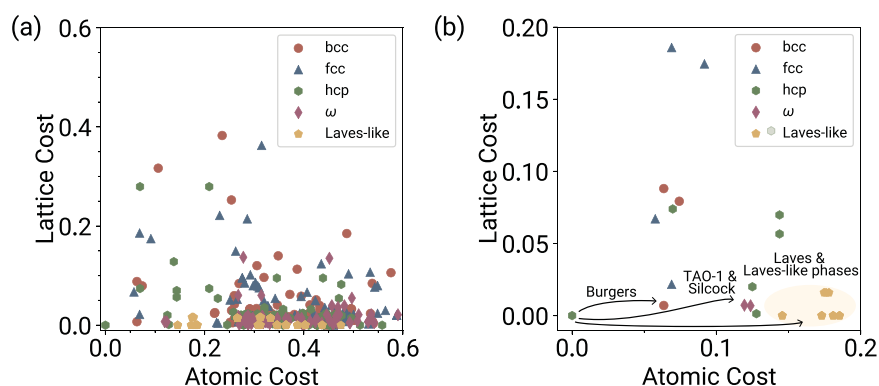


Fig. 8 Mapping costs of common structures onto hcp. Mapping costs of bcc, fcc, hcp, and ω onto a reference hcp crystal. **a** A large number of mappings have large atomic costs. **b** Several previously identified mappings^{20,23,25} have much smaller mapping costs.

Figure 8 also shows mapping costs for transformation pathways that link hcp to a family of topologically close-packed structures. These pathways, as described by Natarajan and Van der Ven²⁵, involve a transformation of a subset of the hcp close-packed triangular layers into kagome nets and a dissociation of other close-packed triangular layers into three new, less dense triangular layers. The formation of a kagome net from a triangular lattice can be realized with in-plane shuffles while the dissociation of triangular lattices into three new triangular lattices occurs with out-of-plane shuffles. A large number of topologically close-packed phases that include C14, C15, and C36 Laves phases, CaCu_5 , Be_3Nb , and Co_7Gd_2 can be generated from hcp through a combination of in-plane and out-of-plane shuffles²⁵.

DISCUSSION

We have described an approach to enumerate mappings between any pair of crystal structures. The approach first identifies symmetrically distinct lattice maps and accompanying strains that deform the lattice of the child structure onto that of the parent. For each lattice map, the basis atoms within the child structure are displaced until they coincide perfectly with the parent structure. As there are an infinite number of ways to mathematically link two crystal structures, we introduced two cost metrics that serve as a measure of the distance along a map. The geometric-cost metric measures the size of the strain and displacement fields required to bring the two structures into coincidence. The symmetry-adapted cost metric measures only that part of the distortion field that breaks the symmetry of the parent crystal structure. Finally we outlined optimization and search algorithms that can be employed to obtain the structure mapping with lowest cost metric and a family of

suboptimal structural transformations. The mapping algorithm can be used to connect ordered crystals to unordered host structures or other ordered phases at the same composition. The algorithm has been implemented within the CASM software package^{15,38,39}. The implementation can be used to generate maps between any pair of crystal structures and to assign a cost to each map.

The pathway generation scheme outlined here has several similarities to that introduced by Trinkle et al.²⁴. For example, the enumeration over possible lattice mappings using unimodular matrices has much in common with the approach of Trinkle et al.²⁴. There are, however, several crucial differences. First, the approach of this work explicitly leverages the symmetry of the underlying parent and child crystal structures to prune the set of possible maps and to discern a symmetry-adapted mapping score. As outlined in section Accounting for symmetry during lattice mappings, strain deformations that are identical under the point group operations of the two crystal structures are screened using eq. (10). Trinkle's algorithm relies on comparisons of the eigenstrains of the deformation tensors to prune elements of the map set \mathcal{M} that represent the same deformation field but are numerically different. Second, our approach employs the Hungarian algorithm (and Murty's algorithm) to systematically enumerate all feasible assignments of atoms between the child and parent structures when generating displacement fields. In contrast, a brute force approach that only counts over "small" displacements of each atom is used in²⁴. Finally, we propose a way of characterizing the complexity of the transformation pathway using a symmetry-adapted cost metric. This cost metric accounts for only symmetry-breaking deformation fields and enables the comparison of qualitatively similar crystal structures that might have very different lattice parameters and atomic coordinates. This is an especially

powerful feature that, to our knowledge, has not yet been exploited in any previously developed crystal mapping algorithms.

Our systematic enumeration of pathways between common crystal structures as summarized in Figs. 5 and 8 has revealed that all known structural transformations can be associated with pathways that have the smallest or close to the smallest mapping costs. The mapping costs enable the systematic exploration of physically meaningful structural transformation pathways. An analysis of transformation pathways relating several commonly occurring crystal structure prototypes⁴⁹ using the algorithm outlined in this study has revealed seven new structural transformation pathways. Using this algorithm to link all known crystal structure prototypes will undoubtedly generate a rich database of phase transformation pathways that will provide insights about a wide variety of structural phase transformations.

The algorithms described here and its implementation within CASM should prove invaluable in organizing high-throughput first-principles databases by enabling a precise cataloguing of crystal structures and the identification of relaxations to new structures due to dynamical instabilities. Similarly, the algorithms will benefit the field of alloy theory^{39,50,51}, where cluster expansion Hamiltonians are trained to first-principles energies of a large number of chemical decorations over the sites of a particular parent crystal structure^{52–64}. In many instances, a subset of orderings on a parent crystal exhibit dynamical instabilities and relax to a decoration on a different parent crystal structure^{59,65}. These relaxations contaminate the training dataset of a cluster expansion and must be detected and eliminated from the training pool. A robust mapping and distance metric is necessary to establish whether a particular ordering can still be considered a decoration on the original parent crystal structure. The mapping tool can also be used to identify symmetric equivalence of deformed crystals, as for example, the crystallographic models of sheared bicrystals used to calculate generalized stacking fault energies⁶⁶.

METHODS

Mapping crystal structures

Transformation pathways connecting various crystal structures were generated with the Clusters Approach to Statistical Mechanics (CASM) code. Figures 4 and 3 were generated by employing the symmetry-adapted and geometric-cost metrics to quantify the complexity of the transformation pathways. The mapping costs shown in Figs. 5 and 8 were generated by searching for symmetrically distinct transformation pathways connecting hcp and bcc to several commonly occurring crystal structures. For each set of parent and child structures, we considered all symmetrically distinct supercells of the child containing at most 12 atoms. Lattice mappings relating the child superstructure to a corresponding superstructure of the parent were generated by counting over all unimodular matrices containing entries with a maximum magnitude of 2. We then recorded the five best atomic maps for each lattice map with the optimality measure being given by the geometric cost. The symmetry-adapted cost metrics were calculated for each of these sets of lattice and atomic maps using Eqs. (24), (31). The symmetry-adapted lattice costs were calculated relative to the parent structures.

DATA AVAILABILITY

The crystal structures used in this study are provided in the Supplementary Information.

CODE AVAILABILITY

The algorithms described in this study have been implemented in the Clusters Approach to Statistical Mechanics (CASM) open source code that is available for download from <https://github.com/prisms-center/CASMcode>. The mapping algorithms are seamlessly integrated into the workflow of CASM and are leveraged when populating internal configurational databases from electronic structure calculations or when importing arbitrary structures through the “casm import” command.

Received: 9 April 2021; Accepted: 1 September 2021;

Published online: 07 October 2021

REFERENCES

- Zagorac, D., Müller, H., Ruehl, S., Zagorac, J. & Rehme, S. Recent developments in the Inorganic Crystal Structure Database: theoretical crystal structure data and related features. *J. Appl. Crystallogr.* **52**, 918–925 (2019).
- Kolli, S. K., Natarajan, A. R., Thomas, J. C., Pollock, T. M. & Van der Ven, A. Discovering hierarchies among intermetallic crystal structures. *Phys. Rev. Mater.* **4**, 113604 (2020).
- Ferro, R. & Saccone, A. *Intermetallic Chemistry* (Pergamon, 2008).
- De Graef, M. & McHenry, M. E. *Structure of Materials*, 1st edn. (Cambridge, 2007).
- Jain, A. et al. Commentary: The Materials Project: a materials genome approach to accelerating materials innovation. *APL Mater.* **1**, 011002 (2013).
- Curtarolo, S. et al. AFLOWLIB.ORG: a distributed materials properties repository from high-throughput ab initio calculations. *Comput. Mater. Sci.* **58**, 227–235 (2012).
- Kirklin, S. et al. The Open Quantum Materials Database (OQMD): assessing the accuracy of DFT formation energies. *npj Comput. Mater.* **1**, 15010 (2015).
- Draxl, C. & Scheffler, M. The NOMAD laboratory: from data sharing to artificial intelligence. *J. Phys. Mater.* **2**, 036001 (2019).
- Talirz, L. et al. Materials Cloud, a platform for open computational science. *Sci. Data* **7**, 299 (2020).
- Cowley, R. Structural phase transitions i. Landau theory. *Adv. Phys.* **29**, 1–110 (1980).
- Otsuka, K. & Wayman, C. M. *Shape memory materials* (Cambridge university press, 1999).
- Christian, J. W. *The Theory of Transformations in Metals and Alloys* (Newnes, 2002).
- Zhang, M.-X. & Kelly, P. M. Crystallographic features of phase transformations in solids. *Prog. Mater. Sci.* **54**, 1101–1170 (2009).
- Khachatryan, A. G. *Theory of Structural Transformations in Solids* (Courier Corporation, 2013).
- Thomas, J. C. & Van der Ven, A. Finite-temperature properties of strongly anharmonic and mechanically unstable crystal phases from first principles. *Phys. Rev. B* **88**, 214111–214111 (2013).
- Thomas, J. C. & Van der Ven, A. Elastic properties and stress-temperature phase diagrams of high-temperature phases with low-temperature lattice instabilities. *Phys. Rev. B* **90**, 224105 (2014).
- Bechtel, J. S., Thomas, J. C. & Van der Ven, A. Finite-temperature simulation of anharmonicity and octahedral tilting transitions in halide perovskites. *Phys. Rev. Mater.* **3**, 113605 (2019).
- Radin, M. D., Thomas, J. C. & Van der Ven, A. Order-disorder versus displacive transitions in Jahn-Teller active layered materials. *Phys. Rev. Mater.* **4**, 043601 (2020).
- Bain, E. & Dunkirk, N. The nature of martensite. *Transactions of the American Institute of Mining and Metallurgical Engineers.* **70**, 25–35 (1924).
- Burgers, W. On the process of transition of the cubic-body-centered modification into the hexagonal-close-packed modification of zirconium. *Physica* **1**, 561–586 (1934).
- Silcock, J. An X-ray examination of the α phase in TiV, TiMo and TiCr alloys. *Acta Metall.* **6**, 481–493 (1958).
- de Fontaine, D. Mechanical instabilities in the bcc lattice and the beta to omega phase transformation. *Acta Metall.* **18**, 275–279 (1970).
- Trinkle, D. R. et al. New mechanism for the α to ω martensitic transformation in pure titanium. *Phys. Rev. Lett.* **91**, 025701 (2003).
- Trinkle, D. R., Hatch, D. M., Stokes, H. T., Hennig, R. G. & Albers, R. C. Systematic pathway generation and sorting in martensitic transformations: titanium α to ω . *Phys. Rev. B* **72**, 014105 (2005).
- Natarajan, A. R. & Van der Ven, A. Connecting the simpler structures to topologically close-packed phases. *Phys. Rev. Lett.* **121**, 255701 (2018).
- Natarajan, A. R. & Van der Ven, A. Toward an understanding of deformation mechanisms in metallic lithium and sodium from first-principles. *Chem. Mater.* **31**, 8222–8229 (2019).
- Natarajan, A. R., Dolin, P. & Van der Ven, A. Crystallography, thermodynamics and phase transitions in refractory binary alloys. *Acta Mater.* **200**, 171–186 (2020).
- Gelato, L. M. & Parthé, E. STRUCTURE TIDY—a computer program to standardize crystal structure data. *J. Appl. Crystallogr.* **20**, 139–143 (1987).
- Burzlaff, H. & Malinovsky, Y. A procedure for the classification of non-organic crystal structures. I. Theoretical background. *Acta Crystallogr. Sec. A Found. Crystallogr.* **53**, 217–224 (1997).
- Dzyabchenko, A. V. Method of crystal-structure similarity searching. *Acta Crystallogr. Sec. B Struct. Sci.* **50**, 414–425 (1994).

31. de la Flor, G., Orobengoa, D., Tasci, E., Perez-Mato, J. M. & Aroyo, M. I. Comparison of structures applying the tools available at the Bilbao Crystallographic Server. *J. Appl. Crystallogr.* **49**, 653–664 (2016).
32. Willighagen, E. L., Wehrens, R., Verwer, P., de Gelder, R. & Buydens, L. M. C. Method for the computational comparison of crystal structures. *Acta Crystallogr. Sec. B Struct. Sci.* **61**, 29–36 (2005).
33. Hundt, R., Schön, J. C. & Jansen, M. CMPZ—an algorithm for the efficient comparison of periodic structures. *J. Appl. Crystallogr.* **39**, 6–16 (2006).
34. Lonie, D. C. & Zurek, E. Identifying duplicate crystal structures: XtalComp, an open-source solution. *Computer Phys. Commun.* **183**, 690–697 (2012).
35. Su, C. et al. Construction of crystal structure prototype database: methods and applications. *J. Phys. Condens. Matter* **29**, 165901 (2017).
36. Ong, S. P. et al. Python Materials Genomics (pymatgen): a robust, open-source python library for materials analysis. *Comput. Mater. Sci.* **68**, 314–319 (2013).
37. Hicks, D. et al. AFLOW-XtalFinder: a reliable choice to identify crystalline prototypes. *npj Comput. Mater.* **7**, 30 (2021).
38. Puchala, B. & Van Der Ven, A. Thermodynamics of the zr-o system from first-principles calculations. *Phys. Rev. B Condens. Matter Mater. Phys.* **88**, 1–15 (2013).
39. Van der Ven, A., Thomas, J. C., Puchala, B. & Natarajan, A. R. First-principles statistical mechanics of multicomponent crystals. *Annu. Rev. Mater. Res.* **48**, 27–55 (2018).
40. Hart, G. L. W. & Forcade, R. W. Algorithm for generating derivative structures. *Phys. Rev. B* **77**, 224115 (2008).
41. Thomas, J. C. & Van der Ven, A. The exploration of nonlinear elasticity and its efficient parameterization for crystalline materials. *J. Mech. Phys. Solids* **107**, 76–95 (2017).
42. Lenstra, A. K., Lenstra, H. W. & Lovász, L. Factoring polynomials with rational coefficients. *Math. Ann.* **261**, 515–534 (1982).
43. Srinivasan, G. *Operations Research: Principles and Applications*, 2nd edn. (Prentice Hall India, 2010).
44. Munkres, J. Algorithms for the assignment and transportation problem. *J. Soc. Indust. Appl. Math.* **5**, 32–38 (1957).
45. Murty, K. G. Letter to the Editor—An algorithm for ranking all the assignments in order of increasing cost. *Operations Res.* **16**, 682–687 (1968).
46. Banerjee, D. & Williams, J. Perspectives on titanium science and technology. *Acta Material.* **61**, 844–879 (2013).
47. Fontaine, D. Simple models for the omega phase transformation. *Metall. Trans. A* **19**, 169–175 (1988).
48. Hatt, B. A. & Roberts, J. A. The ω phase in zirconium base alloys. *Acta Metall.* **8**, 10 (1960).
49. Kolli, S. K., Natarajan, A. R. & Van der Ven, A. Seven new transformation pathways connecting simple crystal structures and common intermetallic crystal structures. (2021).
50. Sanchez, J., Ducastelle, F. & Gratias, D. Generalized cluster description of multicomponent systems. *Phys. A Stat. Mech. Appl.* **128**, 334–350 (1984).
51. de Fontaine, D. Cluster approach to order-disorder transformations in alloys. *Solid State Phys.* **47**, 33–176 (1994).
52. van de Walle, A. & Ceder, G. Automating first-principles phase diagram calculations. *J. Phase Equilib.* **23**, 348 (2002).
53. Hart, G. L. W., Blum, V., Walorski, M. J. & Zunger, A. Evolutionary approach for determining first-principles hamiltonians. *Nat. Mater.* **4**, 391–394 (2005).
54. Mueller, T. & Ceder, G. Bayesian approach to cluster expansions. *Phys. Rev. B* **80**, 024103 (2009).
55. Nelson, L. J., Hart, G. L. W., Zhou, F. & Ozoliņš, V. Compressive sensing as a paradigm for building physics models. *Phys. Rev. B* **87**, 035125 (2013).
56. Nelson, L. J., Ozoliņš, V., Reese, C. S., Zhou, F. & Hart, G. L. W. Cluster expansion made easy with Bayesian compressive sensing. *Phys. Rev. B* **88**, 155105 (2013).
57. Kristensen, J. & Zabarar, N. J. Bayesian uncertainty quantification in the evaluation of alloy properties with the cluster expansion method. *Computer Phys. Commun.* **185**, 2885–2892 (2014).
58. Aldegunde, M., Zabarar, N. & Kristensen, J. Quantifying uncertainties in first-principles alloy thermodynamics using cluster expansions. *J. Comput. Phys.* **323**, 17–44 (2016).
59. Goiri, J. G. & Van der Ven, A. Phase and structural stability in Ni-Al systems from first principles. *Phys. Rev. B* **94**, 094111 (2016).
60. Natarajan, A. R. & Van der Ven, A. Machine-learning the configurational energy of multicomponent crystalline solids. *npj Comput. Mater.* **4**, 1–7 (2018).
61. Gunda, N. H. & Van der Ven, A. First-principles insights on phase stability of titanium interstitial alloys. *Phys. Rev. Mater.* **2**, 083602 (2018).
62. Gunda, N. H., Puchala, B. & Van der Ven, A. Resolving phase stability in the ti-o binary with first-principles statistical mechanics methods. *Phys. Rev. Mater.* **2**, 033604 (2018).
63. Goiri, J. G. & Van der Ven, A. Recursive alloy hamiltonian construction and its application to the ni-al-cr system. *Acta Mater.* **159**, 257–265 (2018).
64. Natarajan, A. R. & Van der Ven, A. Linking electronic structure calculations to generalized stacking fault energies in multicomponent alloys. *npj Comput. Mater.* **6**, 1–9 (2020).
65. Natarajan, A. R. & Van der Ven, A. First-principles investigation of phase stability in the Mg-Sc binary alloy. *Phys. Rev. B* **95**, 214107 (2017).
66. Goiri, J. G. & Van der Ven, A. Multishifter: Software to generate structural models of extended two-dimensional defects in 3d and 2d crystals. *Comput. Mater. Sci.* **191**, 110310 (2021).

ACKNOWLEDGEMENTS

J.C.T. was supported by U.S. Department of Energy, Office of Basic Energy Sciences, Division of Materials Sciences, and Engineering under Award DE-SC0008637 as part of the Center for Predictive Integrated Structural Materials Science (PRISMS Center). Contributions from A.R.N. were supported by the National Science Foundation DMREF grant: DMR-1729166 DMREF/GOAL: Integrated Computational Framework for Designing Dynamically Controlled Alloy -Oxide Heterostructures. AVDV was supported by the NSF IDEAS center, Grant No. 1934641. We thank Jonathon Bechtel and Sanjeev Kolli for software contributions and discussions. Computational resources provided by the National Energy Research Scientific Computing Center (NERSC), supported by the Office of Science and US Department of Energy under Contract No. DE-AC02-05CH11231, are gratefully acknowledged, in addition to support from the Center for Scientific Computing from the CNSI, MRL, and NSF MRSEC (No. DMR-1720256).

AUTHOR CONTRIBUTIONS

J.C.T., A.R.N. and A.V.D.V. conceived the project. J.C.T. developed and implemented the lattice-mapping algorithm and geometric-cost metric, and developed the atomic mapping approach with input from A.R.N. A.R.N. developed and implemented the symmetry-adapted cost metrics with input from J.C.T. and performed the benchmarking analysis of the symmetry-adapted metrics. J.C.T. and A.R.N. performed the software implementation within CASM. J.C.T. initiated and A.R.N. completed the calculations and analysis relating prototype crystal structures. J.C.T., A.R.N. and A.V.D.V. wrote the manuscript.

COMPETING INTERESTS

The authors declare no competing interests

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41524-021-00627-0>.

Correspondence and requests for materials should be addressed to John C. Thomas, Anirudh Raju Natarajan or Anton Van der Ven.

Reprints and permission information is available at <http://www.nature.com/reprints>

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



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2021