Generalized regular **k**-point grid generation on the fly

Wiley S. Morgan[a], John E. Christensen[a], Parker K. Hamilton[a], Jeremy J. Jorgensen[a], Branton J. Campbell[a], Gus L.W. Hart[a,*], Rodney W. Forcade[b]

[a] Department of Physics and Astronomy, Brigham Young University, Provo, UT 84602, USA
[b] Department of Mathematics, Brigham Young University, Provo, UT 84602, USA

ARTICLE INFO

ABSTRACT

In the DFT community, it is common practice to use *regular* **k**-point grids (Monkhorst-Pack, MP) for Brillioun zone integration. Recently Wisesa et al. (2016) and Morgan et al. (2018) demonstrated that *generalized regular* (GR) grids offer an advantage over traditional MP grids. The difference is simple but effective. At the same **k**-point density, GR grids have greater symmetry and 60% fewer irreducible **k**-points. GR grids have not been widely adopted because one must search through a large number of candidate grids; in many cases, a brute force search could take hours. This work describes an algorithm that can quickly search over GR grids for those that have the most uniform distribution of points and the best symmetry reduction. The grids are ~60% more efficient, on average, than MP grids and can now be generated *on the fly* in seconds.

1. Introduction

In computational materials science, the properties of crystalline materials are often calculated using density functional theory (DFT). These codes integrate the electronic energy over the occupied states in the Brillouin zone. In the case of metals, convergence of the total energy is very slow. The convergence rate is approximately proportional to the density of **k**-points used to sample the Brillouin zone. An order of magnitude increase in accuracy requires *at least* an order of magnitude more **k**-points [2].

Additionally, as high throughput [3–22] calculations have become more popular because of their recent successes [23–52], the efficiency of the calculations becomes more important. The accuracy and quantity of calculations within material databases is a crucial component in high throughput and machine learning approaches. Increasing the speed of calculations, without reducing the accuracy, would significantly impact material predictions.

DFT codes generally use regular grids, proposed by Monkhorst and Pack (MP) [53], to define their **k**-point grids. **k**-points within a regular grid are defined by:

$$\mathbf{k}=(\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3)\mathbb{D}^{-1}(n_1, n_2, n_3) = \frac{n_1}{d_1}\mathbf{b}_1 + \frac{n_2}{d_2}\mathbf{b}_2 + \frac{n_3}{d_3}\mathbf{b}_3 \qquad (1)$$

where $\mathbf{b}_i$ are the reciprocal lattice vectors, $\mathbb{D}$ is a diagonal integer matrix with d_i along the diagonal, and n_i runs from 0 to $d_i - 1$.

An alternative, more general method was proposed by Moreno and Soler [54], which involves searching through grids at a desired **k**-point density for those that have the highest symmetry reduction, i.e., the lowest general-point multiplicity or fewest symmetrically distinct **k**-points. High symmetry reduction impacts the computation's cost; the cost of a DFT calculation scales with the number of irreducible **k**-points. The grids are then sorted by the length of the shortest grid generating vector and the grid with the longest vector is choosen, thus selecting the most uniform grid (that is, the grid with the highest packing fraction for densely packed spheres). The Moreno-Soler method involves the construction of superlattices from the real-space parent lattice (primitive lattice)

$$(\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3) = (\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3)\mathbb{H} \qquad (2)$$

where the columns $\mathbf{s}_i$ are the supercell vectors, the columns $\mathbf{a}_i$ are the parent lattice vectors, and $\mathbb{H}$ is an integer matrix. The dual lattice of the superlattice defines the **k**-point grid generating vectors κ_i.

$$(\kappa_1, \kappa_2, \kappa_3)=2\pi((\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3)^{-1})^{\mathsf{T}}=2\pi(((\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3)\mathbb{H})^{-1})^{\mathsf{T}}$$
$$=2\pi((\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3)^{-1})^{\mathsf{T}}(\mathbb{H}^{-1})^{\mathsf{T}} = (\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3)(\mathbb{H}^{-1})^{\mathsf{T}} \qquad (3)$$

Note that the determinant of $\mathbb{H}$ determines the number of **k**-points that lie within the Brillouin zone.

* Corresponding author.
  E-mail address: gus.hart@byu.edu (G.L.W. Hart).
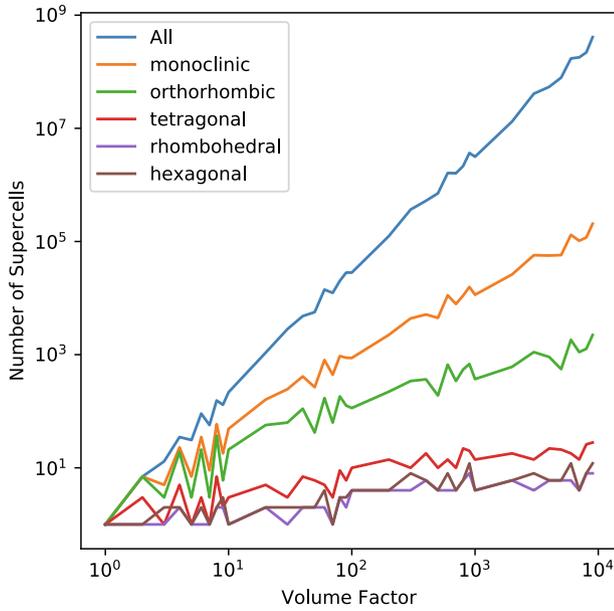
Fig. 1. The number of superlattices that preserve the symmetry of the parent lattice at various volume factors (numbers of k-points). The total number of superlattices that exist is also displayed for comparison. Cubic cells were omitted since they have at most one symmetry-preserving superlattice at any given volume factor.

We refer to grids generated by the Moreno-Soler method as *Generalized Regular* (GR) grids. GR grids have a number of advantages over Monkhorst-Pack grids. Most importantly, GR grids, on average, have a much better symmetry reduction (fewer irreducible k-points), making them more efficient. (Note: With MP grids it is possible to choose the lattice vectors so that the maximum symmetry reduction is achieved, but in practice this is difficult for the user and contrasts with accepted practice where the user typically chooses the most orthogonal set of vectors.) Secondly, the number of grids is much higher for GR grids, allowing the user to choose k-point densities "in between" those possible with MP grids. Finally, the uniformity of GR grids (the packing fraction) is often slightly higher. (Though this is a minor advantage for metals where convergence is so slow that uniformity of the grid has little effect.)

GR grids have never been widely adopted because they require a search over many supercells to select the cell that 1) maximizes the distance between points and 2) has the fewest irreducible k-points, i.e., has the highest symmetry reduction. These searches tend to be time consuming due to the combinatoric explosion in the total number of possible supercells shown in Fig. 1.

Recently Wisesa, McGill, and Mueller [1] (WMM) rectified this by creating a k-point server containing precalculated grids that have high symmetry reduction. These grids can be retrieved via an internet request and have been demonstrated to be 60% more efficient than MP grids [2,1]. However, the requirement of an internet query, which cannot be performed in typical supercomputer environments, makes them difficult to use in some cases. Here we present an algorithm for generating GR grids "on the fly" (avoiding the need for an internet query). This algorithm has been implemented in a code available at https://github.com/msg-byu/autoGR. This code takes the numerical lattice vectors, atomic basis vectors, and grid density from a user and returns the optimal GR grid.

The algorithm is fast, taking just a few seconds for grids of ~50,000 k-points. The combinatorial explosion is avoided by the algorithm below, which limits the candidate grids to those that are symmetry-preserving and which have a sufficiently high packing fraction (a metric equivalent to the maximum real-space lattice spacing of Ref. 1).

2. Algorithm

2.1. Generating symmetry-preserving supercells

The main difficulty in generating GR grids is that the number of distinct supercells grows extremely rapidly[1] with the volume factor (the determinant of $\mathbb{H}$).[2] To optimize the k-point folding efficiency, the k-point grid should result in a real space superlattice[3] that has the same symmetry as the parent lattice. The number of supercells that preserve the symmetry of the parent is always significantly smaller than the number of possible supercells (except in the case of triclinic lattices) as can be seen in Fig. 1. If one can quickly generate only those supercells that preserve the symmetry of the parent, avoiding the combinatorial explosion, the computational burden is drastically reduced.

To generate only the symmetry-preserving supercells, we restrict $\mathbb{H}$ to be an integer matrix in Hermite Normal Form (HNF) subject to the constraints:

$$\mathbb{H} = \begin{pmatrix} a & 0 & 0 \\ b & c & 0 \\ d & e & f \end{pmatrix}$$

$$a, \ c, \ f > 0$$
$$b \geqslant 0, \quad b < c$$
$$d, e \geqslant 0, \quad d, e < f \tag{4}$$

We will use the notation that $\mathbb{A} = (\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3)$ is the parent lattice and $\mathbb{C} = (\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3)$ is a supercell such that $\mathbb{C} = \mathbb{A}\mathbb{H}$. When the lattice symmetries are applied to $\mathbb{A}$, they generate another set of basis vectors,

$$\mathbb{A}' = \mathbf{g}\mathbb{A} \tag{5}$$

(where $\mathbf{g}$ is an element of the point group). Because $\mathbb{A}$ and are related by a symmetry operation of the lattice, they both represent the same lattice and are related by an integer matrix

$$\mathbb{A}' = \mathbb{A}\mathbb{X}$$
$$\mathbb{A}\mathbb{X} = \mathbf{g}\mathbb{A}$$
$$\mathbb{X} = \mathbb{A}^{-1}\mathbf{g}\mathbb{A} \tag{6}$$

where $\mathbb{X}$ is an integer matrix with determinant ± 1. Similarly, if a supercell $\mathbb{C}$ has the same symmetry as $\mathbb{A}$ then all the symmeties of $\mathbb{A}$ will map $\mathbb{C}$ to another basis $\mathbb{C}'$ that will be related to $\mathbb{C}$ by a unimodular transformation

$$\mathbb{C}' = \mathbf{g}\mathbb{C} \quad \forall \ \mathbf{g} \in \mathbf{G}$$
$$\mathbb{C}\mathbb{M} = \mathbf{g}\mathbb{C}$$
$$\mathbb{M} = \mathbb{C}^{-1}\mathbf{g}\mathbb{C} \tag{7}$$

where $\mathbf{G}$ is the set of generators of the point group of $\mathbb{A}$ and $\mathbb{M}$ is an integer matrix. Using Eqs. (6) and (7), it is possible to define restrictions on the entries of $\mathbb{H}$:

$$\mathbb{M} = \mathbb{H}^{-1}\mathbb{X}\mathbb{H}. \tag{8}$$

In other words $\mathbb{H}$ must be such that $\mathbb{M}$ is a transformation of $\mathbb{X}$ that retains integer entries. Eq. (8) yields the following system of linear equations

---

[1] Eq. (2) in Ref. [67]

[2] Note that the determinant of $\mathbb{H}$ determines the number of k-points in the Brillouin zone.

[3] In the mathematical and crystallography literature, these derivative lattices are referred to as sublattices. Although this nomenclature is more correct, we follow the nomenclature typically seen in the physics literature where a derivative lattice whose volume is larger than that of the parent is referred as a superlattice.

$$\alpha_1 = \frac{bx_{12} + dx_{13}}{a}$$

$$\alpha_2 = \frac{cx_{12} + ex_{13}}{a}$$

$$\alpha_3 = \frac{fx_{13}}{a}$$

$$\beta_1 = \frac{-bx_{11} + ax_{21} - b\alpha_1 + bx_{22} + dx_{23}}{c}$$

$$\beta_2 = \frac{-b\alpha_2 + ex_{23}}{c}$$

$$\beta_3 = \frac{-b\alpha_3 + cx_{23}}{c}$$

$$f = \frac{\alpha_4}{c}$$

$$\gamma_1 = \frac{ax_{31} + bx_{32} + dx_{33} - e\beta_1 - d\alpha_1 - dx_{11}}{f}$$

$$\gamma_2 = \frac{-ex_{22} + cx_{32} + ex_{33} - e\beta_2 - d\alpha_2}{f}$$

$$n = a \cdot c \cdot f \tag{9}$$

where x_i are the entries of $\mathbb{X}$, n is the determinant of $\mathbb{H}$ and α_i, β_i, and γ_i are arbitrary names for the expressions used for convenience. $\mathbb{H}$ will generate a supercell that preserves the symmetries of $\mathbb{A}$ when α_1, α_2, α_3, α_4, β_1, β_2, β_3, γ_1, and γ_2 are all integers for each generator in **G**. Even though the solutions to (9) have no closed form, we may use them to build an algorithm that generates $\mathbb{H}$ matrices that preserve the lattice symmetries.

The specific form of $\mathbb{X}$ depends on the basis chosen for the parent lattice; the solutions to (9), and resulting algorithms, will differ depending on the basis. For example, if a base-centered orthorhombic lattice is constructed with the basis

$$\mathbb{A}_1 = (\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3) = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ 1 & -1 & 0 \\ 0 & 0 & 3 \end{pmatrix} \tag{10}$$

then (9) would reduce to (each equation has three outputs because the base centered orthormbic point-group has three generators):

$$\alpha_1 = \left(0,\ 0,\ -\frac{b}{a}\right)$$

$$\alpha_2 = \left(0,\ 0,\ -\frac{c}{a}\right)$$

$$\alpha_3 = \beta_3 = (0,\ 0,\ 0)$$

$$\beta_1 = \left(0,\ 0,\ \frac{-a - b\alpha_1}{c}\right)$$

$$\beta_2 = \left(0,\ 0,\ \frac{b}{a}\right)$$

$$\gamma_1 = \left(0,\ \frac{2d}{f},\ \frac{-d - d\alpha_1 - e\beta_1}{f}\right)$$

$$\gamma_2 = \left(0,\ \frac{2e}{f},\ \frac{-e - d\alpha_2 - e\beta_2}{f}\right) \tag{11}$$

All the equations in (11) must be simultaneously satisfied for the generated $\mathbb{H}$'s to preserve the symmetries of $\mathbb{A}_1$. Alternatively the basis

$$\mathbb{A}_2 = (\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3) = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 1 & -2 & 0 \\ 0 & 0 & 3 \end{pmatrix} \tag{12}$$

could be used to construct the same lattice. When basis $\mathbb{A}_2$ is chosen, the relations in (9) become:

$$\alpha_1 = \alpha_2 = \alpha_3 = \beta_2 = \beta_3 = (0,\ 0,\ 0)$$

$$\beta_1 = \left(0,\ 0,\ \frac{a + 2b}{c}\right)$$

$$\gamma_1 = \left(0,\ \frac{2d}{f},\ \frac{-e\beta_1}{f}\right)$$

$$\gamma_2 = \left(0,\ \frac{2e}{f},\ -\frac{2e}{f}\right) \tag{13}$$

Note the stark difference between the relationships derived from $\mathbb{A}_1$

and $\mathbb{A}_2$. $\mathbb{A}_2$ results in fewer equations to check, however, $\mathbb{A}_1$ gives relationships between a and b, and a and c separately resulting in a faster search since many combinations can be skipped early in the search. By taking care in selecting a basis for each lattice, one can find an efficient set of conditions for generating the supercells of that basis.

2.2. Niggli reduction

Choosing a basis for each type of lattice presents a problem; there are an infinite number of lattices basis choices. The number of bases is substantially reduced by recognizing that any given symmetry-preserving HNF, $\mathbb{H}^{sp}$, will work for *every* lattice of the same symmetry. The sensitivity of the representation of the point group $\mathbb{X}$ on the chosen basis requires a set of representative bases that goes beyond the 14 Bravais lattices. Such a set was constructed by Niggli [55–58], who identified 44 distinct bases. Any given basis of a crystal can be classified as one of these 44 cases by reducing it to the Niggli canonical form and then comparing the lengths of the basis vectors and the angles between them. If two nominally different lattices reduce to the same Niggli case, then the two lattices are "equivalent" and have the same symmetries and the same set of $\mathbb{H}^{sp}$s.

Niggli reduction allows for the user's basis to be mapped to a basis which has convenient solutions to Eqs. (9). The strategy is to define the $\mathbb{H}^{sp}$'s in the selected basis, then generate the supercells for the selected basis and transform them to the $\mathbb{H}$'s for the Niggli reduced basis, $\mathbb{H}_R^{sp}$. Once the $\mathbb{H}_R^{sp}$'s have been determined, they can be applied directly to the user's reduced basis to create a symmetry-preserving supercell of the user's parent cell and thus define an efficient **k**-point grid at the specified density.

2.3. Grid selection

At a given volume factor (i.e., number of **k**-points), the integer relations in Eq. (9) will yield multiple supercells for most lattices, a 2D example of these supercells is provided in Fig. 2(a). It is then neccessary to select one which defines the best **k**-point grid. This is done by transforming each symmetry-preserving supercell to its corresponding **k**-point grid generating vectors as in Eq. (3); see Fig. 2(b). We then search this set of grids for one that has optimal properties—a uniform distribution of points and the best symmetry reduction. To ensure the grid generating vectors are as short as possible we perform Minkowski reduction [59], then sort the grids by the length of their shortest vector. (This is practically equivalent to maximizing the shortest real-space lattice vector, as done in Ref. 1.)

The most uniform grids will have the maximal shortest lattice vector in real space. We filter the grids so that only those with a **k**-point *packing fraction* of greater than 0.3 are considered. (The packing fraction is $\frac{N_{\text{k-points}} V_{\text{k-point}}}{V_{\text{reciprocalcell}}}$, where the volume of a single **k**-point is computed using the distance to the closest neighboring **k**-point.) Each of the remaining grids is then symmetry reduced [60] in order to determine which has the fewest irreducible **k**-points. Table 1 shows the length of the shortest vector and number of irreducible **k**-points for the grids in Fig. 2(b). The grids are sorted first by the length of their shortest vector (eliminating the green and red grids) then by the number of irreducible **k**-points such that the ideal grid appears at the top of the table, i.e., the grid generated by the brown supercell in Fig. 2(a).

It is also possible to offset the **k**-point grid from the origin to improve the grid's efficiency. The origin is not symmetrically equivalent to any other point in the grid; for example, including an offset makes it possible for the point at the origin to be mapped to other points in the grid, decreasing the number of irreducible **k**-points. Different grids
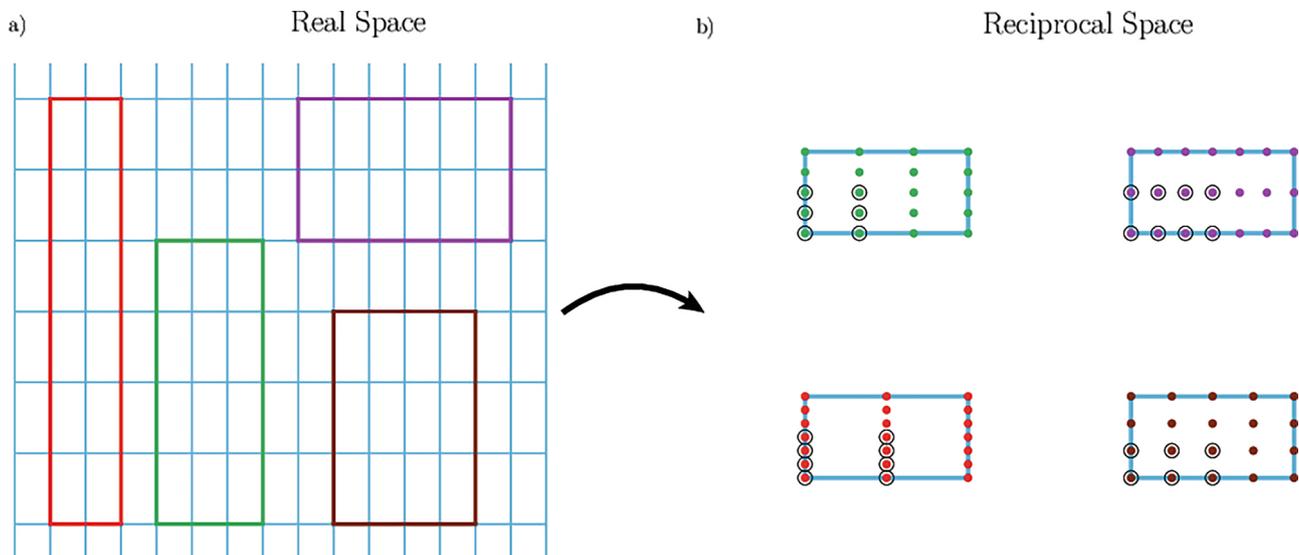
Fig. 2. A 2D example of symmetry-preserving supercells and the **k**-point grids that they would generate for a rectangular lattice. a) contains four symmetry-preserving supercells of the primitive cell, shown in blue, with a volume factor of 12. In b) the primitive cell, blue cells, and the supercells have been mapped to reciprocal space and the grids that would have been generated from each supercell have been placed in a cell. The color of the grid points matches the color of the generating supercell. The circled points are the irreducible **k**-points of each grid.

Table 1
Properties (length of shortest vector and number of irreducible **k**-points) of the grids in Fig. 2.

Grid	Shortest vector length	Number of irreducible **k**-points
Brown	$\frac{1}{6}$	6
Purple	$\frac{1}{6}$	8
Green	$\frac{1}{8}$	6
Red	$\frac{1}{12}$	8

have different symmetry-preserving offsets that should be tested. For example, both simple cubic and face-centered cubic (fcc) grids have one possible offset that preserves the full symmetry of the lattice, $\left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right)$ (expressed as fractions of the grid generating vectors), while a body-centered-cubic lattice has no symmetry-preserving offsets,[4] and simple tetragonal has three symmetry-preserving offsets. (For a full list of the symmetry-preserving offsets by lattice type, see the Appendix.) The grid that has the fewest **k**-points with a given offset is selected.

Not every volume factor will have a symmetry-preserving grid that is uniform; some volume factors will have no symmetry-preserving grids. To ensure that a symmetry-preserving grid is found, it is necessary to include multiple volume factors in the search. The number of additional volume factors to search depends on the lattice type; in general, the search continues until multiple candidate grids have been found. The best grid is then selected from these candidates.

2.4. Method summary

The algorithm can be summarized in the following steps:

1. Identify the Niggli reduced cell of the user's structure.
2. Generate the symmetry-preserving HNFs for the canonical form of the Niggli cell.
3. Map the resulting supercells to the original lattice using the Niggli-reduced basis as an intermediary.
4. Convert the supercells into **k**-point grid generating vectors.
5. Perform Minkowski reduction on the grid generating vectors.
6. Sort the grid generating vectors by the length of their shortest vector.
7. Select the grids that maximize the length of the shortest vectors.
8. Use the symmetry group to reduce the selected grids to find the one with the fewest irreducible **k**-points.

3. Demonstration

To test the above algorithm, we compared the **k**-point grids it generates, GR_{auto}, to those generated by the **k**-point sever [1], GR_{server} in two ways. First, we generated both grids over a range of **k**-point densities for over 100 crystal lattices. These lattices were constructed for nine elemental systems—Al, Pd, Cu, W, V, K, Ti, Y, and Re—with supercells for the cubic systems having between 1 and 11 atoms per cell and supercells for the hexagonal close packed systems having between 2 and 14 atoms per cell. Additional test structures were selected from AFLOW [3]. All tests were conducted without offsetting the grids from the origin, i.e., all tests were done for Γ-centered grids. The tests were performed by querrying the **k**-point server for a wide range of r_{min} values with the `INCLUDEGAMMA = TRUE` flag included. The resulting number of total and irreducible **k**-points was then read from the `KPOINTS` file returned. For a fair comparison each r_{min} value was then converted into a minimum number of **k**-points using the formula [1]:

$$N = \frac{\sqrt{2}}{2} r_{min}^3 / V_p \tag{14}$$

where V_p is the volume of the real space cell. The GR_{auto} algorithm was then run with the input flags `NKPTS = npts` and `SHIFT = 0.0 0.0 0.0`. We then plotted the resulting ratio of irreducible **k**-points to total **k**-points in each grid. Six representative examples of the results are shown in Fig. 3. These tests show that the GR_{auto} grids should be very close in performance to GR_{server} grids. Additionally, the tests show that convergence toward the ideal folding ratio is rapid for all lattice types.

---

[4] For some lattices no symmetry-preserving offsets exist. In these cases using an offset that does not preserve the full symmetry can be beneficial. For example, a body centered cubic system with an offset of $(0, 0, \frac{1}{2})$ can sometimes offer better folding than the same grid with no offset.
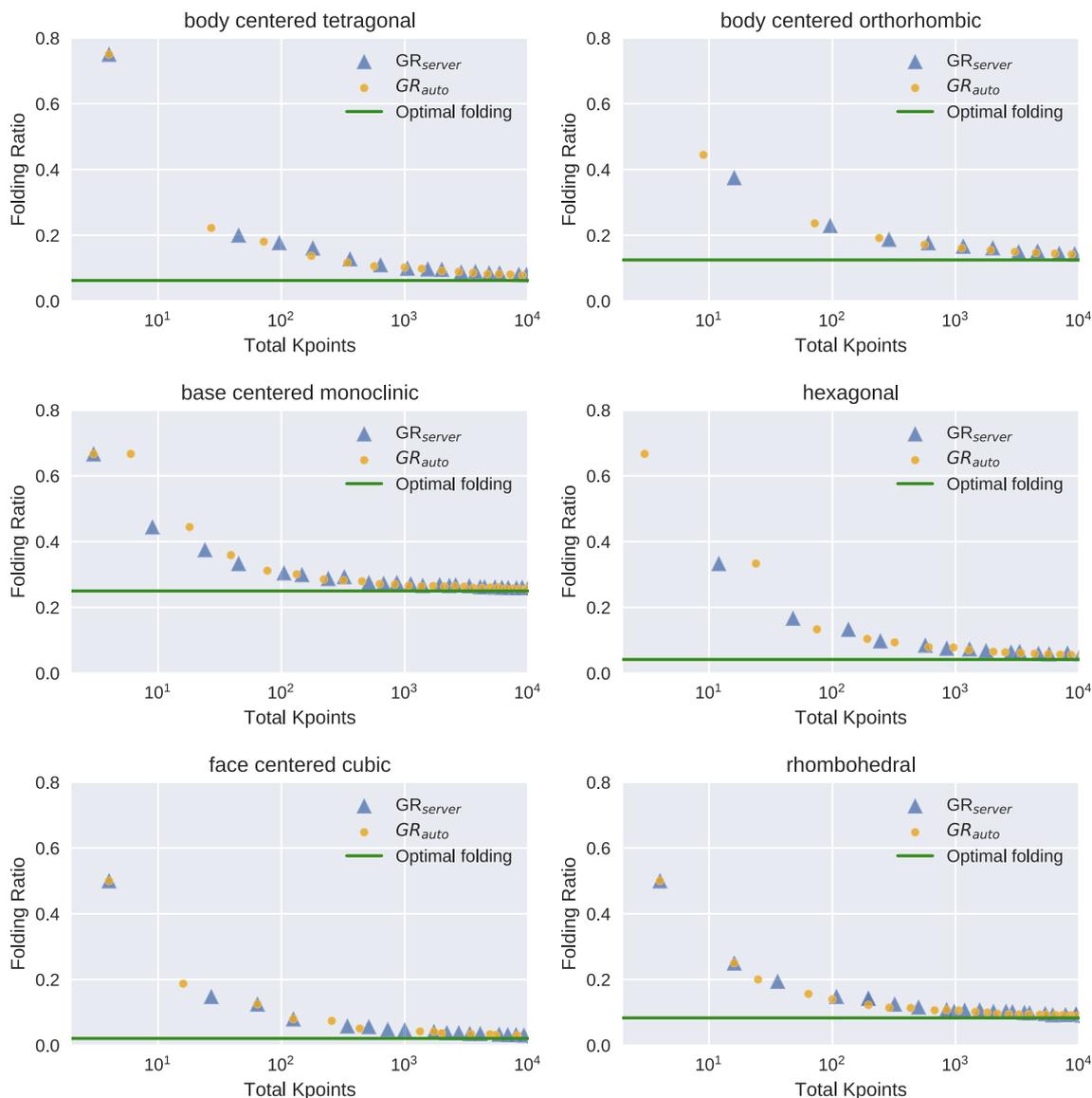
Fig. 3. A comparison of the GR_{auto} and GR_{server} **k**-point grids. For each grid the number of irreducible **k**-points was divided by the total number of **k**-points. This shows that both sets of grids offer similar folding at a given **k**-point denstiy and will have similar efficiencies.

The second test compared the total energy errors of MP (generated by AFLOW), GR_{auto} and GR_{server} grids in the same manner and using the same methods, as done in our previous study of GR grids [2]. We provide a brief review of that method here.

DFT calculations were performed using the Vienna Ab-initio Simulation Package 4.6 (VASP 4.6) [61–64] on the nine pure-element systems mentioned above using PAW PBE pseudopotentials [65,66]. In order to isolate the errors from **k**-point integration, the different cells were crystallographically equivalent to single element cells. (For details, see Ref. [2].) For MP grids, the target number of **k**-points ranged from 10 to 10,000 unreduced **k**-points, for GR_{server} grids the range was 4–240,000 unreduced **k**-points, and for GR_{auto} the range was 8–415,000 unreduced **k**-points. The GR_{auto} grids were generated at densities such that more grids were generated at greater densities. The GR_{server} grids were generated by using small step sizes over the `MINDISTANCE` parameter and throwing out duplicate

grids. In total, we compared errors across more than 7000 total energy calculations. The energy taken as the error-free "solution" in our comparisons was the calculation with the highest **k**-point density for each system. The total error convergence with respect to the **k**-point density is shown in Fig. 4. The total error convergence with respect to the number of irreducible **k**-points was compared using loess regression, see Fig. 5. Ratios of these trend lines were then taken to determine the efficiency of each grid relative to the GR_{server} grids (see Fig. 6).

From Figs. 5 and 6, it can be seen that GR_{auto} grids are up to ~10% more efficient and at worst ~5% less efficient than GR_{server} grids. Both sets of grids outperform MP grids by ~60% at an accuracy target of 1 meV/atom. The runtime for the algorithm to generate GR_{auto} grids at a **k**-point density of 5000 (dense enough to achieve 1 meV/atom accuracy) was ~3 s on average.
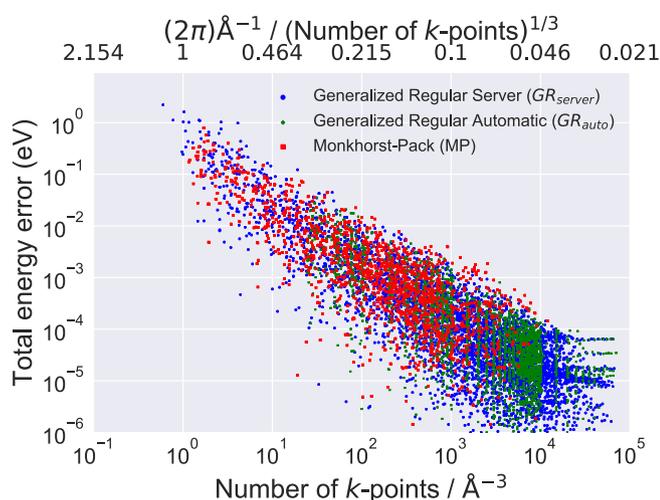
Fig. 4. The total energy convergence with respect to total **k**-point density for MP, GR_{auto} and GR_{server} grids. The top axis shows the linear **k**-point spacing with a factor of 2π included as part of the transformation to reciprocal space. This differs from the linear **k**-point spacing usually used as input in DFT codes by a factor of 2π, i.e., to get the spacing used as input in codes divide the values here by 2π.

4. Conclusion

We have designed an algorithm that generates Generalized Regular (GR) grids "on the fly". These GR_{auto} grids are ~60% more efficient than MP grids at an accuracy target of 1 meV/atom and have similar efficiency to GR_{server} grids [1].

The algorithm is able to reduce the search space for GR grids by only generating grids that preserve the symmetry of the input lattice. The symmetry-preserving grids are then filtered so that only the most efficient grid is returned. For our test cases the average runtime of finding the optimal grid was ~3 s. This algorithm has been implemented and is available for download at: https://github.com/msg-byu/autoGR.

Data availability

The raw data required to reproduce these findings are available to download from https://github.com/msg-byu/autoGR. The processed data required to reproduce these findings are available to download from https://github.com/msg-byu/autoGR.

CRediT authorship contribution statement

Wiley S. Morgan: Conceptualization, Data curation, Formal analysis, Methodology, Software, Visualization, Writing - original draft, Writing - review & editing. **John E. Christensen:** Software. **Parker K. Hamilton:** Software. **Jeremy J. Jorgensen:** Conceptualization, Software, Visualization, Writing - review & editing. **Branton J. Campbell:** Methodology. **Gus L.W. Hart:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Supervision, Validation, Writing - original draft, Writing - review & editing. **Rodney W. Forcade:** Methodology.
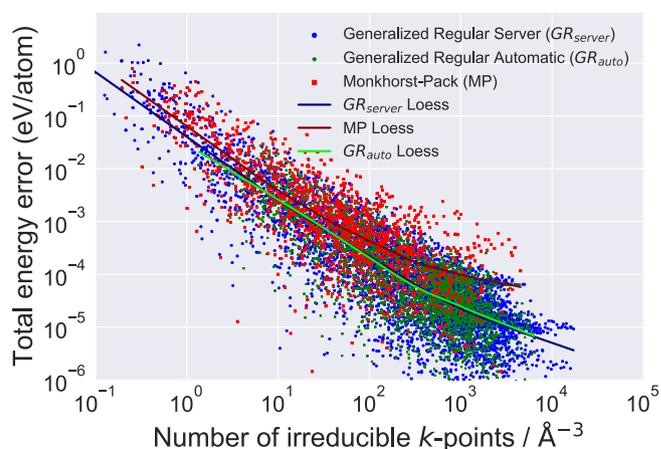
Fig. 5. The total energy convergence with respect to irreducible **k**-point density for MP, GR_{auto} and GR_{server} grids with loess regression applied.
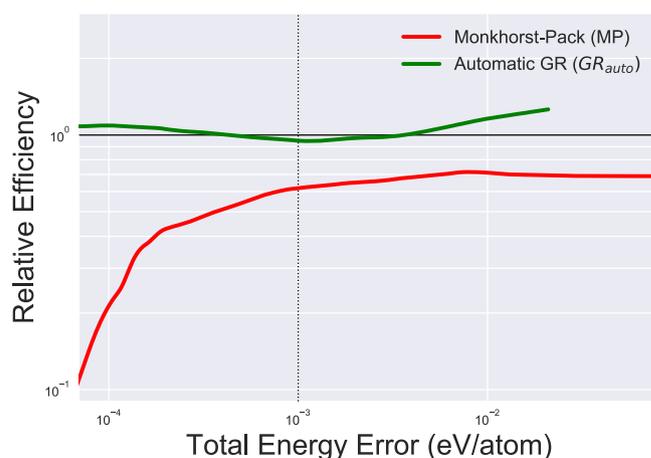
Fig. 6. Along the y-axis are the ratios of the MP and GR_{auto} efficiencies compared to the GR_{server} grid efficiency (black horizontal line at 10^0). Total energy error (per atom) is plotted along the x-axis and decreases to the left. MP grids are ~60% less efficient than both GR_{auto} and GR_{server} grids at a target accuracy of 1 meV/atom. The GR_{auto} grids, however, outperform GR_{server} grids at low densities but otherwise closely agree with GR_{auto} grids.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

A.1. Symmetry-preserving offsets

The following is a table of the symmetry-preserving offsets for each Bravais lattice expressed in terms of fractions of the primitive lattice vectors.

Simple Cubic	$\left(\frac{1}{2},\ \frac{1}{2},\ \frac{1}{2}\right)$
Face Centered Cubic	$\left(\frac{1}{2},\ \frac{1}{2},\ \frac{1}{2}\right)$
Body Centered Cubic	None
Hexagonal	$\left(0,\ 0,\ \frac{1}{2}\right)$
Rhombohedral	$\left(0,\ 0,\ \frac{1}{2}\right)$
Simple Tetragonal	$\begin{pmatrix} 0,\ 0,\ \frac{1}{2} \\ \frac{1}{2},\ \frac{1}{2},\ 0 \\ \frac{1}{2},\ \frac{1}{2},\ \frac{1}{2} \end{pmatrix}$
Body Centered Tetragonal	$\left(0,\ 0,\ \frac{1}{2}\right)$
Simple Orthorhombic	$\begin{pmatrix} 0,\ 0,\ \frac{1}{2} \\ 0,\ \frac{1}{2},\ 0 \\ \frac{1}{2},\ 0,\ 0 \\ 0,\ \frac{1}{2},\ \frac{1}{2} \\ \frac{1}{2},\ 0,\ \frac{1}{2} \\ \frac{1}{2},\ \frac{1}{2},\ 0 \\ \frac{1}{2},\ \frac{1}{2},\ \frac{1}{2} \end{pmatrix}$
Base Centered Orthorhombic	$\begin{pmatrix} 0,\ 0,\ \frac{1}{2} \\ 0,\ \frac{1}{2},\ 0 \\ 0,\ \frac{1}{2},\ \frac{1}{2} \end{pmatrix}$
Face Centered Orthorhombic	$\left(\frac{1}{2},\ \frac{1}{2},\ \frac{1}{2}\right)$
Body Centered Orthorhombic	$\begin{pmatrix} 0,\ 0,\ \frac{1}{2} \\ 0,\ \frac{1}{2},\ 0 \\ \frac{1}{2},\ 0,\ 0 \end{pmatrix}$
Simple Monoclinic	$\begin{pmatrix} 0,\ 0,\ \frac{1}{2} \\ 0,\ \frac{1}{2},\ 0 \\ \frac{1}{2},\ 0,\ 0 \\ 0,\ \frac{1}{2},\ \frac{1}{2}, \\ \frac{1}{2}\ 0,\ \frac{1}{2} \\ \frac{1}{2},\ \frac{1}{2},\ 0 \\ \frac{1}{2},\ \frac{1}{2},\ \frac{1}{2} \end{pmatrix}$
Base Centered Monoclinic	$\begin{pmatrix} 0,\ 0,\ \frac{1}{2} \\ 0,\ \frac{1}{2},\ 0 \\ -\frac{1}{4},\ \frac{1}{2},\ 0 \\ -\frac{1}{4},\ \frac{1}{4},\ \frac{1}{2} \\ \frac{1}{4},\ \frac{1}{4},\ 0 \\ \frac{1}{4},\ \frac{1}{4},\ \frac{1}{2} \\ 0,\ \frac{1}{2},\ \frac{1}{2} \end{pmatrix}$
Triclinic	None

References

[1] P. Wisesa, K.A. McGill, T. Mueller, Phys. Rev. B 93 (2016) 155109.
[2] W.S. Morgan, J.J. Jorgensen, B.C. Hess, G.L. Hart, Comput. Mater. Sci. 153 (2018) 424.
[3] S. Curtarolo, W. Setyawan, G.L. Hart, M. Jahnatek, R.V. Chepulskii, R.H. Taylor, S. Wang, J. Xue, K. Yang, O. Levy, et al., Comput. Mater. 58 (2012) 218.
[4] J.E. Saal, S. Kirklin, M. Aykol, B. Meredig, C. Wolverton, JOM 65 (2013) 1501.
[5] A. Jain, S.P. Ong, G. Hautier, W. Chen, W.D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, et al., APL Mater. 1 (2013) 011002.
[6] S.L. Digabel, C. Tribes, C. Audet, NOMAD User Guide (Tech. Rep. G-2009-37), Les cahiers du GERAD, Quebec, Canada, 2009.
[7] D.D. Landis, J.S. Hummelshøj, S. Nestorov, J. Greeley, M. Dułak, T. Bligaard, J.K. Nørskov, K.W. Jacobsen, Comput. Sci. Eng. 14 (2012) 51.
[8] J. Hachmann, R. Olivares-Amaya, S. Atahan-Evrenk, C. Amador-Bedolla, R.S. Sánchez-Carrera, A. Gold-Parker, L. Vogt, A.M. Brockway, A. Aspuru-Guzik, J. Phys. Chem. Lett. 2 (2011) 2241.
[9] J.S. Hummelshøj, F. Abild-Pedersen, F. Studt, T. Bligaard, J.K. Nørskov, Angew. Chem. 124 (2012) 278.
[10] M. De Jong, W. Chen, H. Geerlings, M. Asta, K.A. Persson, Sci. Data 2 (2015).
[11] M. De Jong, W. Chen, T. Angsten, A. Jain, R. Notestine, A. Gamst, M. Sluiter, C.K. Ande, S. Van Der Zwaag, J.J. Plata, et al., Sci. Data 2 (2015) 150009.
[12] L. Cheng, R.S. Assary, X. Qu, A. Jain, S.P. Ong, N.N. Rajput, K. Persson, L.A. Curtiss, J. Phys. Chem. Lett. 6 (2015) 283.
[13] R. Gómez-Bombarelli, J. Aguilera-Iparraguirre, T.D. Hirzel, D. Duvenaud, D. Maclaurin, M.A. Blood-Forsythe, H.S. Chae, M. Einzinger, D.-G. Ha, T. Wu, et al., Nat. Mater. 15 (2016) 1120.
[14] E.M. Chan, Chem. Soc. Rev. 44 (2015) 1653.
[15] T. Tada, S. Takemoto, S. Matsuishi, H. Hosono, Inorg. Chem. 53 (2014) 10347.
[16] G. Pilania, C. Wang, X. Jiang, S. Rajasekaran, R. Ramprasad, Sci. Rep. 3 (2013).
[17] J. Yan, P. Gorai, B. Ortiz, S. Miller, S.A. Barnett, T. Mason, V. Stevanović, E.S. Toberer, Energy Environ. Sci. 8 (2015) 983.
[18] R. Ramakrishnan, P.O. Dral, M. Rupp, O.A. Von Lilienfeld, Sci. Data 1 (2014) 140022.
[19] J. Hachmann, R. Olivares-Amaya, A. Jinich, A.L. Appleton, M.A. Blood-Forsythe, L.R. Seress, C. Roman-Salgado, K. Trepte, S. Atahan-Evrenk, S. Er, et al., Energy Environ. Sci. 7 (2014) 698.
[20] L.-C. Lin, A.H. Berger, R.L. Martin, J. Kim, J.A. Swisher, K. Jariwala, C.H. Rycroft, A.S. Bhown, M.W. Deem, M. Haranczyk, et al., Nat. Mater. 11 (2012) 633.
[21] R. Armiento, B. Kozinsky, G. Hautier, M. Fornari, G. Ceder, Phys. Rev. B 89 (2014) 134103.
[22] O. Senkov, J. Miller, D. Miracle, C. Woodward, Nat. Commun. 6 (2015).
[23] J. Greeley, T.F. Jaramillo, J. Bonde, I. Chorkendorff, J.K. Nørskov, Nat. Mater. 5 (2006) 909.
[24] R. Gautier, X. Zhang, L. Hu, L. Yu, Y. Lin, T.O. Sunde, D. Chon, K.R. Poeppelmeier, A. Zunger, Nat. Chem. 7 (2015) 308.
[25] A.O. Oliynyk, A. Mar, Acc. Chem. Res. (2017).
[26] H. Chen, G. Hautier, A. Jain, C. Moore, B. Kang, R. Doe, L. Wu, Y. Zhu, Y. Tang, G. Ceder, Chem. Mater. 24 (2012) 2009.
[27] G. Hautier, A. Jain, S.P. Ong, B. Kang, C. Moore, R. Doe, G. Ceder, Chem. Mater. 23 (2011) 3495.
[28] C. Jähne, C. Neef, C. Koo, H.-P. Meyer, R. Klingeler, J. Mater. Chem. A1 (2013) 2856.
[29] T. Moot, O. Isayev, R.W. Call, S.M. McCullough, M. Zemaitis, R. Lopez, J.F. Cahoon, A. Tropsha, Mater. Discov. 6 (2016) 9.
[30] U. Aydemir, J.-H. Pöhls, H. Zhu, G. Hautier, S. Bajaj, Z.M. Gibbs, W. Chen, G. Li, S. Ohno, D. Broberg, et al., J. Mater. Chem. A4 (2016) 2461.
[31] H. Zhu, G. Hautier, U. Aydemir, Z.M. Gibbs, G. Li, S. Bajaj, J.-H. Pöhls, D. Broberg, W. Chen, A. Jain, et al., J. Mater. Chem. C3 (2015) 10554.
[32] W. Chen, J.-H. Pöhls, G. Hautier, D. Broberg, S. Bajaj, U. Aydemir, Z.M. Gibbs, H. Zhu, M. Asta, G.J. Snyder, et al., J. Mater. Chem. C4 (2016) 4414.
[33] G. Ceder, Y.-M. Chiang, D. Sadoway, M. Aydinol, Y.-I. Jang, B. Huang, Nature 392 (1998) 694.
[34] F. Yan, X. Zhang, G.Y. Yonggang, L. Yu, A. Nagaraja, T.O. Mason, A. Zunger, Nat. Commun. 6 (2015).
[35] D. Bende, F.R. Wagner, O. Sichevych, Y. Grin, Angew. Chem. 129 (2017) 1333.
[36] A. Mannodi-Kanakkithodi, A. Chandrasekaran, C. Kim, T.D. Huan, G. Pilania, V. Botu, R. Ramprasad, Mater. Today (2017).
[37] S. Sanvito, C. Oses, J. Xue, A. Tiwari, M. Zic, T. Archer, P. Tozman, M. Venkatesan, M. Coey, S. Curtarolo, Sci. Adv. 3 (2017) e1602241.
[38] H. Yaghoobnejad Asl, A. Choudhury, Chem. Mater. 28 (2016) 5029.
[39] G. Hautier, A. Miglio, G. Ceder, G.-M. Rignanese, X. Gonze, Nat. Commun. 4 (2013) 2292.
[40] A. Bhatia, G. Hautier, T. Nilgianskul, A. Miglio, J. Sun, H.J. Kim, K.H. Kim, S. Chen, G.-M. Rignanese, X. Gonze, et al., Chem. Mater. 28 (2015) 30.
[41] G.H. Johannesson, T. Bligaard, A.V. Ruban, H.L. Skriver, K.W. Jacobsen, J.K. Nørskov, Phys. Rev. Lett. 88 (2002) 255506.
[42] D.P. Stucke, V.H. Crespi, Nano Lett. 3 (2003) 1183.
[43] S. Curtarolo, D. Morgan, G. Ceder, Calphad 29 (2005) 163.
[44] S.F. Matar, I. Baraille, M. Subramanian, Chem. Phys. 355 (2009) 43.
[45] G. Ceder, G. Hautier, A. Jain, S.P. Ong, MRS Bull. 36 (2011) 185.
[46] A.N. Sokolov, S. Atahan-Evrenk, R. Mondal, H.B. Akkerman, R.S. Sánchez-Carrera, S. Granados-Focil, J. Schrier, S.C. Mannsfeld, A.P. Zoombelt, Z. Bao, et al., Nat. Commun. 2 (2011) 437.
[47] Z.W. Ulissi, M.T. Tang, J. Xiao, X. Liu, D.A. Torelli, M. Karamad, K. Cummins, C. Hahn, N.S. Lewis, T.F. Jaramillo, et al., ACS Catal. 7 (2017) 6600.
[48] O. Levy, R.V. Chepulskii, G.L. Hart, S. Curtarolo, JACS 132 (2009) 833.
[49] X. Ma, G. Hautier, A. Jain, R. Doe, G. Ceder, J. Electrochem. Soc. 160 (2013) A279.
[50] K. Yang, W. Setyawan, S. Wang, M.B. Nardelli, S. Curtarolo, Nat. Mater. 11 (2012) 614.
[51] H. Chen, G. Hautier, G. Ceder, JACS 134 (2012) 19619.
[52] S. Kirklin, B. Meredig, C. Wolverton, Adv. Energy Mater. 3 (2013) 252.
[53] H.J. Monkhorst, J.D. Pack, Phys. Rev. B 13 (1976) 5188.
[54] J. Moreno, J.M. Soler, Phys. Rev. B 45 (1992) 13891.
[55] I. Kivý, B. Gruber, Acta Crystallogr. A 32 (1976).
[56] A. Santoro, A.D. Mighell, Acta Crystallogr. A26, 124.https://doi.org/10.1107/S0567739470000177.
[57] R.W. Grosse-Kunstleve, N.K. Sauter, P.D. Adams, Acta Crystallogr. A 60 (2004) 1.
[58] Theo Hahn (Ed.), International Tables for Crystallography. Volume A, Space-Group Symmetry, fifth revised edition, International Union of Crystallography by Kluwer Academic Publishers, Dordrecht; London, 2002.
[59] P.Q. Nguyen, D. Stehlé, D. Buell (Ed.), Algorithmic Number Theory, Springer, Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 338–357.
[60] G.L.W. Hart, J. Jorgensen, W.M. Morgan, R.W. Forcade. arXiv:1809.10261.
[61] G. Kresse, J. Hafner, Phys. Rev. B 47 (1993) 558.
[62] G. Kresse, J. Furthmüller, Comput. Mater. Sci. 6 (1996) 15.
[63] G. Kresse, J. Hafner, Phys. Rev. B 49 (1994) 14251.
[64] G. Kresse, J. Furthmüller, Phys. Rev. B 54 (1996) 11169.
[65] P.E. Blöchl, Phys. Rev. B 50 (1994) 17953.
[66] G. Kresse, D. Joubert, Phys. Rev. B 59 (1999) 1758.
[67] G.L. Hart, R.W. Forcade, Phys. Rev. B 77 (2008) 224115.