A methodology to generate crystal-based molecular structures for atomistic simulations

Christian F A Negre1,*, Andrew Alvarado1,2, Himanshu Singh1, Joshua Finkelstein1, Enrique Martinez2,3 and Romain Perriot1

1 Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, United States of America
2 Department of Mechanical Engineering, Clemson University, Clemson, SC 29623, United States of America
3 Department of Materials Science and Engineering, Clemson University, Clemson, SC 29623, United States of America

E-mail: cnegre@lanl.gov

Received 11 October 2022, revised 23 January 2023
Accepted for publication 8 March 2023
Published 24 March 2023

Abstract
We propose a systematic method to construct crystal-based molecular structures often needed as input for computational chemistry studies. These structures include crystal 'slabs' with periodic boundary conditions (PBCs) and non-periodic solids such as Wulff structures. We also introduce a method to build crystal slabs with orthogonal PBC vectors. These methods are integrated into our code, Los Alamos Crystal Cut (LCC), which is open source and thus fully available to the community. Examples showing the use of these methods are given throughout the manuscript.

Supplementary material for this article is available online

Keywords: quantum chemistry, extended structures, crystal structures, unit cells, miller indices

(Some figures may appear in colour only in the online journal)

1. Introduction
Surface science is essential to understand and predict many physical phenomena including heterogeneous catalysis [1, 2], photo-catalysis [3, 4], material interfaces [5, 6], and optical properties [7, 8]. Surface science is also crucial to study the shape and properties of nanocrystals [9], which are essential to quantum dot applications [10, 11], and 2D materials, exhibiting exceptional electrical, optical and mechanical properties [12, 13]. Beyond physical chemistry, surface science also plays a crucial role in biomedical [14] and bioengineering [15] applications, and dictates crystal growth [16], which is known to affect, for instance, the performance of high explosives [17, 18]. Moreover, in all applications where the material exhibits a high surface to volume ratio, the properties of the surface (exposed crystal faces) largely determine the properties of the material.

Despite the progress of characterization techniques, simulations remain a fundamental part of surface science, either to complement [19] or fully predict [20, 21] the properties of surfaces, interfaces, and nanoparticles. Electronic transport,
optical properties, and even surface reconstructions can be a significant challenge for empirical models and, in order to perform these simulations, *ab initio* level of theory is often required due to the complexity of the phenomena involved [22]. In most cases, the first step involved in these calculations will include building a model crystal slab, which should obey the following constraints: the system must give us access to the surfaces of interest to the particular problem, it must be periodic in all other directions, and, in order to improve computational efficiency, it must be as small as possible (electronic structure calculations are usually performed on hundreds to a few thousands atoms, at most). This results in a crystal-based parallelepiped with planes that are not necessarily orthogonal to each other since, in the general case, the unit cell is triclinic (i.e. the lattice vectors are non-orthogonal to each other with differing lengths and angles to one another).

In order to study crystal surfaces with quantum chemistry methods, it is often necessary to have a crystal slab cut through planes that expose the face one wants to study and that also satisfies the periodic boundary conditions (PBCs) imposed by the crystal unit cell. A typical minimalistic system is depicted in figure 1. This is a z-axis view of a $3 \times 3 \times 3 = 27$ monoclinic unit cell system showing the slab PBC vectors. In this case the $p_2$ vector was enlarged in order to have some vacuum that could expose the surface of interest. At first, this system may not seem complicated as it can be easily built with an *ad-hoc* procedure using an off-the-shelf molecular visualization tool. There are however, many cases in which building such a system in this way could turn into a complicated and time consuming endeavor. Triclinic unit cells exposing some crystal face with large Miller indices fall within this category. Moreover, a lot of time and effort is consumed when errors in the simulations arise due to an ill-chosen system slab. How do we then proceed to construct any desired crystallographic system by just knowing the basic crystallographic data? In this article, we explain a method based on purely algebraic/geometrical transformations that leads to a sufficiently small crystal slab exposing the desired crystal faces. We would like to offer a detailed and simple step-by-step procedure that the readers could fully code up on their own. Moreover, the method developed in this paper can also be used to construct Wulff type of structures provided that the lattice periodicity mismatch is used to assess the validity of the slab.

In addition to GCCM [26] and the method presented in [23], many other commercial and open source tools pertaining to crystal manipulation/building exist. Codes such as CrystalMaker [27], Avogadro [28], Ovito [29], XCRYSDEN [30], SCM [31], VESTA [32], and Atomsk [33], just to mention a few, have some sort of crystal manipulation tool. Except for the methods described in [26] and [23], the remaining codes we know of do not explicitly state the equations used or are completely proprietary codes; therefore, it is difficult to understand what technique is actually being used. Also, while these codes can handle many tasks unavailable in our LCC code (building polycrystalline samples or grain interfaces as in Atomsk, for instance), many present some limitations in the domain of crystalline slab building (the core purpose of LCC), whether related to the type of crystal available (cubic, tetragonal, monoclinic, triclinic, etc), the option to impose periodicity and/or orthogonality, or the inclusion of molecular crystals.

**Figure 1.** Representation of a typical system slab needed to study a particular physical chemistry surface property. The system slab is composed of lattice points that are illustrated as bright magenta spheres together with the slab PBC vectors $p_1$ and $p_2$.
The following sections are organized as follows: We first introduce some basic crystallographic concepts in order to keep consistent notation throughout the manuscript. In section 3 we introduce our method to cut a crystal lattice, and in section 3.2 we develop the techniques to determine the PBC vectors. Section 4 is dedicated to explaining how the method can be used to construct non-periodic solids using Wulff structures as an example. Finally, in section 5 we explain a method to construct crystal slabs with orthogonal PBC vectors. Sections in the Appendices are used for support and clarification throughout the text. Units of length and angles used in all the examples are in Angstroms (Å) and degrees (°) respectively.

2. Background

A crystal lattice is a set of points \( L \subseteq \mathbb{R}^3 \) that is fully determined by the primitive unit cell described by the lattice vectors \( \mathbf{a}_1, \mathbf{a}_2, \) and \( \mathbf{a}_3 \). For any point \( \mathbf{r} \) belonging to \( L \), there exist three integers, \( n_1, n_2, n_3 \) such that:

\[
\mathbf{r} = n_1 \mathbf{a}_1 + n_2 \mathbf{a}_2 + n_3 \mathbf{a}_3.
\]

Formally, \( L = \{ \mathbf{r} \in \mathbb{R}^3 \mid n_1, n_2, n_3 \in \mathbb{Z} \} \). A conventional unit cell (such as the cubical systems by Bravais), is just a more elaborate cell in which symmetry is increased. This increase in symmetry in some cases will, for instance, render lattice vectors that are orthogonal to each other; a highly desirable property for many applications. Regardless of which type of cell we have, the convention in crystallography is to report the so-called lattice parameters \( a, b, c, \alpha, \beta, \) and \( \gamma \); where \( a, b, \) and \( c \) are the lengths of lattice vectors \( \mathbf{a}_1, \mathbf{a}_2, \) and \( \mathbf{a}_3 \), respectively, and \( \alpha, \beta, \gamma \) are, respectively, the angle between vectors \( \mathbf{a}_2 \) and \( \mathbf{a}_1 \), \( \mathbf{a}_3 \) and \( \mathbf{a}_1 \), and \( \mathbf{a}_1 \) and \( \mathbf{a}_2 \) [34]. This reduces the arbitrariness of having to choose a lattice orientation given by the lattice vectors. Note that if the lattice is rotated, our lattice vectors will need to be rotated as well, whereas the lattice parameters will stay the same. Finally, a full representation of the system needs a ‘basis,’ which is the minimal molecular fragment contained by each unit cell. It is common to express the coordinates of the basis in fractions of the lattice vectors. By choosing this coordinate system we make the orientation of the basis invariant to lattice rotations.

Although working with lattice parameters has some advantages, it is convenient to compute the lattice vectors in order to do all the necessary transformations to build a PBC slab. In order to compute the lattice vectors from the lattice parameters one needs to apply the following transformations:

\[
a_{1x} = a, \\
a_{1y} = 0, \\
a_{1z} = 0,
\]

\[
a_{2x} = b \cos(\gamma), \\
a_{2y} = b \sin(\gamma), \\
a_{2z} = 0,
\]

\[
a_{3x} = c \cos(\beta), \\
a_{3y} = c \left( \cos(\alpha) - \cos(\gamma) \cos(\beta) \right), \\
a_{3z} = \sqrt{c^2 - a_{3x}^2 - a_{3y}^2},
\]

where we have arbitrarily set \( \mathbf{a}_1 \) to be aligned with the \( x \)-axis, or in more formal terms, the first canonical vector \( \mathbf{e}_1 = (1,0,0) \) in the canonical basis for \( \mathbb{R}^3 \). An equivalent reverse transformation is used to compute the parameters given the lattice vectors:

\[
a = \sqrt{(a_{1x}^2 + a_{1y}^2 + a_{1z}^2)}, \\
b = \sqrt{(a_{2x}^2 + a_{2y}^2 + a_{2z}^2)}, \\
c = \sqrt{(a_{3x}^2 + a_{3y}^2 + a_{3z}^2)},
\]

\[
\gamma = \frac{360}{2 \pi} \cos^{-1}(\langle \mathbf{a}_1, \mathbf{a}_2 \rangle / (ab)), \\
\beta = \frac{360}{2 \pi} \cos^{-1}(\langle \mathbf{a}_1, \mathbf{a}_3 \rangle / (ac)), \\
\alpha = \frac{360}{2 \pi} \cos^{-1}(\langle \mathbf{a}_2, \mathbf{a}_3 \rangle / (bc)).
\]

3. Building extended systems

Using the lattice vectors, a crystal slab can be built simply by adding lattice points according to equation (1) for a finite number of \( n_i \)’s. The resulting slab would expose the crystalline faces as well as the respective opposite faces given by \( (0h0), (00l) \) and \( (00k) \). Note that this slab will form a parallelepiped whose edge directions are not necessarily orthogonal to one another in \( E \equiv \{ e_1, e_2, e_3 \} \), the standard basis for \( \mathbb{R}^3 \). We shall call this slab the ‘canonical slab.’ An example canonical slab of the monoclinic phase of benzene is shown in figure 2. At this point, a natural question emerges. What if now we need to expose other crystalline faces to perform specific computational physico-chemical studies? In this case, the periodicity of the slab will be key. The strategy we follow in this section is to cut out a crystal slab using Miller planes directly (planes given by a specific set of Miller indices) and then determine the PBC vectors for this slab: \( \mathbf{p}, \mathbf{p}_1, \) and \( \mathbf{p}_2 \). The Miller planes are determined by the normal vectors perpendicular to the desired crystal faces. A Miller plane \( (hk l) \) has normal vector \( \mathbf{n}_{(hk l)} = h \mathbf{b}_1 + k \mathbf{b}_2 + l \mathbf{b}_3 \), where the reciprocal lattice vectors \( \mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3 \) are defined to be:

\[
\mathbf{b}_1 \equiv \frac{2 \pi}{V} \mathbf{a}_2 \times \mathbf{a}_3, \\
\mathbf{b}_2 \equiv \frac{2 \pi}{V} \mathbf{a}_3 \times \mathbf{a}_1, \\
\mathbf{b}_3 \equiv \frac{2 \pi}{V} \mathbf{a}_1 \times \mathbf{a}_2,
\]

where \( V = |\mathbf{a}_1 \cdot \mathbf{a}_2 \times \mathbf{a}_3| \) is the volume of the unit cell. It is easy to see from the definition of the reciprocal vectors above that \( \{ \mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3 \} \) are biorthogonal to \( \{ \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3 \} \), i.e. \( \mathbf{a}_i \cdot \mathbf{b}_j = 0 \) if \( i \neq j \).
For a cut by any plane \( \Pi = \{ \mathbf{r} \in \mathbb{R}^3 \mid (\mathbf{r} - \mathbf{Q}) \cdot \mathbf{N} = 0 \} \) defined by a point \( \mathbf{Q} \) and normal \( \mathbf{N} \), selecting all lattice points \( \mathbf{r} \) (all \( \mathbf{r} \in \Pi \)) such that \( (\mathbf{r} - \mathbf{Q}) \cdot \mathbf{N} < 0 \), the set containing all lattice points lying ‘below’ the plane \( \Pi \) is given by \( \mathbf{r} \in \mathbb{R}^3 \mid (\mathbf{r} - \mathbf{Q}) \cdot \mathbf{N} = 0 \). We then apply this to the particular case of Miller planes in order to build out the desired slab.

We now discuss a general algorithm to perform a cut by any plane and then apply this to the particular case of Miller planes in order to build out the desired slab.

To do this we just need to evaluate the sign of the inner product between the vector \( \mathbf{r} - \mathbf{Q} \) and the normal vector \( \mathbf{N} \). If \( (\mathbf{r} - \mathbf{Q}) \cdot \mathbf{N} < 0 \), then we have:

\[
T = \frac{2\pi}{h \| \mathbf{b}_1 \|} = \frac{2\pi}{2\| \mathbf{a}_1 \times \mathbf{a}_2 \| \cos(\nu)} = \| \mathbf{a}_1 \| \cos(\nu),
\]

so that the periodicity in the direction normal to the (100) face, i.e. the distance between two adjacent planes with Miller indices (100), will then be \( a \cos(\nu) \); where \( \nu \) is the angle between \( \mathbf{a}_1 \times \mathbf{a}_2 \) and \( a \) is the lattice parameter in equation (2).

We can hence normalize the \( \mathbf{N}_{(hkl)} \) direction and cut (select the points below the (hkl) plane) using the expression in equation (8) with \( \mathbf{n} = t\mathbf{N} \) and \( \mathbf{N} \) the unit normal. As \( t \) varies, this will select all planes along the \( \mathbf{N}_{(hkl)} \) direction towards the origin ‘below’ the (hkl) plane. This constitutes a full procedure to cut a crystal shape using Miller planes. Note that the only input variables apart from the lattice parameters are the (hkl) indices and the scaling factor \( t \).
the lattice vectors, the full crystal structure system coordinates will be \[ \{ r_i^j = x_i^j a_1 + y_i^j a_2 + z_i^j a_3 \}_{1 \leq i \leq N, 1 \leq j \leq M}. \]

### 3.2. Computing PBC vectors

In order to construct a crystal slab, we need to define all the plane boundaries that will form the parallelepiped or PBC cell. To do so, we cut using a total of six Miller planes. Given a Miller plane \((hkl)\) and its normal vector \(N_{(hkl)}\), we seek to find two additional Miller planes \((h_1 k_1 l_1)\) and \((h_2 k_2 l_2)\), yielding two additional normal vectors \(N_1\) and \(N_2\), such that all three normal vectors, \(N_{(hkl)}\), \(N_1\), and \(N_2\) are perpendicular to each other with respect to the reciprocal basis \(B \equiv \{ b_1, b_2, b_3 \}\).

Note that the vectors \(N_{(hkl)}, N_1, \) and \(N_2\), when expressed in the canonical basis set \(\mathcal{E}\), might not be orthogonal to one another since \(\mathcal{B}\) is not necessarily an orthogonal basis set, e.g. in the case of a monoclinic or triclinic unit cell. If we have a Miller plane \((hkl)\) with normal vector \(N_{(hkl)} = \alpha b_1 + \beta b_2 + \gamma b_3\), we can, without loss of generality, assume that \(\alpha \neq 0\) and pick a first vector \(N_1 = \alpha b_1 + \beta b_2 + l_1 b_3\) perpendicular to it in the basis \(\mathcal{B}\) by setting \(l_1 = (-\alpha k_1 - \beta l_1)/\gamma\) for given \(k_1\) and \(l_1\). This is an immediate consequence of solving for \(h_1\) in the equation \((h_1, k_1, l_1) \cdot (h_2, k_2, l_2) = 0\). Since \(k_1\) and \(l_1\) are free parameters, we choose to set them to 1 and 0 respectively to get \(h_1 = -k/h\). We can also compute the entries of another vector \(N_2 = (h_2 b_1 + h_3 b_2 + l_2 b_3)\) orthogonal to \(N\) and \(N_1\) that will have components:

\[
\begin{align*}
h_2 &= \frac{k^2 l / h^3}{k^2 / h^2 + 1} - l / h, \\
k_2 &= \frac{-kl / h^2}{k^2 / h^2 + 1}, \\
l_2 &= 1,
\end{align*}
\]

here \(l_2\) is also a free parameter that was set to 1 for convenience. The equations in (10) come from solving for \(h_2, k_2, l_2\) using the system of equations given by \(\{ [N_{(hkl)}]_{\mathbb{R}} \cdot [N_2]_{\mathbb{R}} = 0, [N_1]_{\mathbb{R}} \cdot [N_2]_{\mathbb{R}} = 0 \}\), where we use \([\cdot]_{\mathbb{R}}\) to denote that a vector is expressed in its \(\mathcal{B}\) basis representation. In the case where \(h\) is zero we can always permute two coordinates, apply the formulas and permute back. If fractional numbers are obtained from computing \(h_1, h_2, k_2\) or \(l_2\), one can always divide by the minimum value that was obtained for those entries that are non-zero.

A pseudocode implementing this procedure can be found in appendix A. There are many alternative ways of obtaining two orthogonal vectors to a particular \(N_{(hkl)}\) direction; here we have only proposed one such technique. This way of constructing orthogonal directions allows us to define slabs that are bounded by a parallelepiped constructed out of the corresponding Miller planes. For example, if \((hkl) = (110)\), then the procedure just described produces \((h_1 k_1 l_1) = (\bar{1} 1 0)\) and \((h_2 k_2 l_2) = (00\bar{1})\).

We then cut a slab using the procedure explained above which finds the appropriate bounding planes of a parallelepiped. The faces of this parallelepiped will have normal vectors:

\[
\begin{align*}
n^+ &= r T \hat{N}, \\
n^- &= -r T \hat{N}, \\
n_1^+ &= s T \hat{N}, \\
n_1^- &= -s T \hat{N}, \\
n_2^+ &= r T \hat{N}, \\
n_2^- &= -r T \hat{N},
\end{align*}
\]

and by varying over the parameters \(t, s, r\) and \(r\), we change the aspect ratio as well as the volume of the slab we construct. Once the slab is cut, we then need to find its PBC vectors knowing only the \(h, k, l\) indices with which the faces of the solid were cut. Suppose we have a situation like the one depicted in figure 5 where we have performed two cuts using \(n^+\) and \(n^-\). The vector \(v\) that makes up half of the PBC vector ends right on the surface of the solid; on the plane with normal vector \(n\). The vector \(v\) is then in the ‘reciprocal’ direction and can be computed as:

\[
\begin{align*}
\hat{v}_1 &= (n_2 - n_2) / \| n_2 \times n_2 \|, \\
\hat{v}_2 &= (n_1 - n_1) / \| n_1 \times n_1 \|.
\end{align*}
\]

Then, if \(\mathbf{v} = x \hat{v}\), we have that \((x \hat{v} - n) \cdot n = 0\), from which \(x\) can be solved, leading to \(x = \| n^2 \mathbf{v} / (\hat{v} \cdot n) \). We therefore take the PBC vector \(\mathbf{p}\) to be, \(\mathbf{p} = 2(\| n^2 \mathbf{v} ) / (\hat{v} \cdot n)\) and similarly:

\[
\begin{align*}
p_1 &= 2(\| n_1^2 \mathbf{v} / (\hat{v} \cdot n_1)), \\
p_2 &= 2(\| n_2^2 \mathbf{v} / (\hat{v} \cdot n_2)).
\end{align*}
\]

The result is the set of PBC vectors for our newly created slab.

### 3.3. Reorientation of the surface

We often would like to reorient a particular surface so that the surface normal vector aligns with a particular canonical vector. Let \(\mathbf{p}\) be the vector we would like to be aligned with \(\mathbf{e}_1 = (1, 0, 0)\). This is important because one can easily and artificially include vacuum by enlarging the length of the \(\mathbf{p}\) vector yet keeping the same orientation. In order to do this reorientation we will need a linear transformation \(\mathcal{M}\) that takes \(\mathbf{p} \rightarrow || \mathbf{p} \parallel \mathbf{e}_1\). One way to do this is to apply a transformation
that gives us parameters $A, B, C, \Theta, \Phi, \Psi$ similar to what was explained for the lattice parameters in section 2. We define the transformation $T^\tau : \{p, p_1, p_2\} \to \{A, B, C, \Theta, \Phi, \Psi\}$ by:

$$
A = \|p\|, \\
B = \|p_1\|, \\
C = \|p_2\|, \\
\Theta = \frac{360}{2\pi} \arccos \left( \frac{\langle p \cdot p_1 \rangle}{\|AB\|} \right), \\
\Phi = \frac{360}{2\pi} \arccos \left( \frac{\langle p \cdot p_2 \rangle}{\|AC\|} \right), \\
\Psi = \frac{360}{2\pi} \arccos \left( \frac{\langle p \cdot p_2 \rangle}{\|BC\|} \right), \\
$$

which is a coordinate system that is independent of the orientation of the slab. We can define the 'back-transformation' $T^\psi : \{A, B, C, \Theta, \Phi, \Psi\} \to \{p^\|, p_1^\|, p_2^\|\}$ that will give back a reoriented set of PBC vectors parallel to $e_1$ as follows:

$$
p^\| = Ae_1 = A(1,0,0),
$$

$$
p_1^\| = B(\cos(\Theta), \sin(\Theta), 0),
$$

$$
p_2^\| = C \cos(\Phi),
$$

$$
p_{2x}^\| = C \cos(\Phi),
$$

$$
p_{2y}^\| = C \left( \frac{\cos(\Psi) - \cos(\Theta) \cos(\Phi)}{\sin(\Theta)} \right),
$$

$$
p_{2z}^\| = \sqrt{C^2 - \left( p_{2x}^\| \right)^2 - \left( p_{2y}^\| \right)^2},
$$

note that the composition of these two transformations $M = T^\psi T^\tau : \{p, p_1, p_2\} \to \{p^\|, p_1^\|, p_2^\|\}$ will lead to a vector $p^\|$ aligned with $e_1$. The action of the transformation $M$ can then be expressed using matrix algebra through:

$$
\begin{pmatrix}
p_x & p_{1x} & p_{2x} \\
p_y & p_{1y} & p_{2y} \\
p_z & p_{1z} & p_{2z}
\end{pmatrix}
\begin{pmatrix}
p^\|_x \\
p^\|_y \\
p^\|_z
\end{pmatrix}
= 
\begin{pmatrix}
p^\|_x & p^\|_y & p^\|_z \\
p_1^\|_x & p_1^\|_y & p_1^\|_z \\
p_2^\|_x & p_2^\|_y & p_2^\|_z
\end{pmatrix}
\begin{pmatrix}
p_x \\
p_y \\
p_z
\end{pmatrix}
\begin{pmatrix}
p_{1x} & p_{1y} & p_{1z} \\
p_{2x} & p_{2y} & p_{2z}
\end{pmatrix}^{-1},
$$

so that,

$$
M = 
\begin{pmatrix}
p^\|_x & p^\|_y & p^\|_z \\
p_1^\|_x & p_1^\|_y & p_1^\|_z \\
p_2^\|_x & p_2^\|_y & p_2^\|_z
\end{pmatrix}
\times
\begin{pmatrix}
p_x & p_{1x} & p_{2x} \\
p_y & p_{1y} & p_{2y} \\
p_z & p_{1z} & p_{2z}
\end{pmatrix}^{-1}.
$$

Therefore the coordinates of every point $r$ from the original slab can be transformed using $M$ as $(r^\|)^\prime = Mr^\prime$.

4. Building non-periodic solids

The technique to cut by planes explained above allows us to construct any crystalline convex polyhedron just by simply using a list of planes and their distance to the origin. These planes could have any direction $\mathbf{N}$ (or if they are Miller planes, any specific $\mathbf{N}(hkl)$ direction). A crystal slab could also be viewed as a particular crystalline convex polyhedron in which the shape is a parallelepiped.

Crystalline convex polyhedra can also be cut out by taking some aspect of crystal growth into consideration in order to predict and visualize equilibrium crystal shapes. Observations by Wulff concluded that there is a relationship between the extension of the exposed crystal surface and the speed at which it grows [35, 36]. Different surfaces grow at different rates which ultimately determines the extension of each of the exposed surfaces. Wulff realized that the slower a surface grows, the more extended it will appear in the final crystal shape when thermodynamic equilibrium is reached [35, 36].

If a face $a$ with direction $\mathbf{N}_a$ grows faster than another face $b$ with direction $\mathbf{N}_b$ then, to have an idea of the crystal shape, the plane boundary with normal $\mathbf{N}_a$ will be placed much farther away from the origin than the plane boundary with $\mathbf{N}_b$. In both cases the distances to the origin at which a plane should be used to cut the crystal will be directly proportional to the growth rate. Donnay and Harker’s law [37] expresses that the surface growth rates are inversely proportional to the distance between adjacent $(hkl)$ planes, $d_{hkl}$. This law, combined with Wulff’s observations leads to the Bravais-Friedel-Donnay-Harker (BFDH) criterion [38] which gives a rule for constructing the equilibrium crystal shape based solely on crystallographic data where plane positions are set such that the distance to the origin is proportional to $1/d_{hkl}$. The Wulff crystal convex polyhedron based on BFDH method could be hence defined as:

$$
\left\{ \mathbf{r} \in \mathbb{R}^3 \mid \mathbf{r} = \frac{\alpha}{d_{hkl}} \mathbf{N}, \forall (hkl) \text{ planes} \right\},
$$

where:

$$
\mathbf{N} = \frac{h\mathbf{b}_1 + k\mathbf{b}_2 + l\mathbf{b}_3}{\|h\mathbf{b}_1 + k\mathbf{b}_2 + l\mathbf{b}_3\|}.
$$

and $\alpha$ is some proportionality constant. Using our code is thus trivial to build Wulff structures predicted by the BFDH theory (including plane selection rules based on reflection conditions [39]). In figure 6, we illustrate this by comparing the predicted morphologies of an energetic crystal, monoclinic $\beta$-1,3,5,7-tetranitro-1,3,5,7-tetrazocane ($\beta$-HMX, space group $P2_1/n$, CSD [40] entry OCHTET13), and two proposed surrogates [41], triclinic 5-Iodo-2'-deoxyuridine (IDOX, P1, IDOXUR) and monoclinic 2,3,4,5,6-Pentafluorobenzamide (PFBA, $P2_1/c$, VATNOU). We note that the Wulff structure for $\beta$-HMX shown in figure 6 is the same as presented in references [42, 43], which is also based on the BFDH theory. The three crystals exhibit markedly different shapes and exposed faces, due to distinct crystal structures and symmetries. More complex models, incorporating for instance explicitly calculated surface energies, can also be implemented.
The obtained shapes determine the most important crystalline faces that can then be compared to experimental observations of real crystals [42].

5. Building a crystal slab with orthogonal PBC vectors

In this section we detail a new technique to generate an orthogonal slab exposing a desired \((hkl)\) ‘front’ plane while maintaining periodicity of the lattice. Here we have the opposite viewpoint of the previous section, we now want to first construct the PBC vectors, and then from these, construct the orthogonal slab. The periodicity is maintained up to a user-specified tolerance of how well the orthogonal slab conforms to the lattice vectors. How this can be measured will be explained below. All vectors in this section are expressed in the canonical basis.

To construct the slab, three PBC vectors must be selected. The first vector, \(\mathbf{p}\), will have the direction of the prescribed normal vector \(N_{(hkl)}\) of the given \((hkl)\) plane so that \(\mathbf{p} = c N_{(hkl)}\) for some scalar \(c\). The two remaining PBC vectors, \(\mathbf{p}_1\) and \(\mathbf{p}_2\), are chosen so that all three PBC vectors are orthogonal to one another. To find these orthogonal vectors we use a similar procedure to the one described in section 3.2. Having the vector \(\mathbf{p}\) in hand, we choose the second vector, \(\mathbf{p}_1\), so that \(\mathbf{p} \cdot \mathbf{p}_1 = 0\). For this we define the \(x\)-component of \(\mathbf{p}_1\) as:

\[
\mathbf{p}_{1, x} = -\frac{\mathbf{p}_2}{\mathbf{p}_1},
\]

where the free parameters \(\mathbf{p}_{1, y}\) and \(\mathbf{p}_{1, z}\) have been set to 1 and 0 respectively for convenience. Note that if \(\mathbf{p}\) was chosen such that \(\mathbf{p}_1 = 0\), a relabeling of the axes can be made. The third vector \(\mathbf{p}_2\) can be generated by requiring that \(\mathbf{p} \cdot \mathbf{p}_2 = 0\) and \(\mathbf{p}_1 \cdot \mathbf{p}_2 = 0\) to get:

\[
\mathbf{p}_{2, x} = -\frac{\mathbf{p}_2^3 / \mathbf{p}_1^3 - \mathbf{p}_2}{\mathbf{p}_2^2 / \mathbf{p}_1^2 + 1},
\]

\[
\mathbf{p}_{2, y} = -\frac{\mathbf{p}_2^3 / \mathbf{p}_1^3 - \mathbf{p}_2}{\mathbf{p}_2^2 / \mathbf{p}_1^2 + 1},
\]

where the free parameter \(p_{2, z}\) has been set to 1. Note that by setting different parameters one will obtain different equations which will give orthogonal vectors with different directions. Once a pair of orthogonal vectors \(\mathbf{p}_1\) and \(\mathbf{p}_2\) are obtained, a rotation around the vector \(\mathbf{p}\) can be applied to explore other directions.

5.1. Minimum translation, integer search

Given the vectors \(\mathbf{p}, \mathbf{p}_1, \mathbf{p}_2\) just constructed, we now seek to express a certain rescaling of each of the vectors as a linear combination of the lattice vectors in order to comply with periodicity. For any real number \(t > 0\), a vector \(t\mathbf{p}\) fully respects the periodicity of the unit cell if \(r\mathbf{p}\) can be written as an integer linear combination of the lattice vectors. This condition can be represented by the linear system of equations:

\[
\begin{align*}
\mathbf{r}_x &= a_{1,x} + a_{2,x}b + a_{3,x}c, \\
\mathbf{r}_y &= a_{1,y} + a_{2,y}b + a_{3,y}c, \\
\mathbf{r}_z &= a_{1,z} + a_{2,z}b + a_{3,z}c.
\end{align*}
\]

which solves equation (23). We can then determine for each parameter \(t\) in some pre-defined range how close the \(a(t), b(t),\) and \(c(t)\) translation coefficients are to integers. In other words,
the optimal choice of $t$ needs to result in translation coefficients $a(t), b(t)$ and $c(t)$ that minimize the quantity:

$$
ε(t) = |a(t) - \text{Rnd}(a(t))| + |b(t) - \text{Rnd}(b(t))| + |c(t) - \text{Rnd}(c(t))|,
$$

(26)

where the round-off function

$$\text{Rnd}(x) = \begin{cases} 
[x], & x - [x] < \frac{1}{2}, \\
[x] + 1, & x - [x] \geq \frac{1}{2},
\end{cases}
$$

(27)

takes the closest integer to the argument $x$. Here, $[x]$ and $\lfloor x \rfloor$, denote the floor and ceiling of the real number $x$, respectively. The function $ε$ has a maximum of 1.5 at non-integers and a minimum of 0 when $a, b$ and $c$ are all integers, indicating a perfect periodic arrangement in the given direction.

It is in fact the case that $t$ in equation (25) can be chosen such that $a, b$ and $c$ are exactly integers, however this choice of $t$ may be unacceptably large. Assuming that the entries of $\text{M}$ and $\text{p}$ are rational, the determinant formula for a $3 \times 3$ matrix says that $\text{M}^{-1}$ should also have rational entries. Hence, if $(\text{M}^{-1})_{ij} = m_{ij}/q_{ij}$ and $p_i = u_i/v_i$ are such that the numerators and denominators are relatively prime, i.e. have no common factors, then selecting $t = \prod (q_{(i|m_{ij}=0)},q_{ij} \prod (i|m_{ij}=0)) v_i$ will ensure that the $a, b, c$ translation coefficients are all integers.

Our approach will thus need to balance the size of the slab with the error in the periodicity.

In order to construct the desired orthogonal slab, the above procedure is carried out for each PBC vector over the $t, s$ and $r$ parameter space such that $ε$ is minimized. Using equation (25), we do a complete search over the parameter space for some specified range of values. Both the $p_1$ and $p_2$ vectors can also be rotated by an angle $Δα$ about $p$ to further improve the error during the search. This may result in potentially better $s$ and $r$ parameters, where better here means, lower values of $s$ and $r$ for a given tolerance since that would require a smaller number of atoms needing to be simulated (see figure 7).

Equation (26) is therefore calculated for each angle $α$, at every value of $s$ and $r$. The total dimension of the parameter space to be searched is then four — $t, s, r$ and $α$ — and the time-to-solution will scale as $N_s N_r N_α$, where $N_α = 180/(Δα)$ and $N_s = (x_{\text{max}} - x_{\text{min}})/Δx$ for $x \in \{t, s, r\}$. The max and min values for each of the parameter search can be systematically increased or decreased if the error falls above the desired tolerance after a complete search.

Once the PBC vectors are rescaled by the $t, s$ and $r$ values that were found, we then cut the slab by determining which lattice points lie inside the parallelepiped defined by the orthogonal PBC vectors. The idea is to first transform the lattice point coordinates into their PBC vector basis representation. Given a point $r$ expressed in the canonical basis set, we want a transformation $T$ such that $T r = r'_{\text{PBC}} \text{p} + r'_{\text{PBC}} \text{p}_1 + r'_{\text{PBC}} \text{p}_2$, that is, we want to solve the system:

$$
\begin{align*}
   r_x &= r'_{\text{PBC}} p_x + r'_{\text{PBC}} p_{1x} + r'_{\text{PBC}} p_{2x} \\
   r_y &= r'_{\text{PBC}} p_y + r'_{\text{PBC}} p_{1y} + r'_{\text{PBC}} p_{2y} \\
   r_z &= r'_{\text{PBC}} p_z + r'_{\text{PBC}} p_{1z} + r'_{\text{PBC}} p_{2z}
\end{align*}
$$

(28)

Therefore, $T$ will be given by the change of basis matrix:

$$
T = \begin{pmatrix} p_x & p_{1x} & p_{2x} \\
p_y & p_{1y} & p_{2y} \\
p_z & p_{1z} & p_{2z} \end{pmatrix},
$$

(29)

so that by inverting $T$ and applying it to $r$, we can determine the PBC basis coordinates that we need. We then only accept the coordinates $(r'_{\text{PBC}}, r''_{\text{PBC}}, r'''_{\text{PBC}})$ if $0 < r'_{\text{PBC}} < 1$, $α = x, y, z$. Hence, the set of lattice points to be included in the slab are:

$$\{ r : 0 < (Tr)_{x,y,z} < 1 \}.
$$

(30)

5.2. Example of orthogonal PBC slabs

In this section we demonstrate how to use the method described in section 5.1 to construct a slab with orthogonal PBC vectors by applying it to the $β$-HMX polymorph. $β$-HMX is a monoclinic crystal with a $P2_1/n$ space group. Figure 8(a) depicts the $β$-HMX unit cell containing two HMX molecules for a total of 56 atoms. The lattice vectors expressed in the canonical basis set are: $\text{a}_1 = (6.53, 0, 0)$, $\text{a}_2 = (0, 11.02, 0)$, and $\text{a}_3 = (1.61, 0, 7.18)$.

Given an initial direction of a desired exposed surface (not necessarily generated by a Miller plane) such as for example $\text{p} = (1, 1, 0)$, the orthogonal normal vectors generated by the procedure explained in section 5.1, are $\text{p}_1 = (-1, 1, 0)$ and $\text{p}_2 = (0, 0, 1)$. Applying a full parameter search with an error computed as in equations (25) and (26), we find scaling factors of $t = 32.66$, $s = 32.66$, and $r = 28.71$ for $\text{p}_1$, $\text{p}_1$, and $\text{p}_2$, respectively. These scalar lengths are found within a range of 1 to 100 using a step size of $Δ = 10^{-2}$ and an error tolerance of $δ = 10^{-1}$. Typically, the smaller we set the tolerance $δ$, the larger the resulting volume will be. Although this tolerance may not seem so small, we again must decide how large of a periodic cell we are willing to use. Substantially increasing the range of the parameters would result in smaller errors though at the cost of more atoms to simulate. In order to speed up the search for the scaling factor, one can use multiples
In figure 8(c), we can see that the minimum error for both $\mathbf{p}$ and $\mathbf{p}_1$ is achieved at parameter value $t = s = 32.6$, and the first minimum for $\mathbf{p}_2$ is found at $r = 7.17$. These minima are repeated at integer multiples of these values thereafter. For vector $\mathbf{p}_2$, the error minima correspond to vector lengths that are approximately equal to multiples of the $z$ component value of the third lattice vector for HMX. Due to the error tolerance that was set, the minimum error value we select occurs at $r = 28.71$. Once the rescaling factors are obtained, a slab can be constructed using the procedure explained in equations (28)–(30) using $\mathbf{p} = 32.66(1, 1, 0)$, $\mathbf{p}_1 = 32.66(-1, 1, 0)$, and $\mathbf{p}_2 = 28.71(0, 0, 1)$. The resulting slab is shown in figure 8(b).

6. Conclusions

We have introduced a step-by-step procedure to cut a crystal slab from a crystal lattice which is typically needed for the study of both physical and chemical surface properties in computational chemistry simulations. Our method can be used to recover the PBC vectors for a given crystal slab and generate any crystal-based convex polyhedron such as Wulff structures. Most importantly, we made sure to provide all the equations needed for its implementation. We also introduced a technique to generate a crystal slab with orthogonal PBC vectors given a desired crystal face of interest. This technique is based on a simple parameter search with a clear stopping criterion. Here again we made sure to provide the steps and equations for its implementation.

All these methods can be easily implemented as a preprocessing step for any computational chemistry code. Moreover, these methods are inherently parallelizable and can readily take advantage of the plethora of heterogeneous computing platforms that are becoming more and more widespread. Hence, we expect that the resulting tools would be able to handle extremely large systems, the sole limitation being the memory availability to store the atomic coordinates on a given compute node.

A first version of an in-house code Los Alamos Crystal Cut (LCC) that implements these methods can be found at https://github.com/lanl/LCC. This implementation is based on a command-line tool which is ideal for batch-processing on High Performance Computing (HPC) clusters.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Acknowledgments

This work was supported by the Laboratory Directed Research and Development program of Los Alamos National Laboratory under Project Number 20220431ER. This research used resources provided by the Los Alamos National Laboratory Institutional Computing Program. E M acknowledges support of $\Delta = \min\left(\frac{1}{\max(n)}; \frac{1}{\max(n)}; \frac{1}{\max(n)}\right)$ so that $t = i\Delta$ where $i$ is an integer. This will ensure taking a ‘large enough’ step size.
from Clemson University startup funds. E M was supported in part by the National Science Foundation EPSCoR Program under NSF Award # OIA-1655740. Los Alamos National Laboratory is operated by Triad National Security, LLC, for the National Nuclear Security Administration of U.S. Department of Energy (Contract No. 89233218CNA000001). Additionally, we thank the CCS-7 group and Darwin cluster at Los Alamos National Laboratory for computational resources. Darwin is funded by the Computational Systems and Software Environments (CSSE) subprogram of LANL’s ASC Program (NNSA/DOE), LA-UR-22-29886.

Appendix A. Pseudo-code for finding perpendicular directions to \((hkl)\)

The following pseudo-code takes three integers corresponding to \(h, k,\) and \(l\) respectively and returns two orthogonal Miller indices \((h_1, k_1, l_1)\) and \((h_2, k_2, l_2)\) respectively. The function \texttt{permute()} is used to permute the indices in case \(h = 0\). The \texttt{permuteBack()} function gives back the correct order of the final indices. Finally, \texttt{minValNonZero()} takes the minimum non-zero value given the obtained indices.

```python
def GET_PERP_HKL(h, k, l):
    if (h = 0):
        \((h', k', l') = \text{permute}(h, k, l)\)
    else:
        \(k'_1 = 1\)
        \(l'_1 = 0\)
        \(h'_1 = -k'/h'\)
        \(h'_2 = \frac{\sqrt{4h'^2 - l'^2}}{h'} - l'/h'\)
        \(k'_2 = \frac{-\sqrt{4h'^2 - l'^2}}{h'}\)
        \(l'_2 = 1\)
    if (h = 0):
        \((h, k, l) = \text{permuteBack}(h', k', l')\)
        \((h_1, k_1, l_1) = \text{permuteBack}(h'_1, k'_1, l'_1)\)
        \((h_2, k_2, l_2) = \text{permuteBack}(h'_2, k'_2, l'_2)\)
    else:
        \((h, k, l) = (h', k', l')\)
        \((h_1, k_1, l_1) = (h'_1, k'_1, l'_1)\)
        \((h_2, k_2, l_2) = (h'_2, k'_2, l'_2)\)
        \(m = \text{minValNonZero}(h_1, k_1, l_1, h_2, k_2, l_2)\)
        \(h_1, k_1, l_1 = (h_1/m, k_1/m, l_1/m)\)
        \(h_2, k_2, l_2 = (h_2/m, k_2/m, l_2/m)\)
    return \((h_1, k_1, l_1), (h_2, k_2, l_2)\)
```

Appendix B. Wulff construction table of distances

Table B1 shows a list of the planes and distances used to construct the shapes shown in figure 6.

| \(\beta\)-HMX \((P2_1/n)\) | PFBA \((P2_1/c)\) | IDOX \((P1)\) |
|---------------------------|-----------------|-------------|
| Plane | \(d_{hkl}\) | Plane | \(d_{hkl}\) | Plane | \(d_{hkl}\) |
| \((002)\) | 3.59 | \((002)\) | 4.72 | \((001)\) | 8.73 |
| \((020)\) | 5.51 | \((020)\) | 3.10 | \((010)\) | 6.51 |
| \((200)\) | 3.19 | \((100)\) | 12.53 | \((100)\) | 4.58 |
| \((011)\) | 6.02 | \((011)\) | 5.18 | \((011)\) | 6.46 |
| \((101)\) | 6.02 | \((011)\) | 5.18 | \((111)\) | 6.05 |
| \((101)\) | 4.32 | \((202)\) | 3.49 | \((101)\) | 3.55 |
| \((110)\) | 5.39 | \((202)\) | 4.13 | \((101)\) | 4.86 |
| \((110)\) | 5.52 | \((220)\) | 2.78 | \((110)\) | 3.40 |
| \((222)\) | 2.01 | \((111)\) | 4.63 | \((111)\) | 2.80 |
| \((222)\) | 2.42 | \((111)\) | 4.96 | \((111)\) | 3.73 |
| \((222)\) | 2.01 | \((111)\) | 4.63 | \((111)\) | 3.57 |
| \((222)\) | 2.42 | \((111)\) | 4.96 | \((111)\) | 4.08 |

Appendix C. Plane periodicity

In this section, we provide a formal argument that the plane periodicity (that is, the distance between two adjacent planes) can be computed using the formula:

\[
T = d_{hkl} = \left\| \frac{2\pi}{h b_1 + k b_2 + l b_3} \right\|. \tag{C.1}
\]

Let \(\{\Pi_0, \ldots, \Pi_i, \ldots, \Pi_N, \ldots\}\) be the collection of parallel planes with Miller indices \((hkl)\). Within this set planes \(\Pi_i\) and \(\Pi_{i+1}\) are contiguous or neighboring planes. We find \(T\) such that, \(\text{min}(d(\Pi_i, \Pi_{i+1})) = T\), where \(\Pi_i\) and \(\Pi_{i+1}\) are two adjacent planes. Given the definition of Miller indices, without loss of generality we can say that both points \(\frac{i \Pi}{\Pi}\) and \((i + 1) \sqrt{\Pi}\) belong to \(\Pi_i\) and \(\Pi_{i+1}\) respectively, for otherwise if \(\Pi_i\) did not intersect \(a_i\), we could choose \(a_i\) or \(a_{i+1}\) in its place. In other words, if a plane \(\Pi_i\) cuts the \(a_i\) axis at \(\sqrt{\Pi}\), the ‘next plane’ will cut the \(a_i\) axis at \((i + 1) \sqrt{\Pi}\). Moreover, if \(r_1\) and \(r_2\) belong to \(\Pi_i\) and \(\Pi_{i+1}\), respectively, then:

\[
\left( r_1 - \frac{1}{h} a_i \right) \cdot N = 0, \tag{C.2}
\]
and

\[
\left( r_2 - (i + 1) \frac{1}{h} a_i \right) \cdot N = 0. \tag{C.3}
\]
The minimum distance between these two planes will be found by constructing the segment passing through the normal \( \mathbf{N} = h \mathbf{b}_1 + k \mathbf{b}_2 + l \mathbf{b}_3 \), which will imply that:

\[
(r_2 - r_1) \cdot \frac{\mathbf{N}}{||\mathbf{N}||} = ||r_2 - r_1|| = T. \tag{C.4}
\]

If we now subtract equation (C.2) from equation (C.3), we get:

\[
(r_2 - r_1) \cdot \mathbf{N} = \frac{a_1}{h} \cdot \mathbf{N}. \tag{C.5}
\]

If we now replace \((r_2 - r_1) \cdot \mathbf{N}\) of equation (C.4) by the right hand side (C.5), we get:

\[
\frac{a_1}{h} \cdot \frac{\mathbf{N}}{||\mathbf{N}||} = T, \tag{C.6}
\]

or

\[
\frac{2\pi}{||\mathbf{N}||} = T, \tag{C.7}
\]

where we have used the fact that \(a_1 \cdot b_1 = 2\pi\) and \(a_1 \cdot b_2 = a_1 \cdot b_3 = 0\). Note that over a period \(T\) there is a total of one plane that is found (half of \(\Pi_1\) and half of \(\Pi_{1+}\)). The ‘plane frequency’ \(f\) would then be \(1/T\) and the ‘angular plane frequency’ would be computed as \(2\pi/T\). The latter means that the norm of the normal vector \(\mathbf{N}\) to an (hkl) Miller plane gives us the ‘angular plane frequency’ in the \(\mathbf{N}\) direction.

**ORCID iD**

Himanshu Singh  [https://orcid.org/0000-0002-8437-686X](https://orcid.org/0000-0002-8437-686X)

**References**

[1] Schmickler W and Santos E 2010 *Interfacial Electrochemistry* (Berlin: Springer)

[2] Shetty M et al 2020 *ACS Catal.* 10 12666–95

[3] Basavarajappa P S, Patil S B, Ganganagappa N, Reddy K R, Raghu A V and Reddy C V 2020 *Int. J. Hydrog. Energy* 45 7764–78

[4] Liang Z, Shen R, Ng Y H, Zhang P, Xiang Q and Li X 2020 *J. Mater. Sci. Technol.* 56 89–121

[5] Häkkinen H 2012 *Nat. Chem.* 4 443–55

[6] Batzill M 2012 *Surf. Sci. Rep.* 67 83–115

[7] Anfuso C L, Xiao D, Ricks A M, Negre C F A, Batista V S and Lian T 2012 *J. Phys. Chem. C Nanomater. Interfaces* 116 24107–14

[8] Wang Q H, Kalantar-Zadeh K, Kis A, Coleman J N and Strano M S 2012 *Nat. Nanotechnol.* 7 699–712

[9] Boles M A, Ling D, Hyeon T and Talapin D V 2016 *Nat. Mater.* 15 153

[10] Robin A, Livache C, Ishihara S, Lacaze E, Dubertret B and Lhuillier E 2016 *ACS Appl. Mater. Interfaces* 8 27122–8

[11] Martínez B, Livache C, Goubet N, Jagtap A, Cruguel H, Ouerghi A, Lacaze E, Silly M G and Lhuillier E 2018 *J. Phys. Chem. C* 122 859–65

[12] Novoselov K S, Mishchenko A, Carvalho A and Neto A H C 2016 *Science* 353 1049–53

[13] VahidMohammadi A, Rosen J and Gogotsi Y 2021 *Science* 372 eabf1581

[14] Castner D G and Ratner B D 2002 *Surf. Sci.* 500 28–60

[15] Tirtell M, Kokkoli E and Biesalski M 2002 *Surf. Sci.* 500 61–83

[16] Docherty R, Clydesdale G, Roberts K J and Bennema P 1991 *J. Phys. D: Appl. Phys.* 24 89–99

[17] Setchell R E 1984 *Combust. Flame* 56 343–5

[18] Price D 1988 *J. Energy. Mater.* 6 215–54

[19] Dahal A and Batzill M 2014 *Nanoscale* 6 2548–62

[20] Negre C F A and Sanchez 2008 *J. Chem. Phys.* 129 034710

[21] Fuertes V, Negre C F A, Oviedo M B, Bonafé F P, Oliva F Y and Sánchez C G 2013 *J. Phys.: Condens. Matter* 25 115304

[22] Hammer B and Norskov J 2000 *Impact of Surface Science on Catalysis Theoretical Surface Science and Catalysis-Calculations and Concepts (Advances in Catalysis vol 45)* (New York: Academic) pp 71–129

[23] Sun W and Ceder G 2013 *Surf. Sci.* 617 53–59

[24] Perrot R, Lin Y, Zhakhovsky V, Vincetin N, Los J H, Maillet J B, Soulard L, White C T and Oleynik I I 2012 *AIP Conf. Proc.* 1426 1175–8

[25] Perrot R, Zhakhovsky V, Inogamov N A and Oleynik I I 2014 *J. Phys.: Conf. Ser.* 500 172008

[26] Kroonblawd M P, Mathew N, Jiang S and Sewell T D 2016 *Comput. Phys. Commun.* 207 232–42

[27] Palmer D C 2015 *Z. Kristallogr.* 230 559–72

[28] Hanwell M D, Curtis D E, Lomie D C, Vandermeersch T, Zurek E and Hutchison G R 2012 *J. Cheminform.* 4 17

[29] Stukowski A 2010 *Modelling Simul. Mater. Sci. Eng.* 18 015012

[30] Webmaster X XcrysDen—(X-Window) crystalline structures and densities (available at: [www.xcrysden.org/doc/intro.html](https://www.xcrysden.org/doc/intro.html)) (Accessed 19 January 2023)

[31] Building crystals and slabs—tutorials 2022.1 documentation (available at: [www.scm.com/doc/Tutorials/BuildingStructures/BuildingCrystalsAndSlabs.html](https://www.scm.com/doc/Tutorials/BuildingStructures/BuildingCrystalsAndSlabs.html)) (Accessed 19 January 2023)

[32] Creating a surface slab model using VESTA (available at: [https://ma.issp.u-tokyo.ac.jp/en/app-post/1788](https://ma.issp.u-tokyo.ac.jp/en/app-post/1788)) (Accessed 20 January 2023)

[33] Hirl P 2015 *Comput. Phys. Commun.* 197 212–9

[34] Ashcroft N W and Mermin N D 1976 *Solid State Physics* (New York: Holt, Rinehart and Winston)

[35] Wulf G 1901 *Z. Kristallogr.* 34 449–530

[36] Mutter A 2007 *The bond-valence deficiency model: a new application to describe mineral surface reactions* (available at: [https://miami.uni-muenster.de/Record/da726a97-360a-4d7e-86f0-050e010ba454/Export?style=BibTeX](https://miami.uni-muenster.de/Record/da726a97-360a-4d7e-86f0-050e010ba454/Export?style=BibTeX))

[37] Donnay J D H and Harker D 1937 *Am. Mineral.* 22 446–67

[38] Docherty R, Clydesdale G, Roberts K J and Bennema P 1991 *J. Phys. D: Appl. Phys.* 24 89

[39] Hahn T 2005 *International Tables of Crystallography (A - Space Group Symmetry)* 5th edn vol A (Dordrecht: Springer)

[40] Groom C R, Bruno I J, Lightfoot M P and Ward S C 2016 *Acta Cryst.* B72 171–9

[41] Yeager J D, Higginbotham Duque A L, Shorty M, Bowden P R and Stull J A 2018 *J. Energy. Mater.* 36 253–65

[42] Gallagher H G, Sherwood J N and Vrcelj R M 2017 *J. Cryst. Growth* 475 192–201

[43] Gallagher H G, Sherwood J N and Vrcelj R M 2014 *Chem. Cent. J.* 8 75