Computer Physics Communications 257 (2020) 107515

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

ARTEMIS: *Ab initio* restructuring tool enabling the modelling of interface structures☆,☆☆

Ned Thaddeus Taylor, Francis Huw Davies, Isiah Edward Mikel Rudkin, Conor Jason Price, Tsz Hin Chan, Steven Paul Hepplestone *

Department of Physics and Astronomy, University of Exeter, Stocker Road, Exeter, EX4 4QL, United Kingdom

ARTICLE INFO

Article history:
Received 3 March 2020
Received in revised form 6 July 2020
Accepted 10 July 2020
Available online 27 July 2020

Keywords:
Atomic modelling
Lattice matching
Interface
Crystals
Miller plane
Surfaces

ABSTRACT

A program, ARTEMIS, has been developed for the study of interface structures. This software allows for the generation of interfaces by identifying lattice matches between two parent crystal structures. To allow for further exploration of the energetic space of the interface, multiple surface terminations parallel to the Miller plane, interface alignments and intermixings are used to generate sets of potential interfaces for each lattice match. These interface structures can then be used in atomic simulations to determine the most energetically favourable interface. The software reported here can help to both drastically reduce the work of generating and exploring interfaces, and aid in understanding of how the interface structure influences the subsequent properties. Using several test cases, we demonstrate how ARTEMIS can both identify the location of an interface in existing structures, and also predict an optimum interface separation based upon the parents' atomic structures, which aims to accelerate and inform the study of interface science.

Program summary

Program Title: ARTEMIS
CPC Library link to program files: `https://dx.doi.org/10.17632/5bcrh67xty.1`
Developer's repository link: http://www.artemis-materials.co.uk/
Licensing provisions: CC BY NC 3.0
Programming language: Fortran 2003
Nature of problem: Construction and identification of the interface between any two crystals. Complicating factors include the choice of Miller planes, alignment of the two crystals and potential intermixing of them.
Solution method: This problem is tackled by generating sets of interface structures that allow the user to explore the energy space using atomic simulations in order to identify the most favourable interface to form between two such crystals.
Additional comments: The source code and working examples can be found in the compressed file obtainable from http://www.artemis-materials.co.uk/. The code has been tested and developed using the GNU 7.2.0 and the Intel 17.0.4 Fortran compilers on Unix/Linux operating systems.

1. Introduction

Two quotes are considered when discussing interfaces: (i) "Interface is the device" - H. Kroemer [1], who was emphasising how devices at their core are governed by the physics of the interface, and (ii) "God made the bulk, the surface was invented by the devil" - W. Pauli [2], who was outlining the complexity and disorder that surfaces and hence interfaces possess. Hence, to understand a device one needs to understand the interface.

Heterostructures and metal–semiconductor interfaces display several unique features, such as Andersons- and Schottky-like band alignment [3–9], thermal barrier effects [10–12], atomic reconstructions [13–16], and nanometre scale metamaterials [17,18]. The modelling of these features is generally based upon atomic scale approaches, which require an accurate interface structure, or a close approximation, such that atomic simulations can find the ground state. However, the complexity of the interface is vast, due to the issues of lattice matching, surface termination, intermixing, the large size of the unit cells, poor initial guesses and reaction kinetics, which can be further hindered by unintended human biases. The electronic properties are

☆ The review of this paper was arranged by Prof. D.P. Landau.

☆☆ This paper and its associated computer program are available via the Computer Physics Communication homepage on ScienceDirect (http://www.sciencedirect.com/science/journal/00104655).

* Corresponding author.
E-mail address: S.P.Hepplestone@exeter.ac.uk (S.P. Hepplestone).

https://doi.org/10.1016/j.cpc.2020.107515

sometimes considered using Anderson's rule, but atomic scale physics and measurements have shown repeatedly that this rule is invalid [19–21] and hence to accurately predict the band alignment requires the interface structure. Similarly, thermal boundary resistance is governed by the structure and make up of the interface [10,11]. Thus the need for an accurate atomic structure is great. In addition, interfaces can yield unique physical phenomena such as quantum wells, new material phases [22–25] and conductivity between insulators [26–28], all of which require accurate modelling. All these issues are compounded by the fact that experimental characterisation of an interface is exceptionally difficult. The best characterisations, such as Transmission Electron Microscopy [29,30], X-ray-diffraction [31,32] , and others [33–35], can only provide hints as to the structure, the range of intermixing, and which surface reconstruction formed the interface.

The last 40 years have seen significant interest in methods to define, identify and consider the properties of interfaces and, in particular, for examining and calculating the most favourable interface between two materials. Starting in 1984, Zur et al. [36] presented a method of lattice matching that lay the foundation for subsequent methods [37–40]. Raclariu et al. [37] implemented Zur's method for lattice matching and applied a nearest neighbour method to estimate the optimum position of the two materials with respect to each other. Mathew et al. [38] developed a series of scripts with capabilities to predict the structure of surface structures according to Wulff construction, and also match interfaces using Zur's lattice matching algorithm. Daniele and Jelver et al. [39,40] implemented the method by Zur and extend it by introduction the elastic tensors of the two individual crystals in order to determine the energy cost of performing a lattice match. Whilst these methodologies involve lattice matching, little focus is placed on the surface stoichiometry the materials at this interface and the potential fo diffusion of the two materials across the interface.

Another approach to interface prediction involves random structure searches [41–43]. Von Alfthan et al. [41] developed an approach which randomly swapped atoms at a grain boundary of Si. This approach was expanded by Chua et al. [42], who developed a method for exploring grain boundaries, which involves hopping atoms from one site to another based on a genetic algorithm, creating permutations of the interface. The methodology put forward by Schusteritsch et al. [43] involves generating an interface and randomly moving the atoms within a region around the interface; the generated structures are then modelled using first-principles methods in order to determine the most energetically favourable configuration. All of these approaches are promising, but currently limited to grain boundaries between two materials with matching stoichiometry, simply circumventing the need for lattice matching. The use of these methods allows for the modelling of the diffusion of materials across the interface, which cannot be captured by first-principles relaxation methods alone (due to the large energy barriers involved in relaxing such a system).

There has been little focus on joining these two approaches of lattice matching and random-structure generation, mainly due to the significant number of systems that one would need to explore before finding the true interface for a pair of materials. The software package presented here attempts to go beyond these previous methods by fully exploring the surface stoichiometry of slabs, developing a method to more accurately predict the optimum position of two materials with respect to each other, and developing a methodology for investigating diffusion alongside the lattice matching method.

Here, we present ARTEMIS, a software package designed to produce a set of interface structures between two bulk materials. From these interfaces, one can then use either empirical or *ab initio* software to refine the structure, rank the stability of the various potential interfaces by energy, and then calculate the desired properties. The workflow for producing the interface structures follows a logical progression, allowing the user to modify settings as they wish. ARTEMIS estimates the optimum interface separation and includes the capability to both shift and intermix interfaces. In addition, the software allows the user to submit existing interface structures and use the software for both post processing and further reconstructions/interface generation. These tools aid in a more thorough exploration of the interface and thus reduces human bias. The software is being continuously developed and the latest version can be obtained from http://www.artemis-materials.co.uk/.

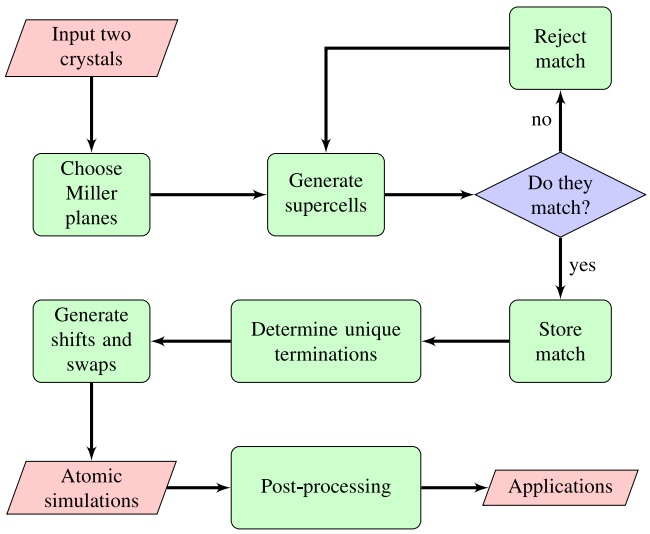

Fig. 1. An overview of the workflow process of ARTEMIS, outlining its key stages.

2. ARTEMIS: overview

The ARTEMIS software package generates interfaces as outlined in Fig. 1. It takes two parents crystals and then considers a set of Miller planes over which to search for matches. On each plane, it generates supercells for both materials and determines whether they match each other, as detailed in Section 2.1. When successful matches are found, unique terminations of those Miller planes are used to generate interfaces. The two slabs of materials are then shifted (displaced) with respect to each other, both parallel and perpendicular to the interface, and in the process, a set of shifted interface structures are generated. Output structure files are then written for each configuration for use in atomic simulation software packages. The software also has the option to take in pregenerated interface structures and perform further shifts or intermixings on them.

The ARTEMIS software package is written in Fortran, using a set of default Fortran functions and in-house developed functions and subroutines and is developed to read and output crystal structure files in Quantum Espresso, CASTEP (.cell) and VASP (POSCAR, .vasp) structure file formats. Additional formats can be achieved by using tools such as VESTA [44].

Built into the ARTEMIS code is a help function accessible via the use of flags. This help function details the possible tags for use in the code's input file. Also, ARTEMIS can be run using flags, though the recommended method of running is using the input file.

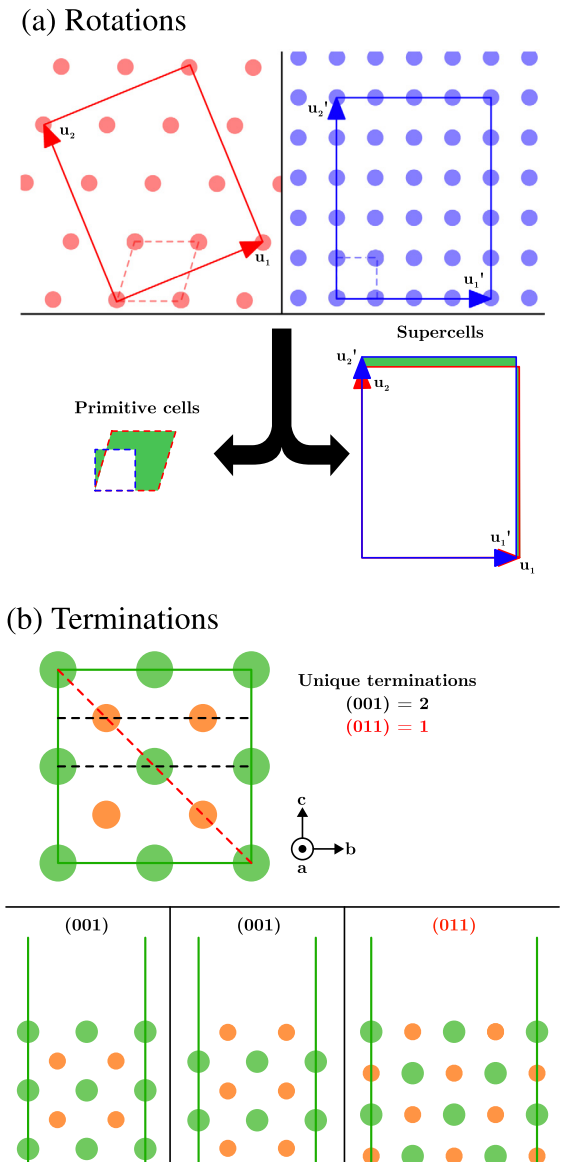

Fig. 2. Schematics outlining two main stages of lattice matching. (a) Rotations (supercells) of two surfaces can lead to better matching and, hence, less strain. The top two panels depict the lattice of two crystal surfaces, along with the primitive and a supercell for each lattice. The bottom panel displays the poor matching (highlighted in green) of the two primitive cells and the better matching of the chosen supercells. (b) Potential surface terminations of the crystal for the (0 0 1) and the (0 1 1). The top panel shows the different potential cleavage planes. The bottom left and bottom middle panels depict the two unique surface terminations for the (0 0 1) Miller plane. The bottom right panel depicts the unique surface terminations for the (0 1 1) plane. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

2.1. Lattice matching

In order to construct an interface between two crystals, one needs to consider how one would match the lattices of the two structures. The first objective is to create a set of interfaces between the two parent crystals that minimises the potential strain caused by mismatch between them. The second is to identify the interfaces with minimal supercell area so as to reduce the total number of atoms in each structure. This second objective allows for more feasible computational simulation of the structures using various atomic scale approaches.

Here, we outline the algorithms for lattice matching and how, within this routine, Miller planes are generated, unique surface terminations are identified and vector matching is performed. A schematic of this can be seen in Fig. 2a. The approach developed here is similar to that developed by Zur et al. [36] and Jelver et al. [40].

Fig. 3 represents the workflow of the lattice matching method. To begin with, we define our two crystals. These input lattice crystals are defined by lattice vectors $(\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3)$ and $(\mathbf{a}'_1, \mathbf{a}'_2, \mathbf{a}'_3)$. Vectors relating to the upper crystal will be defined by a prime ($'$) in this section, with all other vectors being related to the lower crystal. For both crystals, a set of Miller planes are generated to be considered for the lattice matching. Planes are constructed such that (h, k, l) obeys the condition of $-10 \leq h, k, l \geq 10$. Symmetry operations of the crystals are then applied to their respective Miller planes in order to reduce these sets to the unique Miller planes. The two Miller plane sets are then cycled over, determining the lattice vectors for each plane in turn. The shortest surface lattice vectors of each plane, $\mathbf{b}_1$, $\mathbf{b}_2$ ($\mathbf{b}'_1$, $\mathbf{b}'_2$), are obtained using the Lenstra–Lenstra–Lovász lattice basis reduction algorithm [45]. For each Miller plane, integer quantities of the two surface vectors are then combined to generate a list of lattice vectors that span this plane, *i.e.* $\mathbf{u} = i\mathbf{b}_1 + j\mathbf{b}_2$. These lattice vectors are then generated with the condition $|\mathbf{u}| \leq$ MAXLEN (MAXLEN, default = 20 Å). The list of lattice vectors obtained for the two parent crystals are then compared to find matching pairs of $(\mathbf{u}_1, \mathbf{u}'_1)$ and $(\mathbf{u}_2, \mathbf{u}'_2)$. These pairs are obtained when the magnitude of $\mathbf{u}_1$ ($\mathbf{u}_2$) matches the magnitude of $\mathbf{u}'_1$ ($\mathbf{u}'_2$) to within the user-defined tolerance (TOL_VEC, default = 5%). The list of lattice vector pairs defined above is then cycled through to find potential lattice matches. By joining two lattice vector pairs, it can be determined whether the angle between $\mathbf{u}_1$ and $\mathbf{u}_2$ matches that of $\mathbf{u}'_1$ and $\mathbf{u}'_2$ to within a tolerance defined by TOL_ANG (TOL_ANG, default = 1°). Pairs matching this criteria are then stored as a suitable lattice match. The set is then ordered by their vector and angular mismatch. A weighting is applied to favour matches of a smaller area, as such matches are more likely to be modellable. The user can define the number of most favourable lattice matches to be generated (NMATCH, default = 5).

2.2. Surface terminations

A method is implemented in ARTEMIS to distinguish the atomic layers within each material (with respect to the surface plane), thus allowing for unique terminations to be defined. However, in doing so, non-physical surfaces can be generated and care must be taken, especially for 2D materials.

In order to define unique surface terminations, we first need to specify what constitutes a *layer*. We define distinct atomic layers within a crystal as regions parallel to the chosen Miller direction that are separated by vacuum gaps of greater than LAYER_SEP (LAYER_SEP, default = 1 Å). This definition creates atomic layers appropriate to the chosen Miller direction, with the caveat that high atomic density regions remain grouped. Cleaving through such high-density regions is forbidden as this would be expected to result in a high cleavage energy (akin to cutting through a large number of bonds). Using this technique alone to create layers can result in two different layers being defined which are symmetrically equivalent. To determine if a layer is unique, we consider whether there exists a symmetry operation that matches the layer to another layer within the set (excluding mirror symmetries in the Miller direction); if a symmetry operation matches two layers in such a way, then one of these is discarded. From this set, the top atom for each of these layers is then defined as a unique surface termination. This process is performed for both crystals for each Miller plane being considered. The result is a set

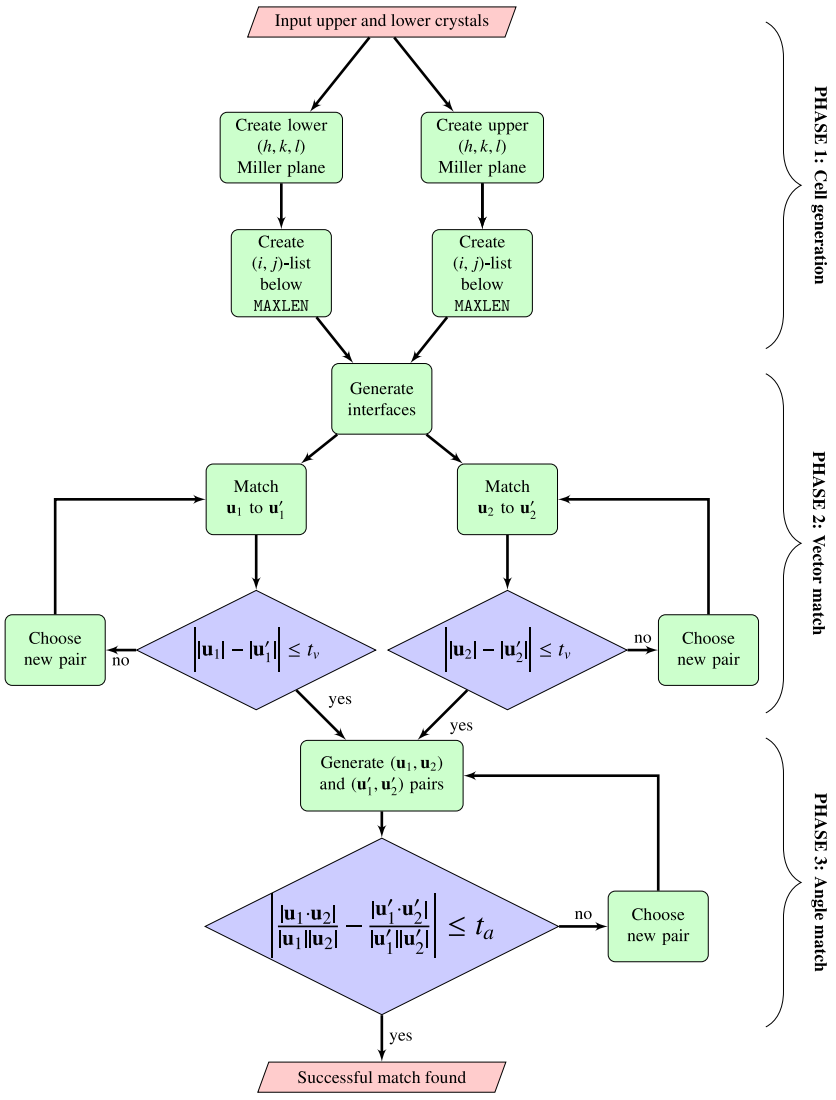

Fig. 3. Workflow of the lattice matching method implemented in ARTEMIS. The method is split into three phases labelled cell generation, vector match and angle match. t_v and t_a correspond to the `TOL_VEC` and `TOL_ANG` input file tags, respectively.

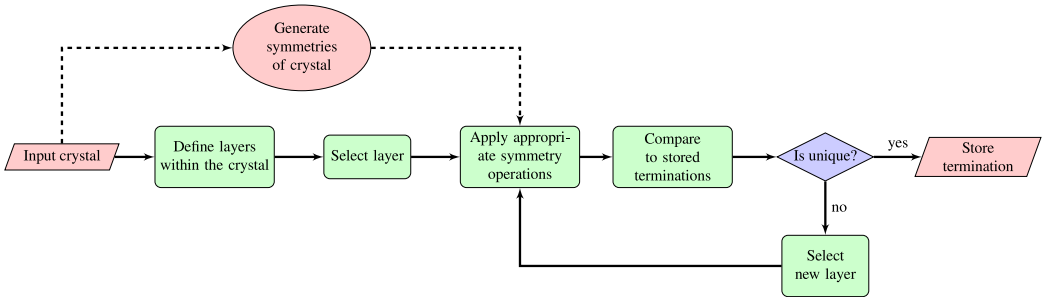

Fig. 4. Surface identification workflow diagram. The process by which ARTEMIS identifies unique surface terminations.

of unique surface terminations for each crystal. Fig. 2b depicts potential terminations of a crystal structure based on the choice of Miller plane and the workflow can be seen in Fig. 4.

The ideal end goal of the surface termination finding process is to generate two symmetrically equivalent interfaces. This is intended to prevent the creation of spurious electric fields, and model a single unique interface per simulation. However, ARTEMIS does allow the user to create asymmetric interfaces if they wish. The initial thickness of each slab is defined by the number of symmetrically equivalent layers, perpendicular to the Miller plane, within the parent crystal. The user can define the number of equivalent layers to be contained within the lower (upper) slab using the parameter `LW_SLAB_THICKNESS` (`UP_SLAB_THICKNESS`).

Two notable exceptions to the surface termination process are allowed within ARTEMIS. Firstly, to account for layered materials,

such as MoS_2, we include an input parameter `LW_LAYERED`. If `LW_LAYERED` (`UP_LAYERED`) is set to true, then only stoichiometric surface terminations are considered for the lower (upper) material. This prevents one from cleaving the constituent atoms within each layer of a van der Waals structure, such as in transition metal dichalcogenides [46]. And second, when a material displays neither S_n nor C_{nh} symmetries [47] ($n = 1, \ldots, 6$) in the Miller plane, such as in the case of fresnoite [24], ARTEMIS will create two unique interfaces in the system and, therefore, users have the option to change the surface termination of each slab. This is done using the `LW_SURFACE` (`UP_SURFACE`) tags. These same parameters can be also be used for symmetric structures if the user wishes.

2.3. Interface identification

Both to examine existing structures that have been made by other users, and to allow reruns on existing interface structures, ARTEMIS includes a subroutine capable of reading a structure and identifying the interfaces within it. The code has the capability to perform interface manipulations (as outlined in Section 2.4) on pregenerated interface structures. To begin with, we define the direction perpendicular to the interface as the 'interface direction'. To locate the interfaces in the pregenerated structures, ARTEMIS identifies the interface direction using a process we name DON and the location of the interfaces along this axis are identified using CAD. These two processes are discussed below.

2.3.1. DON — density of neighbours

In order to determine the interface direction, correlation functions are employed to determine the axis of greatest dissimilarity. To do so, we start by generating a species-dependent density of neighbours (sDON). The DON is an angle-independent description of the distance of atoms akin to the nearest neighbour profile used to characterise amorphous materials [48] and Fig. 5a gives a visual description of how it works. For each atom in the structure, a DON is generated in order to describe their local environments. The sDON and DON are given

$$\mathrm{sDON}_{s,i}(p, \Delta r) = \sum_{j}^{n_{\mathrm{atom},p}} \exp\left(-\frac{\left(\Delta r - |\mathbf{r}_{s,i} - \mathbf{r}_{p,j}|\right)^2}{2\sigma^2}\right), \tag{1}$$

$$\mathrm{DON}_{s,i}(\Delta r) = \sum_{p}^{n_{\mathrm{spec}}} \mathrm{sDON}_{s,i}(p, \Delta r), \tag{2}$$

where $\mathbf{r}_{s,i}$ and Δr are the location of and the distance from atom i in species s, respectively. n_{spec} is the total number of species in the system, $n_{\mathrm{atom},p}$ defines the number of atoms associated with species p, and $\sigma = 0.05$ Å. The sDON of each atom is then compared, with all the other atoms in order to determine the *similarity* of that one atom to the rest of the structure (with a heavier weighting placed on the atoms nearby, offering a description for the similarity of the local environment). We define this similarity, *sim* as

$$\mathrm{sim}_{s,i} = \sum_{x}^{x_{\mathrm{cutoff}}} \sum_{j}^{n_{\mathrm{atom},s}} \mathrm{mDON}_{s,i,j}(x) \exp\left(-\frac{|\mathbf{r}_{s,i} - \mathbf{r}_{s,j}|}{r_0}\right), \tag{3}$$

where $x_{\mathrm{cutoff}} = 4$ Å, $r_0 = 1$ Å and

$$\mathrm{mDON}_{s,i,j}(\Delta r) = \sum_{p}^{n_{\mathrm{spec}}} \min\Big(\mathrm{sDON}_{s,i}(p, \Delta r), \mathrm{sDON}_{s,j}(p, \Delta r)\Big). \tag{4}$$

Then we compare the range of these similarity values associated with each axis and define this range as the *dissimilarity coefficient*,

$$\mathrm{dissim}_{\alpha} = \max\Big(\big|\mathrm{sim}_{s,i} - \mathrm{sim}_{s,j}\big| f\left(\mathbf{r}_{s,i}(\alpha) - \mathbf{r}_{s,j}(\alpha)\right)\Big)_{s,i,j}, \tag{5}$$

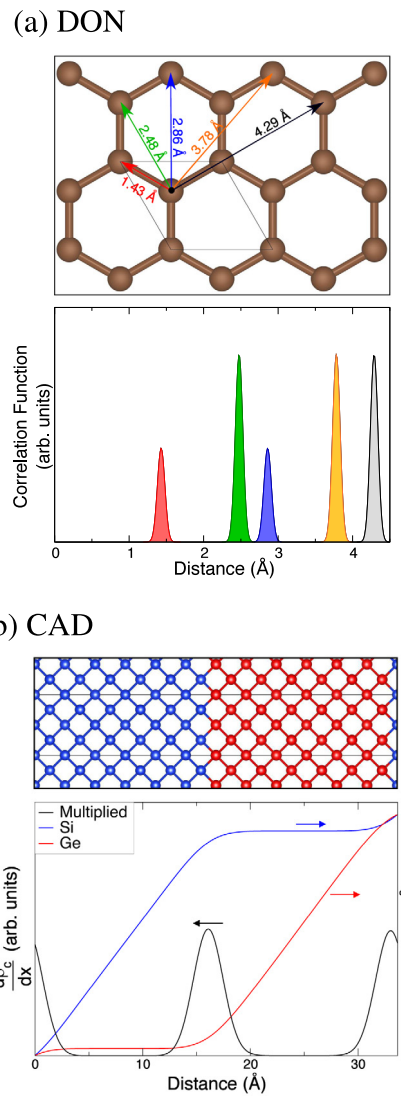

Fig. 5. The two methods implemented in ARTEMIS for interface identification. (a) Density of neighbours (DON), with a graphene monolayer used as an example. The top panel depicts the five shortest unique bond lengths of graphene. The bottom panel displays the radial distribution of bond lengths as Gaussian peaks, where the height of the peak relates to the number of that bond length present. (b) Cumulative atomic density (CAD), with an Si|Ge interface used as an example. The top panel shows the structure of the chosen interface. The bottom panel graph plots the species-dependent cumulative atomic density (ρ_c) and its derivative ($d\rho_c^m/dx$). The arrows on the graph indicate which data relates to which axis.

with

$$f(x) = \begin{cases} 1, & |x| \leq 1 \text{ Å} \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

where $\alpha = 1, 2, 3$ refers to the $\mathbf{a}_1''$, $\mathbf{a}_2''$, $\mathbf{a}_3''$ axes, respectively. The dissimilarity is associated with planes normal to a given axis. For each axis α, the function $f(x)$ is used to constrain similarity value comparisons to atoms within 1 Å along the α axis of each other whilst disregarding the atoms' other axes coordinates. As such, the *dissimilarity coefficient* for an axis is defined by the plane with the largest dissimilarity. Hence, for all axes other than the interfacial axis, this check will result in a large dissimilarity and, therefore, the axis with the smallest *dissimilarity coefficient* determines the interface direction, defined as β. Having determined the interfacial axis, the locations along this axis relating to the two interfaces need to be identified.

2.3.2. CAD — cumulative atomic density

After having defined the interface direction (β), we now need to determine the interface location along that axis. The use of this method can be seen in Fig. 5b. Firstly, the species-dependent cumulative atomic density profile (CAD) for the interfacial axis is calculated from

$$\mathrm{CAD}_{\beta}(s, \Delta r_{\beta}) = \sum_{x=0}^{\Delta r_{\beta}} \sum_{i}^{n_{\mathrm{atom},s}} \exp\left(-\frac{\left(x - r_{\beta,s,i}\right)^2}{2\sigma^2}\right), \tag{7}$$

where $\sigma = 2$ Å, $r_{\beta,s,a}$ is the βth element of the position vector of $\mathrm{atom}_{s,a}$ and Δr_{β} defines the position along the βth axis. The derivative at each point is then determined for each species. Finally, all of the species-dependent cumulative atomic density derivative profiles are multiplied together and the two greatest maxima of this function are defined as the two interfaces of the system (as the structures are periodic, two interfaces are present in a cell).

$$\mathrm{multiCADD}_{\beta}(\Delta r_{\beta}) = \left| \prod_{s}^{n_{\mathrm{spec}}} \frac{d}{dx} \mathrm{CAD}_{\beta}(s, \Delta r_{\beta}) \right|. \tag{8}$$

The CAD process identifies the two interfaces within the cell as the two locations with greatest $\mathrm{multiCADD}_{\beta}(\Delta r_{\beta})$. The location of these interfaces along the interface direction are stored as $x_{\mathrm{intf},1}$ and $x_{\mathrm{intf},2}$. This method works for nearly all interfaces with the exception of planar defects where the material effectively has a stacking fault.

2.4. Interface manipulation

Matching two crystals' lattice vectors is the first stage of making an interface. However, as discussed previously, many other factors need to be considered. Chief among these are sources of stress in the structure. These can originate from the misalignment of the two materials, or from broken bonds or a lack of intermixing across the interface. In this section, we outline the various methods available within ARTEMIS to generate structures with different interface alignments (shifts) and to include atomic diffusion (in the form of atomic swaps). These structures are generated so that the user can examine the energy space dependence on these factors and assess the various interfaces and their favourability.

2.4.1. Shifting

We define the shifting of material A with respect to material B as the 'interfacial alignment'. Interface alignment is the first step in reducing stress. When determining shifts, it is important to consider the bonds that have been cleaved through in order form a surface from a crystal. Here, these *cut* (broken) bonds are termed as missing bonds. Having matched lattice vectors, one needs next to consider the correct alignment of the basis of the two separate structures in order to best accommodate for these missing bonds at the interface. A depiction of interfacial alignment can be seen in Fig. 6a.

A set of methods are available in ARTEMIS to manipulate *interface alignment* via generating and performing sets of shifts, $\boldsymbol{\delta}$. A shift is defined as a displacement of one surface with respect to the other. These methods are selected using the `ISHIFT` tag. `ISHIFT` $= 0$ allows for user-defined shift values, 1 uses random shifts, 2 matches the average interfacial bond to the two materials' bulk bonds and 3 and 4 attempt to predict a set of descriptive bonds. In particular, `ISHIFT` $= 3$, 4 represents ARTEMIS at its most comprehensive, generating a set of interfaces with different shift values. The number of shifted structures generated per lattice match is determined by `NSHIFT` (`NSHIFT`, default $= 5$).

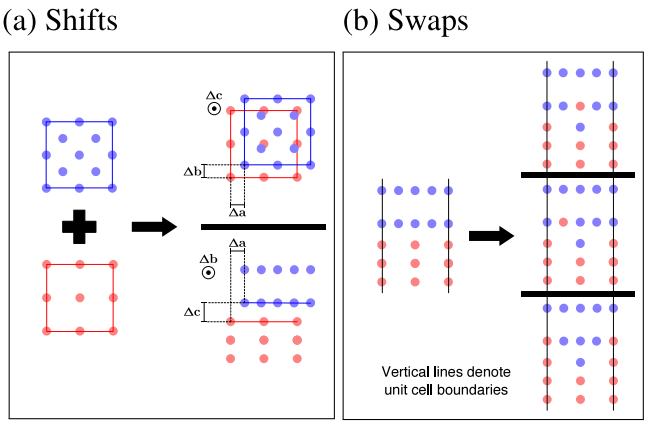

Fig. 6. Schematic outlining shifting (alignment) and swapping (intermixing) used to relieve interface strain. (a) Shifts: The left panel shows two crystal surfaces. The top right panel depicts the shifts in the interface plane, whilst the bottom right panel highlights shifts parallel to the interface direction. (b) Swaps: The left panel depicts a non-intermixed interface. The three unique swap structures are shown in the right panel as top, middle and bottom, with the structures being separated by the bold black bars. The vertical black lines denote the boundaries of the unit cell; as such, any atoms lying on a black line are repeated on the opposite black line.

`ISHIFT` $= 0$ allows the user to specify the tag `SHIFT` $= a\ b\ c$, where a, b and c are the manual shifts defined by the user and each can be any real number.

`ISHIFT` $= 1$ creates `NSHIFT` number of random shifts of the interface structure.

`ISHIFT` $= 2$ ensures that the average minimum separation between atoms either side of the interface best matches to the mean of the two parent crystals' averaged bond lengths.

`ISHIFT` $= 3$ begins using the `ISHIFT` $= 2$ methodology and then produces a set of `NSHIFT` shifts parallel to the interface plane, whilst maintaining its interfacial separation (the spacing parallel to the interface direction). By fixing the interface separation to that determined by `ISHIFT` $= 2$, a new set of shifts (parallel to the interface) are generated that allow for the average interface bond to most and least match to the average bulk bonds (best, worst, second best, second worst, etc.). These sets of shifts are intended to be descriptive of the energetic space of potential interface shifts, which allows the user to determine how drastically the energy can change depending on the alignment for chosen certain system.

`ISHIFT` $= 4$ is a method of shifting which attempts to match the bond lengths across the interface to the missing bulk bond lengths of the surface atoms. By determining the shortest missing bond of the surface atoms of each crystal (and the number of those bonds missing for each of them), the opposite crystal's structure can be used to supplement these absent bonds. This is similar to the broken-bond rule for metal surfaces, which assumes that the cleaving energy is directly related to the number of broken bonds for the metal's surface. Thus our method here assumes that the most energetically favourable alignment of two surfaces is to compensate for the most number of missing bonds [49,50].

To determine the missing bonds for `ISHIFT` $= 4$, we generate for each individual atom within each component slab a DON. We then subtract the individual DONs from their Wyckoff bulk (i.e. ideal) counterparts. This produces a nearest neighbour profile which highlights the missing bonds of each atom. These missing bonds are characterised by both their bond length and the number of such missing bonds. We then store the value of the shortest missing bond for each atom for later reference, where the

shortest bond of an atom must be smaller than MBOND_MAXLEN (MBOND_MAXLEN, default = 4 Å).

Next, the two slabs are placed on top of each other such that the top of the lower slab's surface and the bottom of the upper slab's surface occur at the same point (i.e. zero separation). No initial shift is applied either parallel or perpendicular to the interface plane. In order to determine a set of suitable shifts, the code evaluates a bond ideality factor, $D_{\mathbf{s}}$, for each possible shift ($\boldsymbol{\delta}$) and generates NSHIFT number of structures using the shifts with the smallest bond ideality factors. In the case of a perfectly matching interface, the bond ideality factor has a value of zero. The value of $D_{\mathbf{s}}$ is given as

$$D_{\delta} = d_{\delta,lu} + d_{\delta,ul} \tag{9}$$

where

$$d_{\delta,lu} = \sum_{m}^{l} \left[\left| N_m - \sum_{n}^{u} f(|\mathbf{r}_m + \boldsymbol{\delta} - \mathbf{r}'_n| - b_m) \right| + \sum_{n}^{u} g\left(\frac{|\mathbf{r}_m + \boldsymbol{\delta} - \mathbf{r}'_n| - b_m}{b_m} \right) \right]. \tag{10}$$

$d_{\delta,lu}$ quantifies how well slab u compensates for the missing bonds in the surface of slab l, which is generally distinct from $d_{\delta,ul}$. Here, l and u are the sets of missing bonds for atoms from the lower and upper slabs, respectively. N_m is the total number of missing bonds for atom m, $\mathbf{r}_m$ is the location of atom m and $\mathbf{r}'_n$ is the location of atom n. b_m is the length of the missing bond for atom m. We also include a pair of weighting functions,

$$f(x) = \frac{1 - \tanh(9(|x| - 1/4))}{2} \tag{11}$$

and

$$g(x) = \begin{cases} \frac{8}{3}|\tan(\pi x/2)|, & x < 0 \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

where the $f(x)$ function weights the set of shifts towards ones that compensate for the surface atoms' missing bonds, whilst $g(x)$ helps to prevent shifts that place atoms too close to one another. The specific forms of the two functions, $f(x)$ and $g(x)$, are used in order to satisfy the three following conditions. Firstly, $f(x)$ and $g(x)$ should be at their maximum and minimum, respectively, when a bond across the interface perfectly matches the missing bond length. Secondly, $f(x)$ should decrease with increasing disagreement between the interface bond and missing bond lengths. Finally, $g(x)$ should go to infinity when an interface bond goes to zero.

To further validate this approach, ISHIFT = 4 can be used between two identical bulk crystals. This methodology identifies appropriate shifts to reproduce a crystal's bulk bonds (to within 0.1 Å) across the interface and has been tested on numerous bulks containing either single or multiple species.

2.4.2. Intermixing

In order to enable exploration of diffusion across the interface, sets of structures are generated in which atoms from either material are intermixed (swapped), as demonstrated in Fig. 6b. Intermixing is introduced to serve three main purposes. Firstly, we previously observed [14] that the intermixing both reduces strain and produces interface structures more favourable than the initial abrupt boundary model. The second is that by introducing disorder it allows for interfaces involving more elaborate reconstructions to be explored, such as the GaAs $\beta(2 \times 4)$ surface [51,52]. Finally, such intermixing allows one to model grain boundaries, such as those considered by von Alfthan [41] and Schusteritsch [43].

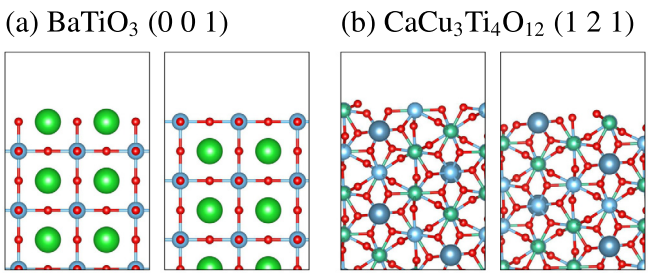

Fig. 7. The two unique surface planes as identified by ARTEMIS for (a) the (0 0 1) Miller plane of $BaTiO_3$ and (b) the (1 2 1) Miller plane of $CaCu_3Ti_4O_{12}$.

The concentration of intermixing is governed by the SWAP_DENSITY tag (units of Å^2) and the number of uniquely intermixed structures generated per shift is determined by NSWAP. Each set of swaps in a structure is generated randomly, excluding symmetrically equivalent swaps. New swaps are performed until the SWAP_DENSITY has been reached. By default, intermixing is turned off (ISWAP = 0). To turn on random intermixing, set ISWAP = 1. To use the physics-based swapping procedure, set ISWAP = 2.

ISWAP = 1 is a purely random swapping routine, based upon the user specifying a depth (SWAP_DEPTH, default = 3 Å), and the swapping density (SWAP_DENSITY, default = 0.05 Å^2). ARTEMIS then randomly swaps atoms either side of the interface to create a more graded interface.

ISWAP = 2 is based upon the observation [14] that the amount of intermixing exponentially decays with distance. As such, we use the following function to weight the chance of swapping

$$s(x) = \exp\left(-\frac{x - x_{\text{intf}}}{2\sigma^2_{\text{swap}}} \right) \tag{13}$$

where x_{intf} is the location of the interface and σ_{swap} defines the decay of the probability of atoms further from the interface being swapped (SWAP_SIGMA, default = 0.05 Å). This function mimics the behaviour seen in Ref. [14] by selecting atoms to swap with an exponential weighting towards selecting atoms closer to the interface.

For a system where the two interfaces are symmetrically equivalent, any swaps performed across one interface are then mirrored to the opposite interface. In doing so, each generated structure still has two interfaces in the unit cell, with those two interfaces still being symmetrically equivalent. In doing so, this will prevent the introduction of spurious electric fields into the system.

3. Test cases

3.1. Surface terminations

Using the LSURF_INFO tag, users can get the surface terminations of the lower (upper) parent crystal along the Miller plane specified by the LW_MILLER (UP_MILLER) tag. The surface is generated using the 1×1 unit cell reconstruction with the shortest lattice vectors on that plane. The number of symmetrically equivalent layers in the lower (upper) slab is specified using the LW_SLAB_THICKNESS (UP_SLAB_THICKNESS) tag.

The surface termination identification method used in ARTEMIS has been tested on a large set of structures. In Fig. 7, the unique surfaces identified by ARTEMIS are shown for $BaTiO_3$ and $CaCu_3Ti_4O_{12}$ for the (0 0 1) and the (1 2 1) Miller planes, respectively. For all of the test cases, ARTEMIS generates the surfaces using the smallest lattice vectors parallel to the surface.

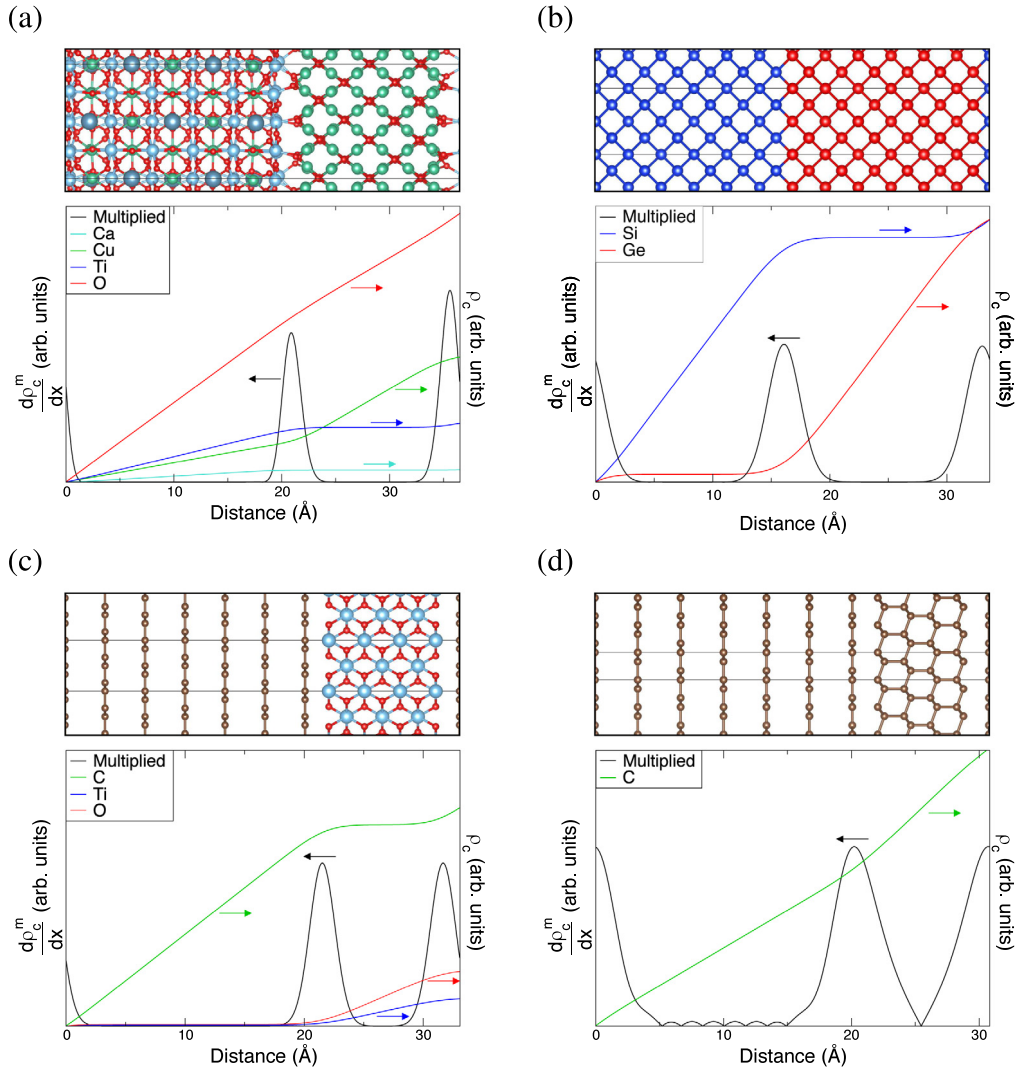

Fig. 8. Interface identification performed on four structures. Cumulative atomic density derivative plots shown for (a) $CaCu_3Ti_4O_{12}$|CuO, (b) Si|Ge, (c) graphite|TiO_2 and (d) graphite|diamond interfaces. The arrows on the graphs indicate which data relates to which axis. These interfaces have been used to display four different qualities interfaces can exhibit. For all plots, the righthand axis denotes the cumulative density for each species in the system, whilst the lefthand axis denotes the 1st-order derivative of the species-multiplied cumulative atomic density (multiCADD).

For the (0 0 1) Miller plane, the $BaTiO_3$ crystal is shown to have two unique surfaces, the AO-rich and the BO_2-rich surfaces (where a perovskite's chemical formula is given as ABO_3). These are the two most common surfaces of perovskites, with the BO_2 being the most commonly studied [53–58].

In the case of examples similar to the $CaCu_3Ti_4O_{12}$ (1 2 1) Miller plane, more care needs to be taken. As can be seen in Fig. 7b, surface terminations are less clearly identifiable and, as such, defining surfaces is more difficult. Still, two unique surface terminations are identified, with these being chosen due to possessing the largest interplanar spacing.

Surface terminations have been generated for the two-atom primitive cell of silicon using the (1 0 1) Miller plane. It is found to properly identify the well-known surface (0 0 1) termination with the [1 1 0] and [1 $\bar{1}$ 0] in-plane lattice vectors [59–61].

3.2. Interface identification

Interface identification is tested on a set of sample structures as shown in Fig. 8. As can be seen, ARTEMIS identifies the interface in all cases. These four interface structures are used to exhibit different structural features an interface may have (only the $CaCu_3Ti_4O_{12}$|CuO structure has been relaxed, all others are generated examples for testing). The Si|Ge system is used as an example of structurally identical regions that differ only by species. The graphite|TiO_2 system shows an interface between 2D layered and a 3D bulk crystals. The $CaCu_3Ti_4O_{12}$|CuO system demonstrates a chemically diverse and disordered interface. Finally, the graphite|diamond interface demonstrates a transition between two phases of the same material. In each structural scenario, the interface is successfully identified.

3.3. Shifting

Here, the `ISHIFT` = 4 interface alignment setting is tested and presented on three 2D layered|3D bulk (Fig. 9a) and two 3D bulk|3D bulk (Fig. 9b) interface structures. To test this method, we generate a set of 20 interface separations within the range shown on the separation axes for each structure and calculate the energy for each. The interface separation predicted by ARTEMIS is then indicated on these lines by a red dot. We find that, in all cases, the default interface separation prediction is close to the most energetically favourable separation (to within 0.2 Å in these cases shown). This initial estimate of the separation of the

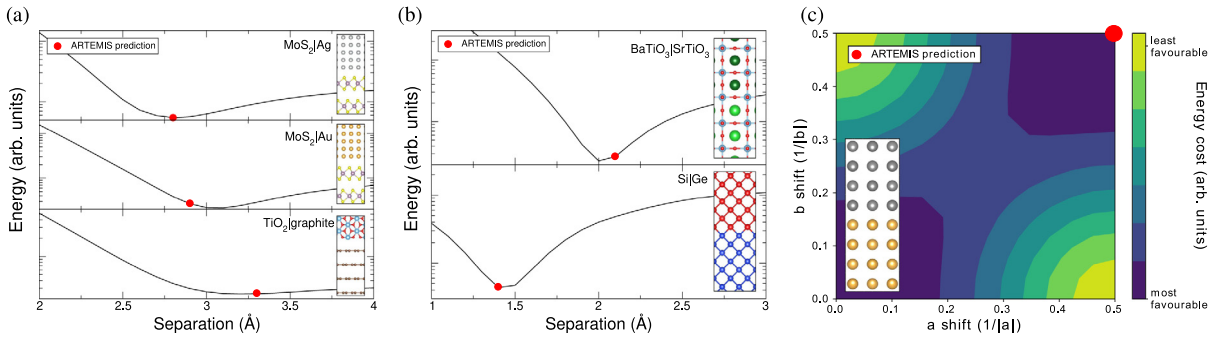

Fig. 9. The relationship between energy and the alignment of the two parent crystals. (a) & (b) show the relation between energy and the interface separation for 2D|3D and 3D|3D interfaces, respectively. The optimum separation as estimated by ARTEMIS for each interface structure is denoted by the red dots. The energies are obtained using the VASP density functional theory software package. (c) displays the energy cost of different shifts parallel to the interface plane for an Au|Ag interfacial structure. The optimum shift as estimated by ARTEMIS for each interface structure is denoted by the red dot. The inset displays the Au|Ag interface structure used to display the effect of shifting on the interface energy.

two slabs means that the structure can be relaxed to its optimum distance using external first-principles software packages. This suggests the method of `ISHIFT` $= 4$ discussed in Section 2.4.1 is an appropriate method for interface separation estimation.

For shifts parallel to the interface, a test case of an Au|Ag interface has been chosen. The chosen Miller planes are (0 0 1) for both parent crystals. The energetic space of the in-plane shifts is shown in Fig. 9c, with the shift predicted by ARTEMIS being labelled with a red dot. The shift determined by ARTEMIS is found to be the most favourable alignment for this metal–metal interface. Currently, this method has only been found to automatically identify the most favourable parallel shift for interfaces between simple, single-species crystals. For more complicated structures, the number of required structures (as generated by `ISHIFT` $=$ 4) increases and the user needs to consider each case. However, it should be emphasised that interface separation predicted by ARTEMIS is generally found within the first iteration.

All energies reported in this subsection are obtained using density functional theory as implemented in the Vienna Ab initio Simulation Package (VASP). The energy for each shifted structure has been obtained by performing self-consistent field calculations on the generated, unrelaxed, structures. The calculations have been performed using the PBE GGA functional with PAW pseudopotentials and an energy convergence of 10^{-7} eV. k-point grids of $3 \times 3 \times 2$ have been used for each structure.

3.4. Multiple interfaces

As mentioned previously, ARTEMIS can be used to reduce the difficulty of exploring the potential interfaces between crystals. There can exist a wide range in the number of potential lattice matches between any two crystals, which depends on their individual lattice shapes and sizes. The ideal goal of ARTEMIS is to make the process of determining potential lattice matches easier, along with identifying the potential surface terminations for each crystal and the relative alignments of the two (outlined in Sections 3.1 and 3.3, respectively).

For an insight into the capability of ARTEMIS to aid in determining the most energetically favourable interface structure from parent crystals, a recent article by the authors studies the interfaces for the systems of $CaCu_3Ti_4O_{12}$|CuO and $CaCu_3Ti_4O_{12}$|Cu_2O [28]. For the study, a total of 20 interface structures, including various shifts have been explored. The two most energetically favourable interface structures are the results reported in the article, which were also the most representative of the overall set.

4. Inputs and outputs

The code requires either one or two material structure files (depending on whether the user is supplying a preconstructed interface or two bulk materials) and one ARTEMIS input file.

If the user has supplied two material structure files for the purposes of generating an interface between them, then Sections 2.1 and 2.4 are used. If a preconstructed interface structure file is supplied, then Sections 2.3 and 2.4 are used.

The ARTEMIS input file has a set of cards and tags that can be explored and understood by using the built-in help function

```
./bin/artemis --help all
```

This will print all potential tags to the terminal, with a brief description of each. For further information on tag `[TAGNAME]`, use the command

```
./bin/artemis --search [TAGNAME]
```

With this, users can find the full set of tags available in later versions of ARTEMIS released after this manuscript. A set of important tags have been listed in Table 1.

5. Porting the code

The ARTEMIS code requires no external libraries. Hence, no additional subroutines other than those provided are required. This code has been compiled using both gcc 7.2.0 and the Intel 17.0.4 Fortran compilers, with no difference having been observed. Earlier compilers may encounter problems.

6. Conclusions

We have developed a software package, ARTEMIS, for the generation of potential interfaces between any two materials; here the parent materials can be single-species or multi-species, layered or bulk. The generated interface structure-files can be output in a set of different formats that allow ease of use for DFT calculations using either QuantumEspresso, CASTEP or VASP, with an aim to output in additional file formats for large-scale empirical calculations in the future, though. Through the use of the ARTEMIS software, users can more easily automate the task of interface generation and identification; for example, sets of interfacial alignments and intermixings can be easily generated, allowing for more thorough studies of potential interfaces between two parent crystals and their energy landscapes. With the lattice matching method able to scan over large sets of Miller planes, users can quickly identify low-strain matches which would otherwise take a lot of time to find. The test cases

Table 1
A selection of important ARTEMIS input file tags.

Tag	Default
Settings	General settings
STRUC1_FILE - *Parent crystal 1 filename*	POSCAR1
STRUC2_FILE - *Parent crystal 2 filename*	POSCAR2
MASTER_DIR - *Output directory*	DINTERFACES
Interfaces	Interface-specific settings
NMATCH - *Number of lattice matches to generate*	5
NINTF - *Total number of interface structures*	100
LGEN_INTERFACES - *Generate interface files*	TRUE
LSURF_GEN - *Generate surface termination files*	FALSE
TOL_VEC - *Lattice match vector tolerance (%)*	5
TOL_ANG - *Lattice match angle tolerance (°)*	1
LW_MILLER - *Lower crystal Miller plane*	(empty)
UP_MILLER - *Upper crystal Miller plane*	(empty)
LW_SLAB_THICKNESS - *Lower crystal slab thickness*	3
UP_SLAB_THICKNESS - *Upper crystal slab thickness*	3
ISHIFT - *Shifting method*	4
NSHIFT - *Number of shifts per match*	5
ISWAP - *Swapping method*	0
NSWAP - *Number of swap structures per shift*	5

outlined in Section 3 show how the ISHIFT = 4 method can help in identifying the most favourable interface separation. ARTEMIS is under continuous development at the University of Exeter, with the intent of adding additional features, tools and methodologies for predicting the structure and properties of interfaces.

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.

Data availability

The software supporting this publication are openly available from the University of Exeter's institutional repository at: http://hdl.handle.net/10871/121480. The source code and working examples can be found in the compressed file obtainable from http://www.artemis-materials.co.uk/.

Acknowledgements

Via our membership of the UK's HEC Materials Chemistry Consortium, which is funded by EPSRC, United Kingdom (EP/L000202, EP/R029431), this work used the ARCHER UK National Supercomputing Service (http://www.archer.ac.uk). Also, all of the authors acknowledge funding from the EPSRC, United Kingdom (EP/L015331/1). Finally, the authors thank Elizabeth L. Martin, Shane Graham Davies, Joe Pitfield, Edward Allery David Baker for their useful discussions. N. T. T, F. H. D and S. P. H developed the theory behind ARTEMIS. N. T. T developed the majority of the code. F. H. D. and I. E. M. R. developed some of the subroutines and functions used within the software. F. H. D., C. J. P and T. H. C helped to test the code and were heavily involved in discussions with N. T. T. about methods developed for and implemented the software. S. P. H. conceived and supervised the project and developed the initial prototype.

References

[1] H. Kroemer, Rev. Modern Phys. 73 (3) (2001) 783–793, http://dx.doi.org/10.1103/RevModPhys.73.783, URL https://link.aps.org/doi/10.1103/RevModPhys.73.783.
[2] B. Jamtveit, P. Meakin (Eds.), Growth, Dissolution and Pattern Formation in Geosystems, Springer Netherlands, Dordrecht, 1999, http://dx.doi.org/10.1007/978-94-015-9179-9, URL http://link.springer.com/10.1007/978-94-015-9179-9.
[3] R.L. Anderson, IBM J. Res. Dev. 4 (3) (1960) 283–287, http://dx.doi.org/10.1147/rd.43.0283.
[4] D.O. Scanlon, C.W. Dunnill, J. Buckeridge, S.A. Shevlin, A.J. Logsdail, S.M. Woodley, C.R.A. Catlow, M.J. Powell, R.G. Palgrave, I.P. Parkin, G.W. Watson, T.W. Keal, P. Sherwood, A. Walsh, A.A. Sokol, Nature Mater. 12 (9) (2013) 798–801, http://dx.doi.org/10.1038/nmat3697.
[5] R. Saive, H. Emmer, C.T. Chen, C. Zhang, C. Honsberg, H. Atwater, IEEE J. Photovolt. 8 (6) (2018) 1568–1576, http://dx.doi.org/10.1109/JPHOTOV.2018.2861724.
[6] F.H. Davies, C.J. Price, N.T. Taylor, S.G. Davies, S.P. Hepplestone, Manuscript submitted for publication, 2020.
[7] N.F. Mott, Math. Proc. Cambridge Philos. Soc. 34 (4) (1938) 568–572, http://dx.doi.org/10.1017/S0305004100020570.
[8] J. Bardeen, Phys. Rev. 71 (10) (1947) 717–727, http://dx.doi.org/10.1103/PhysRev.71.717.
[9] K.T. Delaney, N.A. Spaldin, C.G. Van De Walle, Phys. Rev. B - Condens. Matter Mater. Phys. 81 (16) (2010) 1–11, http://dx.doi.org/10.1103/PhysRevB.81.165312.
[10] S.-M. Lee, D.G. Cahill, R. Venkatasubramanian, Appl. Phys. Lett. 70 (22) (1997) 2957–2959, http://dx.doi.org/10.1063/1.118755.
[11] S.P. Hepplestone, G.P. Srivastava, Phys. Rev. B - Condens. Matter Mater. Phys. 84 (11) (2011) 1–13, http://dx.doi.org/10.1103/PhysRevB.84.115326.
[12] L. Wang, X.H. Zhong, Y.X. Zhao, J.S. Yang, S.Y. Tao, W. Zhang, Y. Wang, X.G. Sun, Int. J. Heat Mass Transfer 79 (2014) 954–967, http://dx.doi.org/10.1016/j.ijheatmasstransfer.2014.08.088.
[13] R.M. Martin, J. Vac. Sci. Technol. 17 (5) (1980) 978–981, http://dx.doi.org/10.1116/1.570651.
[14] S.P. Hepplestone, P.V. Sushko, J. Appl. Phys. 116 (19) (2014) http://dx.doi.org/10.1063/1.4902009.
[15] F. Liu, D.Y. Xie, M. Bugnet, T. Majdi, J.S. Preston, J. Wang, G. zhen Zhu, Adv. Mater. Interfaces 5 (12) (2018) 2–7, http://dx.doi.org/10.1002/admi.201701664.
[16] H. Yoo, R. Engelke, S. Carr, S. Fang, K. Zhang, P. Cazeaux, S.H. Sung, R. Hovden, A.W. Tsen, T. Taniguchi, K. Watanabe, G.C. Yi, M. Kim, M. Luskin, E.B. Tadmor, E. Kaxiras, P. Kim, Nature Mater. 18 (5) (2019) 448–453, http://dx.doi.org/10.1038/s41563-019-0346-z, arXiv:1804.03806.
[17] S.P. Hepplestone, G.P. Srivastava, Phys. Rev. B - Condens. Matter Mater. Phys. 82 (14) (2010) 1–6, http://dx.doi.org/10.1103/PhysRevB.82.144303.
[18] T. Wu, P. Gao, Materials (Basel) 11 (6) (2018) 1–32, http://dx.doi.org/10.3390/ma11060999.
[19] J.F. Wager, Thin Solid Films 516 (8) (2008) 1755–1764, http://dx.doi.org/10.1016/j.tsf.2007.06.164.
[20] S. Gariglio, N. Reyren, A.D. Caviglia, J.M. Triscone, J. Phys. Condens. Matter 21 (16) (2009) 2–5, http://dx.doi.org/10.1088/0953-8984/21/16/164213.
[21] M. Ribeiro, L.R. Fonseca, L.G. Ferreira, Epl 94 (2) (2011) http://dx.doi.org/10.1209/0295-5075/94/27001.
[22] U.-C. Chung, C. Elissalde, F. Mompiou, J. Majimel, S. Gomez, C. Estournès, S. Marinel, A. Klein, F. Weill, D. Michau, S. Mornet, M. Maglione, J. Am. Ceram. Soc. 93 (3) (2010) 865–874, http://dx.doi.org/10.1111/j.1551-2916.2009.03474.x.
[23] G. Schusteritsch, S.P. Hepplestone, C.J. Pickard, Phys. Rev. B - Condens. Matter Mater. Phys. 92 (5) (2015) 1–7, http://dx.doi.org/10.1103/PhysRevB.92.054105.
[24] N.T. Taylor, F.H. Davies, S.P. Hepplestone, Mater. Res. Express 4 (12) (2017) 125904, http://dx.doi.org/10.1088/2053-1591/aa99e8.
[25] B. Hinterleitner, I. Knapp, M. Poneder, Y. Shi, H. Müller, G. Eguchi, C. Eisenmenger-Sittner, M. Stöger-Pollach, Y. Kakefuda, N. Kawamoto, Q. Guo, T. Baba, T. Mori, S. Ullah, X.Q. Chen, E. Bauer, Nature 576 (7785) (2019) 85–90, http://dx.doi.org/10.1038/s41586-019-1751-9.
[26] A. Ohtomo, H.Y. Hwang, Nature 427 (6973) (2004) 423–426, http://dx.doi.org/10.1038/nature04773.

[27] H. Hwang, Y. Iwasa, M. Kawasaki, B. Keimer, N. Nagaosa, Y. Tokura, Nature Mater. 11 (2) (2012) 103–113, http://dx.doi.org/10.1038/nmat3223.
[28] N.T. Taylor, F.H. Davies, S.G. Davies, C.J. Price, S.P. Hepplestone, Adv. Mater. 31 (51) (2019) 1904746, http://dx.doi.org/10.1002/adma.201904746.
[29] W.H. Zou, X.D. Han, R. Wang, Z. Zhang, W.Z. Zhang, J.K. Lai, Mater. Sci. Eng. A 219 (1–2) (1996) 142–147, http://dx.doi.org/10.1016/S0921-5093(96)10418-4.
[30] L. Zou, J. Li, D. Zakharov, E.A. Stach, G. Zhou, Nature Commun. 8 (1) (2017) 1–8, http://dx.doi.org/10.1038/s41467-017-00371-4.
[31] E.D. Specht, G.E. Ice, C.J. Peters, C.J. Sparks, N. Lucas, X.-M. Zhu, R. Moret, H. Morkoç, Phys. Rev. B 43 (15) (1991) 12425–12430, http://dx.doi.org/10.1103/PhysRevB.43.12425, arXiv:arXiv:1011.1669v3.
[32] H. Chen, Mater. Chem. Phys. 43 (2) (1996) 116–125, http://dx.doi.org/10.1016/0254-0584(95)01618-5.
[33] D.A. Bonnell, I. Solomon, G.S. Rohrer, C. Warner, Acta Metall. Mater. 40 (SUPPL.) (1992) 161–171, http://dx.doi.org/10.1016/0956-7151(92)90276-K.
[34] N.I. Plusnin, J. Electron Spectrosc. Relat. Phenom. 137–140 (SPEC. ISS.) (2004) 161–164, http://dx.doi.org/10.1016/j.elspec.2004.02.091.
[35] M.J. Zachman, J.A. Hachtel, J.C. Idrobo, M. Chi, Angew. Chem. - Int. Ed. (2019) 1384–1396, http://dx.doi.org/10.1002/anie.201902993.
[36] A. Zur, T.C. McGill, J. Appl. Phys. 55 (2) (1984) 378–386, http://dx.doi.org/10.1063/1.333084, URL http://aip.scitation.org/doi/10.1063/1.333084.
[37] A.M. Raclariu, S. Deshpande, J. Bruggemann, W. Zhuge, T.H. Yu, C. Ratsch, S. Shankar, Comput. Mater. Sci. 108 (PA) (2015) 88–93, http://dx.doi.org/10.1016/j.commatsci.2015.05.023.
[38] K. Mathew, A.K. Singh, J.J. Gabriel, K. Choudhary, S.B. Sinnott, A.V. Davydov, F. Tavazza, R.G. Hennig, Comput. Mater. Sci. 122 (2016) 183–190, http://dx.doi.org/10.1016/j.commatsci.2016.05.020.
[39] S. Daniele, J. Line, S. Søren, S. Kurt, J. Phys. Condens. Matter 29 (18) (2017) 185901.
[40] L. Jelver, P.M. Larsen, D. Stradi, K. Stokbro, K.W. Jacobsen, Phys. Rev. B 96 (8) (2017) 1–7, http://dx.doi.org/10.1103/PhysRevB.96.085306.
[41] S. von Alfthan, P.D. Haynes, K. Kaski, A.P. Sutton, Phys. Rev. Lett. 96 (5) (2006) 055505, http://dx.doi.org/10.1103/PhysRevLett.96.055505, URL https://link.aps.org/doi/10.1103/PhysRevLett.96.055505.
[42] A.L. Chua, N.A. Benedek, L. Chen, M.W. Finnis, A.P. Sutton, Nature Mater. 9 (5) (2010) 418–422, http://dx.doi.org/10.1038/nmat2712.
[43] G. Schusteritsch, C.J. Pickard, Phys. Rev. B - Condens. Matter Mater. Phys. 90 (3) (2014) 1–7, http://dx.doi.org/10.1103/PhysRevB.90.035424, arXiv:1407.2153.
[44] K. Momma, F. Izumi, J. Appl. Crystallogr. 44 (6) (2011) 1272–1276, http://dx.doi.org/10.1107/S0021889811038970.
[45] A.K. Lenstra, H.W. Lenstra, L. Lovász, Math. Ann. 261 (4) (1982) 515–534, http://dx.doi.org/10.1007/BF01457454.
[46] A.K. Geim, I.V. Grigorieva, Nature 499 (7459) (2013) 419–425, http://dx.doi.org/10.1038/nature12385, arXiv:1307.6718.
[47] I. Novak, Eur. J. Phys. 16 (4) (1995) 151–153, http://dx.doi.org/10.1088/0143-0807/16/4/001, arXiv:0143-0807/16/4/001.
[48] A.V. Kimmel, P.V. Sushko, J. Phys. Condens. Matter 27 (47) (2015) 475006, http://dx.doi.org/10.1088/0953-8984/27/47/475006.
[49] I. Galanakis, G. Bihlmayer, V. Bellini, N. Papanikolaou, R. Zeller, S. Blügel, P.H. Dederichs, Europhys. Lett. 58 (5) (2002) 751–757, http://dx.doi.org/10.1209/epl/i2002-00413-7, arXiv:0105207.
[50] Y.Y. Sun, H. Xu, Y.P. Feng, A.C. Huan, A.T. Wee, Phys. Rev. Lett. 93 (13) (2004) 1–4, http://dx.doi.org/10.1103/PhysRevLett.93.136102.
[51] J.E. Northrup, S. Froyen, Phys. Rev. B 50 (3) (1994) 2015–2018, http://dx.doi.org/10.1103/PhysRevB.50.2015.
[52] G. Srivastava, S. Jenkins, Phys. Rev. B - Condens. Matter Mater. Phys. 53 (19) (1996) 12589–12592, http://dx.doi.org/10.1103/PhysRevB.53.12589.
[53] S. Piskunov, E.A. Kotomin, E. Heifets, Microelectron. Eng. 81 (2–4) (2005) 472–477, http://dx.doi.org/10.1016/j.mee.2005.03.049.
[54] S. Piskunov, E.A. Kotomin, E. Heifets, J. Maier, R.I. Eglitis, G. Borstel, Surf. Sci. 575 (1–2) (2005) 75–88, http://dx.doi.org/10.1016/j.susc.2004.11.008.
[55] R.I. Eglitis, Internat. J. Modern Phys. B 28 (17) (2014) http://dx.doi.org/10.1142/S0217979214300096.
[56] R.I. Eglitis, A.I. Popov, J. Saudi Chem. Soc. 22 (4) (2018) 459–468, http://dx.doi.org/10.1016/j.jscs.2017.05.011.
[57] Y.L. Lee, D. Morgan, Phys. Rev. B - Condens. Matter Mater. Phys. 91 (19) (2015) http://dx.doi.org/10.1103/PhysRevB.91.195430.
[58] J.J. Brown, Z. Ke, W. Geng, A.J. Page, J. Phys. Chem. C 122 (26) (2018) 14590–14597, http://dx.doi.org/10.1021/acs.jpcc.8b03322.
[59] S. Jenkins, G. Srivastava, Phys. Rev. B - Condens. Matter Mater. Phys. 57 (15) (1998) 8794–8796, http://dx.doi.org/10.1103/PhysRevB.57.8794.
[60] R.J. Dixon, C.F. McConville, S.J. Jenkins, G.P. Srivastava, Phys. Rev. B 57 (20) (1998) R12701–R12704, http://dx.doi.org/10.1103/PhysRevB.57.R12701.
[61] R.H. Miwa, G.P. Srivastava, Phys. Rev. B - Condens. Matter Mater. Phys. 74 (3) (2006) 1–6, http://dx.doi.org/10.1103/PhysRevB.74.035301.