



ARL-TR-9799 • SEP 2023



Constructing a Model for Gamma-Alumina

by William Mattson

DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited.

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.



Constructing a Model for Gamma-Alumina

William Mattson

DEVCOM Army Research Laboratory

REPORT DOCUMENTATION PAGE

1. REPORT DATE		2. REPORT TYPE		3. DATES COVERED	
September 2023		Technical Report		START DATE	END DATE
				October 2020	September 2022
P4. TITLE AND SUBTITLE					
Constructing a Model for Gamma-Alumina					
5a. CONTRACT NUMBER		5b. GRANT NUMBER		5c. PROGRAM ELEMENT NUMBER	
5d. PROJECT NUMBER		5e. TASK NUMBER		5f. WORK UNIT NUMBER	
6. AUTHOR(S)					
William Mattson					
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)				8. PERFORMING ORGANIZATION REPORT NUMBER	
DEVCOM Army Research Laboratory ATTN: FCDD-RLA-WA Aberdeen Proving Ground, MD 21005				ARL-TR-9799	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSOR/MONITOR'S ACRONYM(S)	11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT					
DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited.					
13. SUPPLEMENTARY NOTES					
ORCID ID: William Mattson, 0000-0002-3922-1279					
14. ABSTRACT					
<p>Aluminum is widely used throughout industry, research, and development as its oxidized form alumina. In fact, alumina rapidly forms on the surface of aluminum, which makes its properties potentially important to any system with aluminum. Alumina has many phases, and one that is of interest at high temperatures for devices from catalytic converters to hydrogen separation systems is gamma-alumina. The importance of accurate simulations for the optimization of its chemical properties requires an accurate structure of gamma-alumina. Experimental evidence indicates that the structure is a spinel with aluminum vacancies at tetrahedral sites along planar defects. Crystallographic analysis has provided the details for inserting a single defect into a crystal. However, simulation requires many defects in complex arrangements. This report introduces a detailed method for constructing a periodic crystal of gamma-alumina consistent with experimental results and appropriate for simulation. Some important symmetries of the defects and other simulation considerations are also discussed. The MATLAB code implementing the method and an example of its use are provided.</p>					
15. SUBJECT TERMS					
Energy Sciences, Sciences of Extreme Materials, catalysts, spinel defects, nano-aluminum					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT		18. NUMBER OF PAGES
a. REPORT	b. ABSTRACT	c. THIS PAGE	UU		43
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED			
19a. NAME OF RESPONSIBLE PERSON				19b. PHONE NUMBER (Include area code)	
William Mattson				(410) 306-1903	

STANDARD FORM 298 (REV. 5/2020)

Prescribed by ANSI Std. Z39.18

Contents

List of Figures	iv
1. Introduction	1
2. The Spinel Structure	3
3. The Defect Structure	5
4. A Defect Structure for Simulations	7
5. Generating Structures	12
6. Simulations	13
7. Conclusions	14
8. References	16
Appendix A. Code	18
Appendix B. Example	27
List of Symbols, Abbreviations, and Acronyms	35
Distribution List	36

List of Figures

- Fig. 1 Thermogravimetric analysis of alumina-coated aluminum particles in both a steam and dry oxygen bath. Oxidation reaction rates in the steam bath are clearly more rapid than in the dry oxygen bath. 2
- Fig. 2 Building the spinel structure from sub-lattices. Left to right: oxygen sub-lattice, octahedral aluminum sub-lattice, tetrahedral aluminum sub-lattice, and full spinel lattice. 3
- Fig. 3 The Al_3O_4 alumina spinel structure. The structure is displayed with views from the top, right, and front, in the upper-left, upper-right, and lower frames, respectively. The oxygen atoms are presented as smaller than the aluminum atoms and in red for clarity. The aluminum atoms are colored by depth into the page in the top view. Tetrahedral aluminum atoms are pastel colors, while the octahedral aluminum atoms are in deeper hues. Only bonds between octahedral aluminum atoms with a depth layer are shown for pedagogical purposes..... 5
- Fig. 4 Transformed alumina, the crystallographic description. The crystallographic description from Martin et al. for the left and up transformation is shown. Oxygen is shown in red and aluminum is colored according to its depth into the page in the top frames. The upper frames look down into the second dimension, while the lower frames look back into the third dimension. Only bonds between octahedral aluminum atoms with a depth layer are shown for pedagogical purposes. In the first panel (left frame) of upper and lower frames the perfect spinel of Al_3O_4 can be seen, and the region to be removed is highlighted by the red rectangle. In the second panel the region has been removed and the translation to be performed is shown by the orange rectangles. The final panel shows the defected crystal with the blue ovals indicating pairs of aluminum atoms that the defect has brought too close together and one of each pair needs to be removed, for a total of four aluminum atom vacancies. The yellow and light blue aluminum atoms are in the octahedral positions, which for this simple defect are the preferred vacancies and need to be removed as a result of the left and up transformation..... 6
- Fig. 5 The oxygen lattice. Oxygen atoms are not bonded to other oxygen atoms, so no bonds are shown. If the oxygen lattice is thought of as a BCC lattice with the unit cell vectors along the face diagonals, then the left and up transformation is simply a move from the same position in one unit cell to another, and so the defect does not affect the oxygen lattice. Therefore, nothing needs to be done to the oxygen atoms to create this defect. 8

- Fig. 6 Octahedral alumina colored by layer (bonds only shown within layers). The left and up transformation simply shifts atoms in the orange and blue layers from one site to another within the layer (shown in the sets of blue circle pairs on the left). Therefore, like the oxygen, no transformation is required for these layers. However, the same transformation on the green and purple layers (shown in the sets of blue circle pairs on the right) causes them to swap layers. Therefore, a swapping transformation is needed for these layers. 9
- Fig. 7 Transformed octahedral aluminum (bonds only shown within layers). In the lower frame the swapping of the green and purple layers is shown, as highlighted in the light blue ovals. In the upper frame, the broken bonds in the green and purple layers due to the defect are highlighted with blue ovals. Almost no change is seen on the right frame since the surface from that point of view is unchanged; however, a close inspection shows a few bonds have changed color due to the defect. 9
- Fig. 8 Tetrahedral aluminum. Tetrahedral atoms are not bonded to other tetrahedral atoms, so no bonds are shown. The left and up transformation involves the light green layer yellow layer swapping (circle pairs on the left) and the light purple and light blue layer swapping (circles pairs on the right). Therefore, all layers need to be transformed. 10
- Fig. 9 Transformed tetrahedral aluminum. In the lower frame the transformed region is designated by the blue rectangle. The upper-left frame designates the defect planes with blue ovals. The upper-right frame clearly shows the mixing of layers, with light green and yellow layers interlaced, as well as light blue and light purple. 10
- Fig. 10 Transformed alumina: entire crystal after the transformation with oxygen atoms shown in red and aluminum atoms colored by layer with bonds only shown between aluminum atoms within a layer. The defect planes are designated by dashed black lines. No plane is designated in the right frame since they are normal to the point of view. Compare with Fig. 4, which shows the same defect as the left one here from the crystallographic description. 11

1. Introduction

Aluminum particles are a common additive to energetic materials to improve performance due to their high energy content, reactivity, reaction mass, and ready availability.¹⁻³ As an additive, aluminum has demonstrated thermobaric effects, increasing the temperature and pressure of reactions.³ However, the limiting factor in the oxidation of aluminum is the timescale on which the oxygen penetrates the aluminum particle.⁴ So, for faster reactions smaller particles are preferable, since there is less distance for the oxygen to travel through the particle. This diffusion of oxygen into the aluminum is hindered by the alumina layer that forms on the surface of the aluminum particles to a depth of between 2 and 6 nm,¹ which is largely independent of particle size. Therefore, smaller particles have a larger ratio of the energetically inert alumina on the surface to the reactive aluminum in the core of the particle, decreasing the available energy by mass or volume. Several methods have been investigated to mitigate these competing limiting mechanisms for aluminum particles, including coatings to prevent oxidation, additives to increase the available oxygen, and heat to facilitate reaction.⁵ To optimize coatings and additives, a fundamental understanding of the mechanism for oxygen penetrating the aluminum particles is required. Of primary importance are the mechanisms that will allow oxygen to reach unreacted aluminum more rapidly.

Schoenitz et al.⁶ demonstrated that aluminum particles in thermogravimetric experiments react significantly more rapidly in the presence of water than in a dry environment.^{6,7} As shown in Fig. 1, after what the data indicates to be the alumina shell transitioning from amorphous to gamma (γ)-alumina and the aluminum core melting, the aluminum core oxidizes in the presence of water at significantly faster timescales and lower temperatures than it does in the presence of dry oxygen. It also appears to fully oxidize before the transition to α -alumina. In dry oxygen there is a slight increase in the oxygen absorption at the expected transition to γ -alumina, then a much more gradual increase in oxygen absorption until the expected transition to α -alumina, at which time the absorption significantly slows. To attempt to verify the alumina is γ -alumina during the more rapid oxidation of the aluminum core, they collected samples just below and above the melting point of the aluminum core and performed X-ray diffraction (XRD) on the samples. The only crystalline phase of alumina observed in either sample was γ -alumina. To understand this mechanism for accelerated oxygen absorption, a realistic structure of γ -alumina is needed.

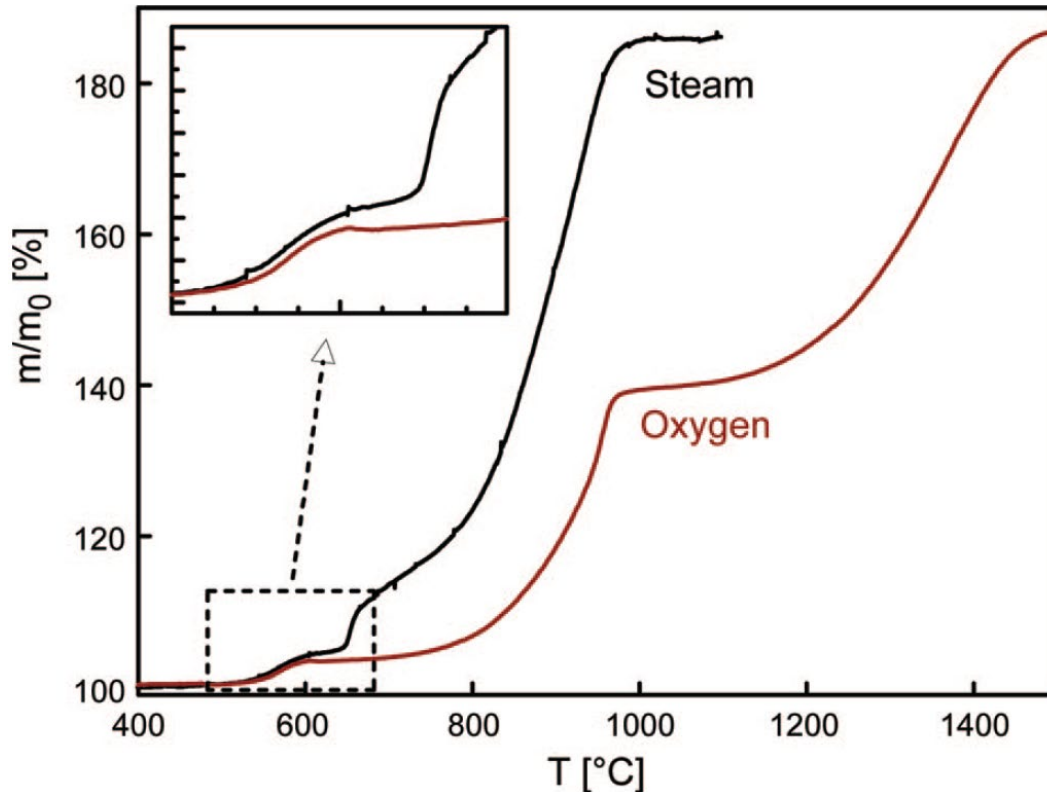


Fig. 1 Thermogravimetric analysis of alumina-coated aluminum particles in both a steam and dry oxygen bath. Oxidation reaction rates in the steam bath are clearly more rapid than in the dry oxygen bath.⁶

Other XRD experiments⁸ on γ -alumina give a spinel structure with a stoichiometry of Al_3O_4 . To get the correct Al_2O_3 stoichiometry, one in nine aluminum atoms must be replaced by a vacancy. Different choices of which aluminum atoms are removed are likely to have significant effects on the material properties of γ -alumina. Successful modelling will depend on choosing an appropriate set of vacancies. Random removal has been shown to not agree with selected area electron diffraction (SAED) or XRD.⁹ Recent spectroscopy indicates that the vacancies form from planar defects in the spinel structure that cause pairs of aluminum atoms (one tetrahedral and one octahedral) to come close together, and one needs to be replaced by a vacancy.¹⁰ Of the two possible vacancies, density functional theory (DFT) calculations on perfect crystals indicate that the tetrahedral vacancy is energetically preferable.¹¹ These defect planes form a nanoscale 3D network effectively dicing the γ -alumina. As the vacancies are concentrated on the defect planes, making them more porous and less tightly bound, these planes are likely weaker than the bulk material. When the aluminum core of the particle melts and expands, it will strain the alumina surface. The γ -alumina is likely to stretch around the defected planes, and as the strain from the core builds and stretching increases,

pores may open sufficiently to allow for the rapid infusion of small particles, including water, hydrogen, or hydroxy radicals. Therefore, the detailed nature of the γ -alumina structure may play a primary role in the mechanism of interest, and so complex realistic structures of γ -alumina is needed.

As the spinel structure has an Al_3O_4 stoichiometry, the vacancies along the defect planes reduce the aluminum content and give the entire structure its Al_2O_3 stoichiometry. However, since the vacancies are concentrated along the defect planes, the layers that contain those planes are aluminum poor and oxygen rich. Conversely, as no aluminum atoms have been removed from the small regions between the planes, they will have a local stoichiometry unchanged from the Al_3O_4 spinel structure, making them aluminum rich-oxygen poor blocks. Therefore, these structures are not uniform on a nanoscale, and when generating a sample structure, it will not necessarily have the exact stoichiometry expected if it does not contain a precise mixture of defect planes and spinel regions. If the precise stoichiometry is required or if maintaining charge balance is desirable for the target simulations, it is important to take care when choosing defect planes to ensure that the proper overall stoichiometry is obtained.

2. The Spinel Structure

It is important to start with an understanding of the spinel structure. The spinel structure can be thought of as three intercalated sub-lattices, as shown in Fig. 2. The first sub-lattice, shown in the first pane, is a unitary simple cubic lattice where every other site is oxygen. The oxygen sub-lattice can also be thought of as a body-centered cubic (BCC) lattice by rotating the perspective along a cubic diagonal.

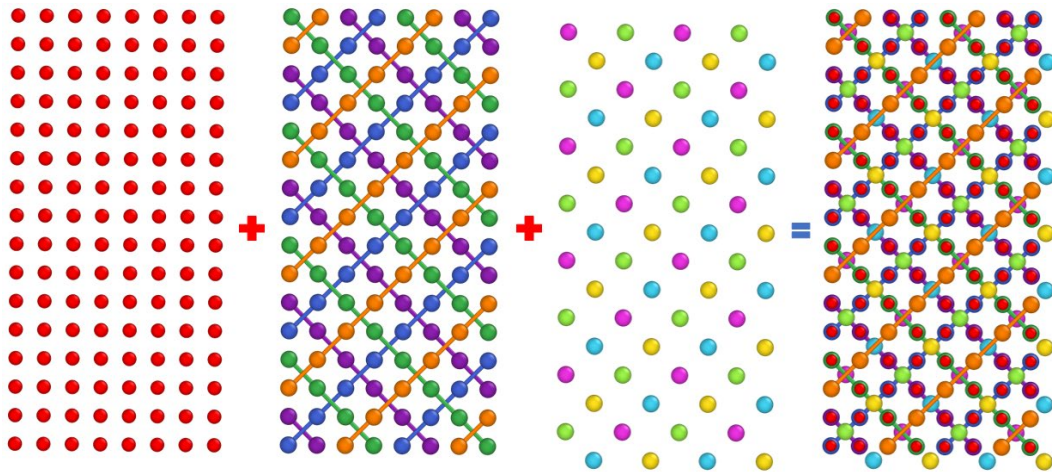


Fig. 2 Building the spinel structure from sub-lattices. Left to right: oxygen sub-lattice, octahedral aluminum sub-lattice, tetrahedral aluminum sub-lattice, and full spinel lattice.

The second sub-lattice, shown in the second pane, is on the same simple cubic lattice as the oxygen, with the aluminum atoms occupying half of the unoccupied simple cubic sites. To understand the arrangement of these aluminum atoms, it is useful to consider the sub-lattice as a series of four single layers repeating. The first single layer, shown in second pane by the orange aluminum atoms, has the aluminum atoms filling every other top-right to bottom-left diagonals not occupied by oxygen atoms. The second layer, shown in green, has every other top-left to bottom-right diagonal filled. The third layer, in blue, goes back to top right to bottom left, but now the diagonals that were left empty in the first layer are filled and those that were filled in the first layer are empty (each diagonal is shifted over one). The fourth layer, purple, is to the second like the third is to the first, with top-left to bottom-right diagonals. So, each of the 2D diagonals is occupied, in a series of four layers. These aluminum atom sites are considered octahedral, and they have six oxygen atoms for nearest neighbors.

The third sub-lattice, shown in the third pane, is another aluminum lattice where the aluminum atoms can occupy what would be the body-centered sites for the simple cubic lattice described previously. These are referred to as tetrahedral sites. This creates a sub-lattice where each aluminum atom has four oxygen atoms as its nearest neighbors. However, the sites must be chosen carefully to ensure that no octahedral aluminum atom is a nearest neighbor. Therefore, only one in eight of these sites can be, and are, occupied. The structure of this sub-lattice can be understood, once again, by considering a series of four layers, and colors are again used to indicate to which layer an atom belongs (it may be helpful to examine Fig. 3, which shows the full structure with the same coloring). In each layer of tetrahedral atoms, we consider the layers of octahedral aluminum atoms immediately above and below this layer. Only those tetrahedral sites that are a maximum distance away from the occupied diagonals in the adjacent layers will be occupied. In other words, the tetrahedral site at the center of the grid made by the occupied diagonals in the adjacent layers is occupied. So, the yellow tetrahedral aluminum atoms are in the centers of the grid made by the orange and purple octahedral aluminum atoms, similarly the light green centered between the orange and green, the light blue between the green and blue, and the light purple between the blue and purple. This ensures that the site occupied will not have any aluminum atom as a nearest neighbor, and it is the only site of the eight available for which this is true.

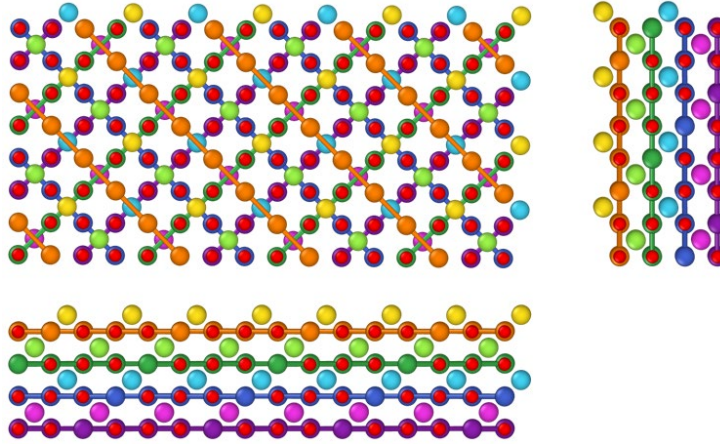


Fig. 3 The Al_3O_4 alumina spinel structure. The structure is displayed with views from the top, right, and front, in the upper-left, upper-right, and lower frames, respectively. The oxygen atoms are presented as smaller than the aluminum atoms and in red for clarity. The aluminum atoms are colored by depth into the page in the top view. Tetrahedral aluminum atoms are pastel colors, while the octahedral aluminum atoms are in deeper hues. Only bonds between octahedral aluminum atoms with a depth layer are shown for pedagogical purposes.

This gives the Al_3O_4 stoichiometry with oxygen taking four of the eight octahedral sites, and aluminum taking two of the eight octahedral sites and one of the eight tetrahedral sites. Interestingly, the crystal has a handedness that is determined by the order chosen for the octahedral aluminum atom layers. If the second and fourth layer are swapped in their order, then the handedness is changed. This becomes particularly apparent when studying the ordering of the tetrahedral atoms, as they will spiral either clockwise or counterclockwise into the plane depending on the choice of the ordering of the octahedral planes. This asymmetry will come in later when the defects also have a handedness.

3. The Defect Structure

The difference between the perfect crystal and one with a single defect plane can then be simply described from the crystallography in a frame of reference where the plane appears vertically and into the page. There are three steps to generate the defect, as shown in Fig. 4 with the top and front views of each step: first, the defect plane is selected and a slab of unit thickness (the unit of the simple cubic lattice) of the crystal is removed; second, the portion of crystal on the right is translated one unit to the left and up one unit (the order of the translation is commutable); and third, as the previous steps make some tetrahedral and octahedral aluminum atoms nearest neighbors, the tetrahedral of each pair of nearest neighbor aluminum atoms is removed to create vacancies. We will refer to this set of steps as the left and up transformation. This corresponds to removing four aluminum atoms in the merged layer, reducing the aluminum content of the local region considerably, and of the

whole crystal slightly. Multiple defect planes are usually required to reach the desired stoichiometry, and while the defect planes can all be parallel, experiment shows that they are distributed in all three dimensions. Furthermore, since one in nine aluminum atoms need to be removed for the correct stoichiometry, care must be taken in choosing defects and dimensions if that is desired.

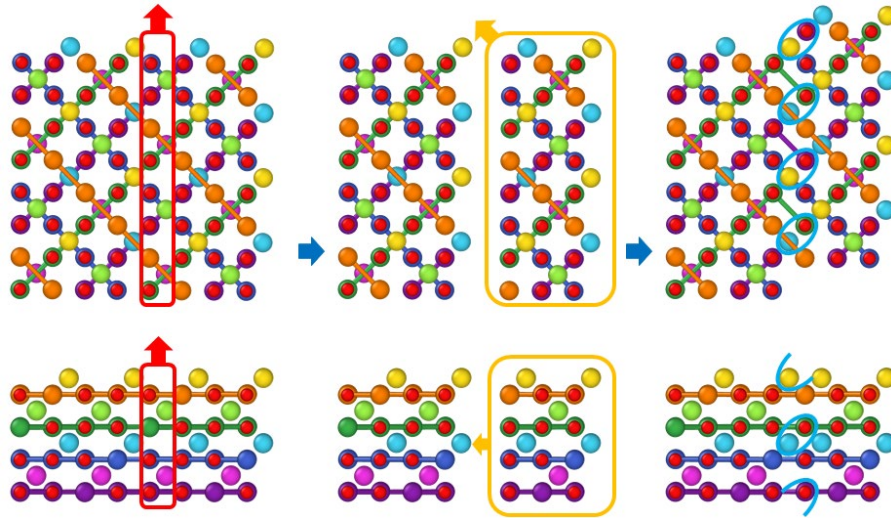


Fig. 4 Transformed alumina, the crystallographic description. The crystallographic description from Martin et al.¹⁰ for the left and up transformation is shown. Oxygen is shown in red and aluminum is colored according to its depth into the page in the top frames. The upper frames look down into the second dimension, while the lower frames look back into the third dimension. Only bonds between octahedral aluminum atoms with a depth layer are shown for pedagogical purposes. In the first panel (left frame) of upper and lower frames the perfect spinel of Al_3O_4 can be seen, and the region to be removed is highlighted by the red rectangle. In the second panel the region has been removed and the translation to be performed is shown by the orange rectangles. The final panel shows the defected crystal with the blue ovals indicating pairs of aluminum atoms that the defect has brought too close together and one of each pair needs to be removed, for a total of four aluminum atom vacancies. The yellow and light blue aluminum atoms are in the octahedral positions, which for this simple defect are the preferred vacancies and need to be removed as a result of the left and up transformation.

These steps in left and up transformation are rotationally invariant with respect to the Cartesian axes and can be followed regardless of the orientation of the defect plane. Additionally, a left and down transformation is equivalent but results in a defect plane with a different handedness. The importance of this is discussed in the next section. Whether left and up or down, these steps can be repeated multiple times to generate more defect planes to create the desired plane structure and stoichiometry. However, changes in the order in which the defect planes are created may result in different crystal structures. Additionally, at the intersection of planes, depending on where the defect planes are in the structure, the aluminum atoms may not be in the proximate positions so that none need to be removed to form a

vacancy. So, changing the position and order of creation of the defect planes can change the stoichiometry. This makes the variation in vacancy creation at defect plane intersections complex to predict.

As the proposed structure is based on defect planes, it is important to understand the orientation and spacing of these planes. The defect planes occur in all three dimensions and are normal to the Cartesian axes. To get the correct stoichiometry the defect planes must have an appropriate average spacing. Furthermore, the defect structure is isotropic in the (100) and (010) directions but anisotropic in the (001) direction. Taking all these factors into consideration the spacing of the planes in bulk γ -alumina works out to an average of about 1.5 and 0.5 nm in the [(100), (010)] and (001) directions, respectively.

While this works out when averaging over large samples, getting the correct stoichiometry for small samples requires some experimentation. To facilitate this, we have developed a method that allows users to rapidly create small scale structural models of γ -alumina with defects in one or more dimensions at user specified positions, and with periodic boundary conditions (PBCs) so that they may be used in simulations representing bulk material. The user can then adjust the parameters to get the correct stoichiometry if desired, though some trial and error is required.

4. A Defect Structure for Simulations

The previous model of defect formation is useful for understanding the nature of the defect planes. However, it ignores the need for PBCs in simulations. When PBCs are required, unlike the example in Section 3, the defect planes must occur in pairs so that the translations to one side of the defect plane can be undone, otherwise an accidental defect will be formed across the PBC parallel with the defect plane. The implementation of this requirement invokes the need for a more complex transformational method discussed in this section. The code for generating the structures along with commentary is given in Appendix A.

To accommodate PBCs in simulations, a more complex model is needed for generating pairs of defect planes in γ -alumina. If a direct transformation operation is used on a region of the structure that is between the defects being introduced instead of the example in Section 3 where a section of crystal is removed, then not only can periodicity for PBCs be preserved but the total atom number can also be preserved. This makes it easier for simulation designers to obtain a system of a specific size. A direct transformation can be thought of as simply moving atoms to a new position within the unit cell, whole regions of atoms are not translated, no

slabs of atoms are removed, and if chosen carefully, atoms stay close to their original position.

In a direct transformation, the oxygen atoms are incredibly easy to transform, as they do not move at all, as can be seen in Fig. 5. Since the oxygen atoms occupy every other site, moving an atom left and up moves it to another oxygen site. The same is not true for aluminum. The defects in the defect planes differ from the pure crystal in the positions of the aluminum atoms. However, the transformation for the tetrahedral and the octahedral aluminum atoms are different so we will look at them individually.

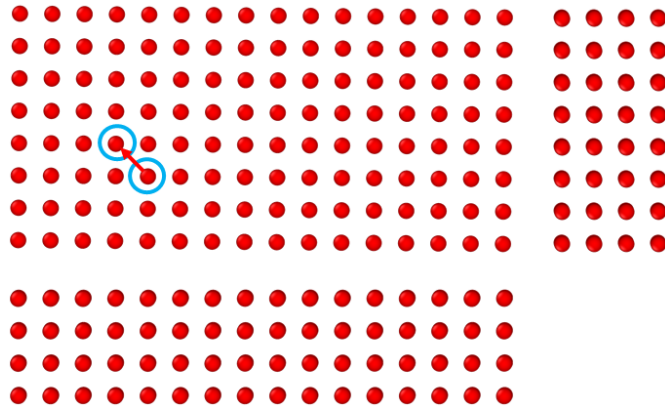


Fig. 5 The oxygen lattice. Oxygen atoms are not bonded to other oxygen atoms, so no bonds are shown. If the oxygen lattice is thought of as a BCC lattice with the unit cell vectors along the face diagonals, then the left and up transformation is simply a move from the same position in one unit cell to another, and so the defect does not affect the oxygen lattice. Therefore, nothing needs to be done to the oxygen atoms to create this defect.

In Fig. 6 we can see four layers of the octahedral aluminum atoms. For the transformation described in Fig. 4, we can see that shifting the top layer of orange-colored atoms left and up. The orange atoms will stay on orange atom position, so no transformation is required. The same is true for the third layer of blue atoms. However, applying the same transformation to the green or purple atoms results in them switching lattice sites. This corresponds to these two layers swapping Z coordinates in the region between the defects as seen in Fig. 7. For a left and down transformation, it is the first and third layers that are swapped, and the second and fourth layers stay the same.

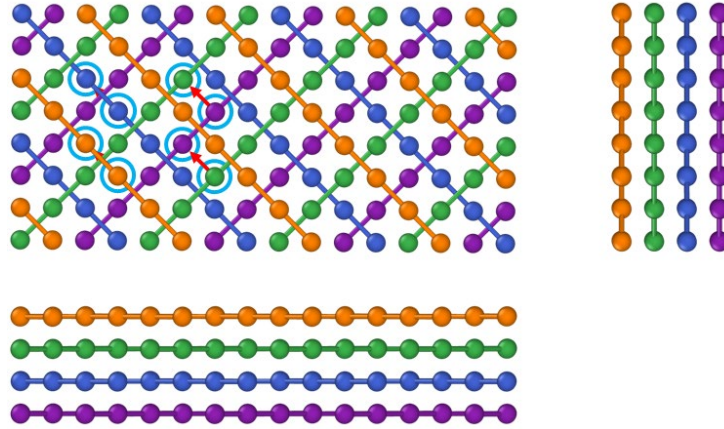


Fig. 6 Octahedral alumina colored by layer (bonds only shown within layers). The left and up transformation simply shifts atoms in the orange and blue layers from one site to another within the layer (shown in the sets of blue circle pairs on the left). Therefore, like the oxygen, no transformation is required for these layers. However, the same transformation on the green and purple layers (shown in the sets of blue circle pairs on the right) causes them to swap layers. Therefore, a swapping transformation is needed for these layers.

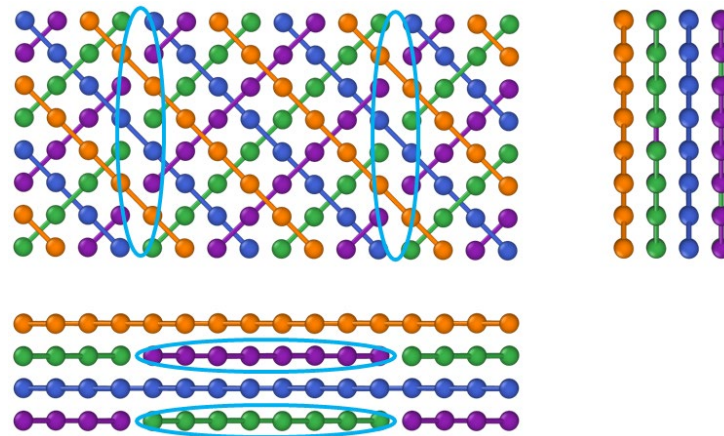


Fig. 7 Transformed octahedral aluminum (bonds only shown within layers). In the lower frame the swapping of the green and purple layers is shown, as highlighted in the light blue ovals. In the upper frame, the broken bonds in the green and purple layers due to the defect are highlighted with blue ovals. Almost no change is seen on the right frame since the surface from that point of view is unchanged; however, a close inspection shows a few bonds have changed color due to the defect.

For the tetrahedral aluminum atoms, shifting left and up results in the light green and yellow layers swapping Z coordinates as shown in Fig. 8, and light blue and light purple layers swapping as shown in Fig. 9. This is a fundamentally different translation than for the octahedral aluminum atoms, and it will be important in considering how to apply the direct transformation for larger systems. This transformation also has a different left and down version where the light green and light blue layers swap, as do the yellow and light purple.

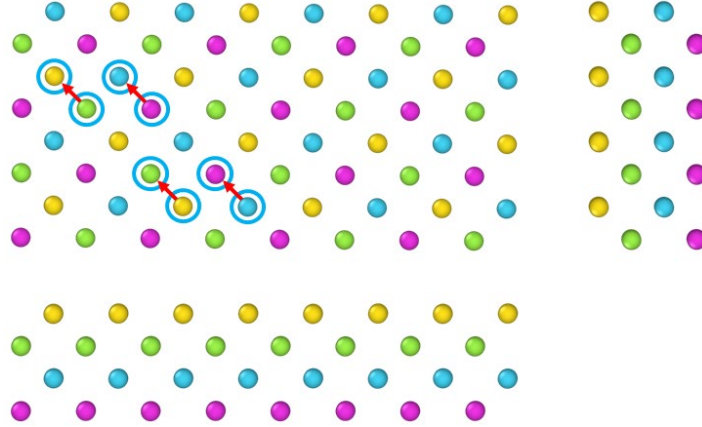


Fig. 8 Tetrahedral aluminum. Tetrahedral atoms are not bonded to other tetrahedral atoms, so no bonds are shown. The left and up transformation involves the light green layer yellow layer swapping (circle pairs on the left) and the light purple and light blue layer swapping (circles pairs on the right). Therefore, all layers need to be transformed.

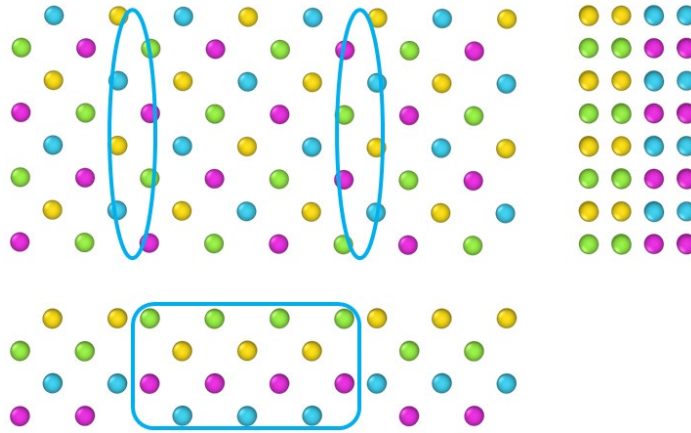


Fig. 9 Transformed tetrahedral aluminum. In the lower frame the transformed region is designated by the blue rectangle. The upper-left frame designates the defect planes with blue ovals. The upper-right frame clearly shows the mixing of layers, with light green and yellow layers interlaced, as well as light blue and light purple.

The final complete shifted structure can be seen in Fig. 10 and its details can be found in Appendix B. In the figure, blue ovals indicate pairs of aluminum atoms that the defect has brought too close together and one of each pair needs to be removed. This will lower the ratio of aluminum to oxygen atoms, approaching the desired stoichiometry for Al_2O_3 . Eight such pairs of atoms are represented in the figure. In these frames, four of the pairs are light blue tetrahedral and purple octahedral pairs (the purple atoms are hiding behind oxygen atoms, and there are two pairs for each oval in the lower frame). Four other pairs are green octahedral and yellow tetrahedral pairs (the green atoms may be hiding behind an oxygen atom). Some of the green-yellow pairs cross the PBCs as represented by the half ovals. The right frame shows the same pairs; however, because of the perspective,

the purple atoms are hiding behind the green, and the green atoms are hiding behind purple, making it a little harder to see. The figure shows both the defect given by the left and up transformation on the left and also the defect required for PBCs on the right. Examining the right-side defect, it corresponds to the exact same type of left and up defect, since left and up is rotationally equivalent to right and down. This is very convenient and verifies that the defect has rotational symmetry. Additionally, it verifies the conclusion that in addition to the left and up defect, which we will consider a left-handed defect, there should exist a left and down defect, or a right-handed defect, that is not rotationally equivalent. The right-handed defect is the one that corresponds to swapping different layers. This handedness may become important for larger simulations as a mixture of defects should be included to be representative because they are energetically equivalent. The code in Appendix A only includes left-handed defects, which for our smaller simulation purposes is sufficient.

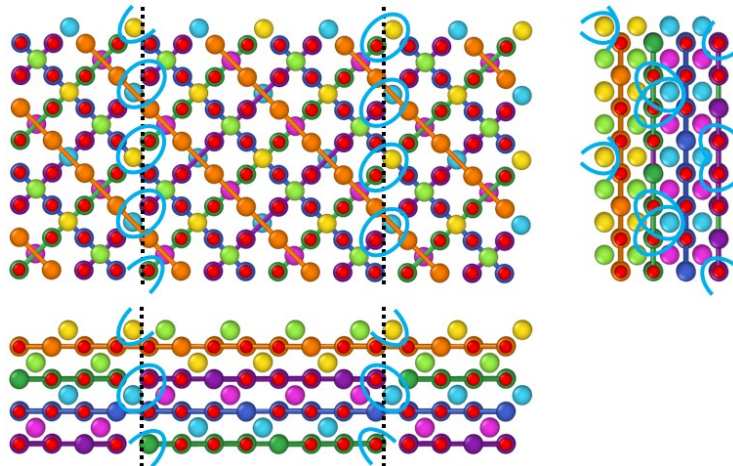


Fig. 10 Transformed alumina: entire crystal after the transformation with oxygen atoms shown in red and aluminum atoms colored by layer with bonds only shown between aluminum atoms within a layer. The defect planes are designated by dashed black lines. No plane is designated in the right frame since they are normal to the point of view. Compare with Fig. 4, which shows the same defect as the left one here from the crystallographic description.

These steps in left and up transformation are rotationally invariant with respect to the Cartesian axes and can be followed regardless of the orientation of the defect plane. They can be repeated multiple times to generate more defect planes to create the desired plane structure and stoichiometry. However, changes in the order in which the defect planes are created may result in different crystal structures. Additionally at the intersection of planes, depending on where the defect planes are in the structure, the aluminum atoms may not be in the proximate positions so that none need to be removed to form a vacancy. So, changing the position and order of creation of the defect planes can change the stoichiometry. As the variation in

vacancy creation at defect plane intersections is complex to predict, simple trial and error methods are used to obtain the desired stoichiometry. Other than stoichiometry, there is no clear metric for choosing one defect structure over another.

5. Generating Structures

Structures can be generated by running the code in Appendix A. The code includes several functions that should be saved individually in files with the same name as the function (case-sensitive). The main routine to call is `GammaAluminaStructure`. Its calling signature is `[coords, lv, atomTypes] = GammaAluminaStructure(superCell, latticeConstant, defectPlanes, defectDimension, octahedralVacancy, cutOff, numVacancies, removeDuplicates, vacancyLabel, fileName)`. The function writes a file and returns three variables. The return variables are `coords`, the matrix of atomic coordinates of the defected structure with each row corresponding to a single atom; `lv`, the lattice vector matrix; and `atomTypes`, the atom element label for each atom also indexed by row. There are 10 input parameters to the function:

Specifying the Supercell

- `SuperCell` is a comma-separated list of the number of replications in each dimension, where a [2,3,4] super cell will have 2 cells in the X dimension, 3 in the Y dimension and 4 in Z dimension.
- `LatticeConstant` is the spacing of the simple cubic lattice occupied by the oxygen and octahedral aluminum atoms.

Specifying the Defect Positions

- `DefectPlanes` is a semicolon-separated list of defect plane index pairs. Defect plane indices start at one, and there are four planes in a unit cell. The defect planes will be inserted just after the first index and just before the second index in the pair. The dimension that the defect plane is inserted in is specified by `defectDimension` and enumerated in XYZ order, such that a defect plane normal to the Z axis is specified by the number 3.
- `DefectDimension` is a comma-separated list that has the same order as the defect planes list.

Controlling the Function Behavior

- `OctahedralVacancy` is true if octahedral vacancies are desired and false if tetrahedral vacancies are desired.

- CutOff is used as the maximum distance for aluminum atoms to be separated by to be considered nearest neighbors and for one of them to be a candidate for a vacancy. A value of 1/4 is standard for the cutOff.
- NumVacancies. The number of desired vacancies up to the total number of nearest neighbor aluminum pairs can be created as specified in numVacancies. Use a value of -1 to specify that a vacancy will be created for every nearest neighbor aluminum pair.
- RemoveDuplicates is true if the vacancies should be created by removing the aluminum atom from the list of atoms coordinates and types or should be false to have the atom type for the vacancy replaced with vacancyLabel. The latter is useful for debugging. If removeDuplicates is true, then vacancyLabel should be a blank space.
- VacancyLabel is a string used to replace the atom type ['Al', 'O'] of atoms at the vacancy sites.

Specifying the XYZ File Name

- Filename is a string specifying the output file and can include a specified path.

The two supercells used in the figures describing a periodic system were generated with a superCell of [4,1,2], where in the three view figures, from the front view (bottom frame) X is to the right, Y is up, and Z is into the plane. The lattice constant is 2.0. For the defect-free structure, defectPlanes and defectDimensions are equal to the empty set, or vector of zero length specified by [], and the others do not matter as long as they are valid. For the defected structure, defectPlanes was [4,13], the defectDimension was [1], cutOff was 1/4, numVacancies was -1, removeDuplicates was true, and vacancyLabel was a blank space. Modifying the code requires familiarity with the inner workings of the code, and there are helpful comments included in the code to facilitate this.

6. Simulations

Several γ -alumina structures were tested and validated with DFT calculations. All DFT calculations were performed with CP2K using a mixed basis set consisting of Gaussians and planewaves.¹² The generalized gradient approximation¹³ was used to treat electron exchange and correlation, and calculations are performed with spin-polarization with the DZVP, or Double Zeta Valence plus Polarization, basis set¹⁴ and a planewave cut-off of 350 Rydbergs. Then, larger structures were created to use in classical MD simulations with Large-scale Atomic/Molecular Massively

Parallel Simulator (LAMMPS).¹⁵ Within LAMMPS we used the ReaxFF potential for aluminum, oxygen, hydrogen, and carbon, developed by Aktulga et al.¹⁶ We validated this potential for use in these simulations as well as simulations of graphene oxide, a proposed coating for alumina.

All simulations were performed on crystals that were stoichiometrically Al_2O_3 . While this is not necessarily a requirement for simulations, it is helpful to ensure the correct charge balance for the systems. We created simulation cells consistent with the method discussed in Section 4. The first structure has, for simplicity, defects in one dimension; the system included 192 atoms with Al_2O_3 stoichiometry. A second structure with defects in two dimensions with a total of 480 atoms is also given. The 192-atom structure gave similar results for both DFT and LAMMPS, with optimized geometries that were the same to within 1%. Both codes showed the structure to be stable for finite times at finite temperatures. While previous work demonstrated that tetrahedral aluminum vacancies were energetically favorable, we found in simulations of the 480-atom system that octahedral vacancies were preferable for some planar intersections. This was identified by some aluminum atoms near planar intersections moving from octahedral to vacant tetrahedral sites in the DFT optimizations. While this did not occur in the LAMMPS optimization, perhaps due to different methods, LAMMPS did show the structure to be similar in energy and stable. Further simulations were performed in LAMMPS on the 2D system where the structure was doubled in the direction without defects, for a total of 960 atoms. This structure was then adapted for simulation of a free surface, with the surface normal to the defect planes. The periodic boundary in dimension corresponding to the surface was removed, the simulation cell expanded to create a vacuum next to the surface, and the 240 atoms on the opposite side were held fixed, to approximate the bulk and allow only one free surface. The surface was allowed to relax at temperature (900 K, the temperature in the experiment that showed accelerated oxygen absorption), creating a relaxed unstrained surface. The surface showed minor restructuring of the surface layer only and proved to be stable at 900 K for simulations over 100 ps, demonstrating that this experimentally consistent structure for γ -alumina is stable for simulations.

7. Conclusions

We have demonstrated a method for constructing γ -alumina structures suitable for simulations that are consistent with the latest experimental evidence. These structures are Al_3O_4 spinels that include planar defects in three dimensions, with aluminum vacancies around the defects, pushing the large-scale stoichiometry to Al_2O_3 . The defects are shown to have a handedness that may be important to sample

properly for large-scale simulations. The resulting structures are shown to be stable, except for vacancy hopping, to temperatures of up to 900 K, and surface reconstruction.

8. References

1. Brooks KP, Beckstead MW. Dynamics of aluminum combustion. *J Propul Power*. 1995;11:769–780.
2. Wang LL, Munir ZA, Maximov YM. Thermite reactions: their utilization in the synthesis and processing of materials. *J Mater Sci*. 1993;28:3693–3708.
3. Rossi C, Zhang K, Esteve D, Alphonse P, Tailhades P, Vahlas C. Nanoenergetic materials for MEMS: a review. *J Microelectromech Syst*. 2007;16:919–931.
4. Dreizin EL. Metal-based reactive nanomaterials. *Prog Energy Combust Sci*. 2009;35:141–167.
5. Jiang Y, Deng S, Hong S, Zhao J, Huang S, Wu C-C, Gottfried JL, Nomura K-I, Li Y, Tiwari S, et al. Energetic performance of optically activated aluminum/graphene oxide composites. *ACS Nano*. 2018;12:11366–11375.
6. Schoenitz M, Chen C-M, Dreizin EL. Oxidation of aluminum particles in the presence of water. *J Phys Chem B*. 2009;113(15):5136–5140. doi:10.1021/jp807801m.
7. Wischert R, Laurent P, Cope´ret C, Delbecq F, Sautet P. Alumina: the essential and unexpected role of water for the structure, stability, and reactivity of “defect” sites, *J Am Chem Soc*. 2012;134:14430–14449.
8. Miller ME, Mixture ST. Idealizing γ -Al₂O₃: in situ determination of nonstoichiometric spinel defect structure. *J Phys Chem C*. 2010;114:13039–13046.
9. Ayoola HO, House SD, Bonifacio CS, Kisslinger K, Saidi WA, Yang JC. Evaluating the accuracy of common γ -Al₂O₃ structure models by selected area electron diffraction from high-quality crystalline γ -Al₂O₃, *Acta Mater*. 2020;182:257–266.
10. Rudolph M, Motylenko M, Rafaja D. Structure model of γ -Al₂O₃ based on planar defects. *IUCrJ*. 2019;6:116–127. ISSN: 2052-2525. <https://doi.org/10.1107/S2052252518015786>.
11. Ferreira AR, Martins MJF, Konstantinova E, Capaz RB, Souza WF, Chiaro SSX, Leitao AA. Direct comparison between two gamma-alumina structural models by DFT calculations. *J Solid State Chem*. 2011;184:1105–1111.

12. VandeVondele J, Krack M, Mohamed F, Parrinello M, Chassaing T, Hutter, J. QUICKSTEP: fast and accurate density functional calculations using a mixed Gaussian and plane waves approach. *Comp Phys Comm.* 2005;167:103–128.
13. Perdew JP, Burke K, Ernzerhof M. Generalized gradient approximation made simple. *Phys Rev Lett.* 1996;77:3865–3868.
14. Goedecker S, Teter M, Hutter J. Separable dual-space Gaussian pseudopotentials. *Phys Rev B.* 1996;54:1703.
15. Thompson AP, Aktulga HM, Berger R, Bolintineanu DS, Brown WM, Crozier PS, in 't Veld PJ, Kohlmeyer A, Moore SG, Nguyen TD, et al. LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso-, and continuum scales. *Comp Phys Comm.* 2022;271:10817.
16. Aktulga HM, Fogarty JC, Pandit SA, Grama AY. Parallel reactive molecular dynamics: Numerical methods and algorithmic techniques. *Parallel Comput.* 2012;38:245–259.

Appendix A. Code

The implementation of this method for generating the γ -Al₂O₃ structure proceeds with four major steps: 1) the Al₃O₄ spinel structure of the desired size is created, 2) the specified defect planes are created, 3) a list of aluminum atom pairs that are within a specified cutoff is created, and 4) the desired number of aluminum atoms chosen randomly from the list are removed.

```
function [realCoords, realLV, realAtomTypes] = GammaAluminaStructure( ...
    superCell, latticeConstant, defectPlanes, defectDimension, octahedralVacancy, ...
    cutOff, numVacancies, removeDuplicates, vacancyLabel, fileName )
%UNTITLED Summary of this function goes here
% Detailed explanation goes here

lc = 4.0000000 * latticeConstant;

lv = [lc          0.0000000000          0.0000000000
      0.0000000000          lc          0.0000000000
      0.0000000000          0.0000000000          lc];

superCellLV = [superCell(1),0,0;0,superCell(2),0;0,0,superCell(3)];

[oCoords,alTetraCoords, alOctaCoords, ...
tetraLayers, octaLayers, oTypes, alTetraTypes, alOctaTypes, elements] = ...
GenerateGammaFractionals();

[newOCoords,newAlTetraCoords, newAlOctaCoords, ...
newTetraLayers, newOctaLayers, ...
newOTypes, newAlTetraTypes, newAlOctaTypes] = GenerateGammaSuperCell( ...
oCoords,alTetraCoords, alOctaCoords, ...
tetraLayers, octaLayers, oTypes, alTetraTypes, alOctaTypes, superCell);

for i = 1:max(size(defectDimension ))
    [newAlTetraCoords, newAlOctaCoords, ...
    newTetraLayers, newOctaLayers, ~] = ...
    CreateGammaDefect( newAlTetraCoords, newAlOctaCoords, newTetraLayers, newOctaLayers, ...
    defectDimension(i), defectPlanes(i,:));
end

%The vacancies occur at the aluminum atoms that are brought into close
%proximity by the shift along the defect plane. If the possible vacancies
%are to be placed at the tetrahedral sites, they are simply all the
%tetrahedral site in that plane, but if they are octahedral then they are
%only some of the octahedral atoms that are within a small distance of
%the tetrahedral atoms. For robustness with complex
%assemblies of defects we simply find all atom pairs within some cutoff
%and add the Octahedral or Tetrahedral atoms to the list as specified.
[vacancies] = ...
    FindAtomsWithinCutOffForLayers( newAlTetraCoords, ...
    newAlOctaCoords, superCellLV, cutOff , octahedralVacancy);

[coords, atomTypes, vacancies] = ConcatGamma( ...
    newOCoords, newAlTetraCoords, newAlOctaCoords, ...
    newOTypes, newAlTetraTypes, newAlOctaTypes, ...
    vacancies, octahedralVacancy );

if( numVacancies > 0 || removeDuplicates )
    [newCoords, newAtomTypes] = CreateRandomVacancies( coords, atomTypes, ...
    vacancies, numVacancies, removeDuplicates, vacancyLabel);
else
    newCoords = coords;
    newAtomTypes = atomTypes;
end

realCoords = newCoords * lv;
realLV = superCellLV * lv;
realAtomTypes = newAtomTypes;
if( vacancyLabel(1,1) ~= ' ' )
    elements = [elements;vacancyLabel];
end
```

```

end

numAtomsOfElements = AtomTypeCount(realAtomTypes, elements);

WriteGammaAlXYZ(fileName, realLV, realCoords, realAtomTypes, ...
    superCell, defectPlanes, defectDimension, octahedralVacancy, ...
    cutOff, numVacancies, removeDuplicates, vacancyLabel, ...
    elements, numAtomsOfElements);

end

```

The fractional coordinates for γ -alumina are generated as the fractions for the alumina spinel, along with information needed to create the defect planes in a lossless manner, where the simulation size does not shrink. This is important if periodic boundary conditions are to be maintained. These coordinates and data are kept in separate structures for oxygen, tetrahedral aluminum, and octahedral aluminum to facilitate defect creation and identification of the appropriate aluminum atom to remove as the tetrahedral atoms are removed.

```

function [oxygenCoords, alTetraCoords, alOctaCoords, ...
    tetraLayers, octaLayers, oTypes, alTetraTypes, alOctaTypes, elements] = ...
    GenerateGammaFractionals()
%GenerateGammaFractionals Unit cell fractionals, atom types, and layer
%indices for the perfect spinel of gamma alumina Al3O4.
% Coordinates kept separately for translation when defects are inserted.
alLabel = 'Al';
oLabel = 'O ';
oxygenCoords = [
    0.7500 0.7500 0.7500 0.2500 0.2500 0.2500
    0.5000 0.7500 0 0.7500
    0.5000 0 0.2500
    0 0.7500 0.5000
    0 0.2500 0.5000
    0.7500 0.5000 0
    0.2500 0.5000 0
    0 0.5000 0.2500
    0 0.5000 0.7500
    0.5000 0.2500 0
    0.5000 0.7500 0
    0.2500 0 0.5000
    0.7500 0 0.5000
    0.2500 0.7500 0.7500
    0.7500 0.2500 0.2500
    0.5000 0.5000 0.2500
    0.5000 0.5000 0.7500
    0 0.2500 0
    0 0.7500 0
    0 0 0.7500
    0 0 0.2500
    0.5000 0.7500 0.5000
    0.5000 0.2500 0.5000
    0.7500 0.2500 0.7500
    0.2500 0.7500 0.2500
    0.2500 0.5000 0.5000
    0.7500 0.5000 0.5000
    0.7500 0 0
    0.2500 0 0
    0.7500 0.7500 0.2500
    0.2500 0.2500 0.7500];
alOctaCoords = [
    0.5000 0.5000 0.5000
    0.2500 0.7500 0
    0.7500 0.2500 0
    0.7500 0 0.2500
    0.2500 0 0.7500
    0 0.2500 0.7500

```

```

    0    0.7500    0.2500
    0.5000    0    0
    0.2500    0.2500    0.5000
    0.7500    0.7500    0.5000
    0.7500    0.5000    0.7500
    0.2500    0.5000    0.2500
    0    0.5000    0
    0.5000    0.2500    0.2500
    0.5000    0.7500    0.7500
    0    0    0.5000];
alTetraCoords = [    0.1250    0.1250    0.1250
    0.8750    0.8750    0.8750
    0.6250    0.1250    0.6250
    0.3750    0.8750    0.3750
    0.1250    0.6250    0.6250
    0.8750    0.3750    0.3750
    0.6250    0.6250    0.1250
    0.3750    0.3750    0.8750];
octaLayers = [    2    2    2
    1    3    0
    3    1    0
    3    0    1
    1    0    3
    0    1    3
    0    3    1
    2    0    0
    1    1    2
    3    3    2
    3    2    3
    1    2    1
    0    2    0
    2    1    1
    2    3    3
    0    0    2];
tetraLayers = [    0    0    0
    3    3    3
    2    0    2
    1    3    1
    0    2    2
    3    1    1
    2    2    0
    1    1    3];

aluminum = [alLabel;alLabel];
aluminum = [aluminum;aluminum];
aluminum = [aluminum;aluminum];

oxygen = [oLabel;oLabel];
oxygen = [oxygen;oxygen];
oxygen = [oxygen;oxygen];
oxygen = [oxygen;oxygen];
oTypes = [oxygen;oxygen];
alTetraTypes = aluminum;
alOctaTypes = [aluminum;aluminum];

elements = [alLabel; oLabel];

end

```

Once the fractional coordinates have been obtained, the desired supercell is created from the fractional coordinates, and the administrative data is also replicated appropriately.

```

function [newOxygenCoords, newAlTetraCoords, newAlOctaCoords, ...
    newTetraLayers, newOctaLayers, ...
    newOTypes, newAlTetraTypes, newAlOctaTypes] = GenerateGammaSuperCell( ...
    oxygenCoords, alTetraCoords, alOctaCoords, ...
    tetraLayers, octaLayers, oTypes, alTetraTypes, alOctaTypes, superCell)
%GenerateGammaSuperCell Generates supercell for each component of the

```

```

%spinel for gamma alumina fractionals.
% First for oxygen, then for tetraAL, for octaAL, then atomTypes

% Oxygen
newOxygenCoords = GenerateFractionalSuperCell( oxygenCoords, superCell, 1 );

%Tetra Al Coords
newAlTetraCoords = GenerateFractionalSuperCell( alTetraCoords, superCell, 1 );

%Tetra Al Layers
newTetraLayers = GenerateFractionalSuperCell( tetraLayers, superCell, 4 );

%Octa Al Coords
newAlOctaCoords = GenerateFractionalSuperCell( alOctaCoords, superCell, 1 );

%Octa Al Layers
newOctaLayers = GenerateFractionalSuperCell( octaLayers, superCell, 4 );

%Oxygen Atom Types
newOTypes = GenerateAtomTypeSuperCell( oTypes, superCell );

%Tetra Al Atom Types
newAlTetraTypes = GenerateAtomTypeSuperCell( alTetraTypes, superCell );

%Octa Al Atom Types
newAlOctaTypes = GenerateAtomTypeSuperCell( alOctaTypes, superCell );

end

```

When generating a supercell, coordinates copied from the original cell must be translated into place. This is simple for fractionals as the translational unit is 1.

```

function [newCoords] = GenerateFractionalSuperCell(coords, superCell, spacing)
%GenerateFractionalSuperCell generates a supercell for single component
dims = max(size( superCell ));
factor = prod( superCell );

nAtoms = size(coords, 1);
newNAtoms = factor*nAtoms;
newCoords = zeros( newNAtoms, 3);
newCoords(1:nAtoms,:) = coords;

startI = 1;
endI = nAtoms;
startJ = endI + 1;
endJ = 2 * nAtoms;

for d = 1:dims
    lv = zeros(1,dims);
    lv(1,d) = spacing;
    for reps = 1:superCell(d) - 1

        newCoords(startJ:endJ,:) = newCoords(startI:endI,:) + ...
            ones(endI-startI+1,1) * ( reps * lv);
        startJ = endJ + 1;
        endJ = startJ + endI - startI;

    end

    endI = startJ - 1;
    endJ = startJ + endI - startI;

end
end

```

Generating non-coordinate atom type data for a super cell does not require any type of translation and so a separate function is used.

```

function [newTypes] = GenerateAtomTypeSuperCell(aTypes, superCell)
%GenerateAtomTypeSuperCell generates atom types for a single component of a
%super cell.
% Detailed explanation goes here
dims = max(size( superCell ));
factor = prod( superCell );

nAtoms = size(aTypes, 1);
newTypes(1:nAtoms,:) = aTypes;

startI = 1;
endI    = nAtoms;
startJ = endI + 1;
endJ    = 2 * nAtoms;

for d = 1:dims
    for reps = 1:superCell(d) - 1

        newTypes(startJ:endJ,:) = newTypes(startI:endI,:);
        startJ = endJ + 1;
        endJ   = startJ + endI - startI;

    end

    endI = startJ - 1;
    endJ = startJ + endI - startI;

end
end

function [newTetraCoords, newOctaCoords, ...
    newTetraLayers, newOctaLayers, newPossibleVacancies] = ...
    CreateGammaDefect( tetraCoords, octaCoords, tetraLayers, octaLayers, ...
        defectDimension, defectPlanes )
%CreateGammaDefect Inserts two defects perpendicular to the dimension
%specified in defectDimension, specified by the two lays in defectPlanes.
% Inserts a defect in a gamma spinel to correspond to gamma alumina,
% which requires two defect planes for periodic boundary conditions. The
% possible vacancies can be either octahedral or tetrahedral as
% specified. There are four layers in each dimension per unit cell.
% The defect transforms the region from the first plane + 1 to the last
% plane in defectPlanes.

if( defectDimension < 1 || defectDimension > 3 )
    error( 'CreateGammaDefect requires a defect dimension between one and three.' );
end
if( max(size(defectPlanes)) ~= 2 )
    error( 'CreateGammaDefect requires two defect plane layers.' );
end

newPossibleVacancies = [];
shiftingDim = [3,1,2];
dim = defectDimension;
tDim = shiftingDim(dim);
maxLayer = max(max(tetraLayers(:,dim)),max(octaLayers(:,dim)));
minLayer = min(min(tetraLayers(:,dim)),min(octaLayers(:,dim)));
if( min(defectPlanes) < minLayer || max(defectPlanes) > maxLayer )
    error( 'CreateGammaDefect requires a defect plane within the cell.' );
end

minLayer = min( defectPlanes );
maxLayer = max( defectPlanes );
if( maxLayer - minLayer < 2 )
    error( 'CreateGammaDefect requires two defect plane layers at least two layers abpart.' );
end

newTetraCoords = tetraCoords;
newTetraLayers = tetraLayers;
newOctaCoords = octaCoords;
newOctaLayers = octaLayers;

```

```

shiftingTetraAtoms = newTetraLayers(:,dim) > minLayer & ...
    newTetraLayers(:,dim) <= maxLayer;
shiftingOctaAtoms = newOctaLayers(:,dim) > minLayer & ...
    newOctaLayers(:,dim) <= maxLayer;
oddTetraLayers = mod(newTetraLayers(:,tDim),2);
evenOctaLayers = shiftingOctaAtoms & ~mod(newOctaLayers(:,tDim),2);

%TetraLayers
shiftingUpT = shiftingTetraAtoms & ~oddTetraLayers;
shiftingDownT = shiftingTetraAtoms & oddTetraLayers;
newTetraCoords(:, tDim) = newTetraCoords(:,tDim) + shiftingUpT ./4 - shiftingDownT ./ 4;
newTetraLayers(:, tDim) = newTetraLayers(:,tDim) + shiftingUpT - shiftingDownT;

%OctaLayers
shiftingUpO = evenOctaLayers & ~mod(newOctaLayers(:,tDim),4);
shiftingDownO = evenOctaLayers & ~mod(newOctaLayers(:,tDim)+2,4);
newOctaCoords(:, tDim) = newOctaCoords(:,tDim) + shiftingUpO ./ 2 - shiftingDownO ./ 2;
newOctaLayers(:, tDim) = newOctaLayers(:,tDim) + 2 * shiftingUpO - 2 * shiftingDownO;

end

function [atoms] = FindAtomsWithinCutOffForLayers(coordsA,coordsB, lv, cutOff, defectsInB )
%FindAtomsWithinCutOffForLayers Finds atom pairs with one atom in coordsA
%and one in coordsB that have a minimum separation of less than or equal to
%cutoff.
% Detailed explanation goes here
numA = size(coordsA,1);
numB = size(coordsB,1);
numFound = 0;
atomsA = [];
atomsB = [];
for i = 1:numA
    for j = 1:numB
        displacement = coordsA(i,:) - coordsB(j,:);
        cells = round( displacement/lv );
        displacement = displacement - cells * lv;
        distance = sqrt( sum( displacement.^2));
        if distance <= cutOff
            numFound = numFound + 1;
            atomsA(numFound) = i;
            atomsB(numFound) = j;
        end
    end
end

if defectsInB
    atoms = atomsB';
else
    atoms = atomsA';
end

end

function [newCoords, , newVacancies] = ConcatGamma( ...
    oCoords,alTetraCoords, alOctaCoords, ...
    oTypes, alTetraTypes, alOctaTypes, vacancies, octahedralVacancy)
%UNTITLED3 Summary of this function goes here
% Detailed explanation goes here
newCoords = [oCoords; alOctaCoords; alTetraCoords];
newAtomTypes = [oTypes; alOctaTypes; alTetraTypes];
numO = size(oCoords, 1);
numALO = size(alOctaCoords, 1);

if octahedralVacancy
    newVacancies = vacancies + numO;
else
    newVacancies = vacancies + numO + numALO;
end

```

```

end

function [newCoords, newAtomTypes] = CreateRandomVacancies( ...
    coords, atomTypes, inPossibleVacancies, numVacancies, ...
    removeDuplictes, vacancyLabel)
%UNTITLED Creates vacancies at random from the list of atomic indices in
%possible Vacancies.
% returns a modified coordinate and atom type list with numVacancies
% from the possibleVacancies either with atoms corresponding to the
% new vacancies removed, or if vacancyLabel is not equal
% to " ", the vacancies have atom type == vacancyLabel. Vacancies
% can have non-uniform probabilities if they occur more than once in the
% possible vacancy list, unless removeDuplictes is true, then duplicates
% in possibleVacancies are removed and the probability is uniform.

if removeDuplictes
    possibleVacancies = RemoveDuplictes( inPossibleVacancies );
else
    possibleVacancies = sort( inPossibleVacancies );
end

numPossibleVacancies = size( possibleVacancies, 1 );
if( numVacancies > 1 )
    numToRemove = numVacancies;
else
    if( removeDuplictes )
        numToRemove = numPossibleVacancies;
    else
        error('Can not use default -1 vacancies if not removing duplicates.');
```



```

%RemoveDuplicates Removes duplicates in one dimension vector indices
% Sorts then removes duplicate values.
num = size( indices, 1 );

if( size( num ) ~= 1 )
    error( 'RemoveDuplicates requires a one-dimensional array' );
end

if( num < 1 )
    error( 'RemoveDuplicates requires a non-empty array' );
end

cleanIndices = sort( indices );

index = 1;
while index < num
    if cleanIndices( index ) == cleanIndices( index + 1 )
        cleanIndices(index : num -1) = cleanIndices( index + 1 : num );
        num = num - 1;
    else
        index = index + 1;
    end
end

cleanIndices = cleanIndices(1:num);

end

function [numAtomsOfElement] = AtomTypeCount(atomType, elements)
%UNTITLED2 Summary of this function goes here
% Detailed explanation goes here
if( size( atomType,2 ) ~= size(elements,2))
    error( 'atom types and element labels must be the same length, use white space padding if needed.' );
end

numAtoms = size( atomType, 1);
numChar = size(atomType, 2);
numElement = size(elements, 1);
numAtomsOfElement = zeros(numElement, 1);

for element = 1:numElement
    for atom = 1:numAtoms
        match = true(1);
        for c =1:numChar
            if atomType(atom, c) ~= elements(element, c)
                match = false(1);
            end
        end
        if match
            numAtomsOfElement(element, 1) = numAtomsOfElement(element, 1) + 1;
        end
    end
end

end

```

Appendix B. Example

Herein is presented the coordinates for the structure shown in Fig. 10 of the main report. It includes 256 oxygen atoms with 128 octahedral and 64 tetrahedral aluminum atoms. The eight tetrahedral aluminum atoms that form nearest neighbor pairs with octahedral aluminum atoms have not been removed, for pedagogical reasons. A simple distance calculation on all pairs with periodic boundary conditions (PBCs) will identify them, and the tetrahedral atoms are the last in the list. Coordinates are given in unit coordinates.

```

o 2.00000000e+00 2.00000000e+00 2.00000000e+00
o 6.00000000e+00 6.00000000e+00 6.00000000e+00
o 4.00000000e+00 0.00000000e+00 6.00000000e+00
o 4.00000000e+00 0.00000000e+00 2.00000000e+00
o 0.00000000e+00 6.00000000e+00 4.00000000e+00
o 0.00000000e+00 2.00000000e+00 4.00000000e+00
o 6.00000000e+00 4.00000000e+00 0.00000000e+00
o 2.00000000e+00 4.00000000e+00 0.00000000e+00
o 0.00000000e+00 4.00000000e+00 2.00000000e+00
o 0.00000000e+00 4.00000000e+00 6.00000000e+00
o 4.00000000e+00 2.00000000e+00 0.00000000e+00
o 4.00000000e+00 6.00000000e+00 0.00000000e+00
o 2.00000000e+00 0.00000000e+00 4.00000000e+00
o 6.00000000e+00 0.00000000e+00 4.00000000e+00
o 2.00000000e+00 6.00000000e+00 6.00000000e+00
o 6.00000000e+00 2.00000000e+00 2.00000000e+00
o 4.00000000e+00 4.00000000e+00 2.00000000e+00
o 4.00000000e+00 4.00000000e+00 6.00000000e+00
o 0.00000000e+00 2.00000000e+00 0.00000000e+00
o 0.00000000e+00 6.00000000e+00 0.00000000e+00
o 0.00000000e+00 0.00000000e+00 6.00000000e+00
o 0.00000000e+00 0.00000000e+00 2.00000000e+00
o 4.00000000e+00 6.00000000e+00 4.00000000e+00
o 4.00000000e+00 2.00000000e+00 4.00000000e+00
o 6.00000000e+00 2.00000000e+00 6.00000000e+00
o 2.00000000e+00 6.00000000e+00 2.00000000e+00
o 2.00000000e+00 4.00000000e+00 4.00000000e+00
o 6.00000000e+00 4.00000000e+00 4.00000000e+00
o 6.00000000e+00 0.00000000e+00 0.00000000e+00
o 2.00000000e+00 0.00000000e+00 0.00000000e+00
o 6.00000000e+00 6.00000000e+00 2.00000000e+00
o 2.00000000e+00 2.00000000e+00 6.00000000e+00
o 1.00000000e+01 2.00000000e+00 2.00000000e+00
o 1.40000000e+01 6.00000000e+00 6.00000000e+00
o 1.20000000e+01 0.00000000e+00 6.00000000e+00
o 1.20000000e+01 0.00000000e+00 2.00000000e+00
o 8.00000000e+00 6.00000000e+00 4.00000000e+00
o 8.00000000e+00 2.00000000e+00 4.00000000e+00
o 1.40000000e+01 4.00000000e+00 0.00000000e+00
o 1.00000000e+01 4.00000000e+00 0.00000000e+00
o 8.00000000e+00 4.00000000e+00 2.00000000e+00
o 8.00000000e+00 4.00000000e+00 6.00000000e+00
o 1.20000000e+01 2.00000000e+00 0.00000000e+00
o 1.20000000e+01 6.00000000e+00 0.00000000e+00
o 1.00000000e+01 0.00000000e+00 4.00000000e+00
o 1.40000000e+01 0.00000000e+00 4.00000000e+00
o 1.00000000e+01 6.00000000e+00 6.00000000e+00
o 1.40000000e+01 2.00000000e+00 2.00000000e+00
o 1.20000000e+01 4.00000000e+00 2.00000000e+00
o 1.20000000e+01 4.00000000e+00 6.00000000e+00
o 8.00000000e+00 2.00000000e+00 0.00000000e+00
o 8.00000000e+00 6.00000000e+00 0.00000000e+00
o 8.00000000e+00 0.00000000e+00 6.00000000e+00
o 8.00000000e+00 0.00000000e+00 2.00000000e+00
o 1.20000000e+01 6.00000000e+00 4.00000000e+00
o 1.20000000e+01 2.00000000e+00 4.00000000e+00
o 1.40000000e+01 2.00000000e+00 6.00000000e+00
o 1.00000000e+01 6.00000000e+00 2.00000000e+00

```


Al 3.00000000e+01 6.00000000e+00 4.00000000e+00
Al 2.40000000e+01 0.00000000e+00 4.00000000e+00
Al 2.00000000e+01 1.20000000e+01 0.00000000e+00
Al 1.80000000e+01 1.00000000e+01 0.00000000e+00
Al 2.20000000e+01 1.40000000e+01 0.00000000e+00
Al 1.60000000e+01 8.00000000e+00 0.00000000e+00
Al 2.80000000e+01 1.20000000e+01 4.00000000e+00
Al 2.60000000e+01 1.00000000e+01 4.00000000e+00
Al 3.00000000e+01 1.40000000e+01 4.00000000e+00
Al 2.40000000e+01 8.00000000e+00 4.00000000e+00
Al 2.00000000e+00 0.00000000e+00 6.00000000e+00
Al 0.00000000e+00 2.00000000e+00 6.00000000e+00
Al 6.00000000e+00 4.00000000e+00 6.00000000e+00
Al 4.00000000e+00 6.00000000e+00 6.00000000e+00
Al 1.00000000e+01 0.00000000e+00 6.00000000e+00
Al 8.00000000e+00 2.00000000e+00 6.00000000e+00
Al 1.40000000e+01 4.00000000e+00 6.00000000e+00
Al 1.20000000e+01 6.00000000e+00 6.00000000e+00
Al 2.00000000e+00 8.00000000e+00 6.00000000e+00
Al 0.00000000e+00 1.00000000e+01 6.00000000e+00
Al 6.00000000e+00 1.20000000e+01 6.00000000e+00
Al 4.00000000e+00 1.40000000e+01 6.00000000e+00
Al 1.00000000e+01 8.00000000e+00 6.00000000e+00
Al 8.00000000e+00 1.00000000e+01 6.00000000e+00
Al 1.40000000e+01 1.20000000e+01 6.00000000e+00
Al 1.20000000e+01 1.40000000e+01 6.00000000e+00
Al 1.80000000e+01 0.00000000e+00 6.00000000e+00
Al 1.60000000e+01 2.00000000e+00 6.00000000e+00
Al 2.20000000e+01 4.00000000e+00 6.00000000e+00
Al 2.00000000e+01 6.00000000e+00 6.00000000e+00
Al 2.60000000e+01 0.00000000e+00 6.00000000e+00
Al 2.40000000e+01 2.00000000e+00 6.00000000e+00
Al 3.00000000e+01 4.00000000e+00 6.00000000e+00
Al 2.80000000e+01 6.00000000e+00 6.00000000e+00
Al 1.80000000e+01 8.00000000e+00 6.00000000e+00
Al 1.60000000e+01 1.00000000e+01 6.00000000e+00
Al 2.20000000e+01 1.20000000e+01 6.00000000e+00
Al 2.00000000e+01 1.40000000e+01 6.00000000e+00
Al 2.60000000e+01 8.00000000e+00 6.00000000e+00
Al 2.40000000e+01 1.00000000e+01 6.00000000e+00
Al 3.00000000e+01 1.20000000e+01 6.00000000e+00
Al 2.80000000e+01 1.40000000e+01 6.00000000e+00
Al 1.00000000e+00 1.00000000e+00 1.00000000e+00
Al 5.00000000e+00 5.00000000e+00 1.00000000e+00
Al 9.00000000e+00 1.00000000e+00 3.00000000e+00
Al 1.30000000e+01 5.00000000e+00 3.00000000e+00
Al 1.00000000e+00 9.00000000e+00 1.00000000e+00
Al 5.00000000e+00 1.30000000e+01 1.00000000e+00
Al 9.00000000e+00 9.00000000e+00 3.00000000e+00
Al 1.30000000e+01 1.30000000e+01 3.00000000e+00
Al 1.70000000e+01 1.00000000e+00 3.00000000e+00
Al 2.10000000e+01 5.00000000e+00 3.00000000e+00
Al 2.50000000e+01 1.00000000e+00 1.00000000e+00
Al 2.90000000e+01 5.00000000e+00 1.00000000e+00
Al 1.70000000e+01 9.00000000e+00 3.00000000e+00
Al 2.10000000e+01 1.30000000e+01 3.00000000e+00
Al 2.50000000e+01 9.00000000e+00 1.00000000e+00
Al 2.90000000e+01 1.30000000e+01 1.00000000e+00
Al 3.00000000e+00 7.00000000e+00 3.00000000e+00
Al 7.00000000e+00 3.00000000e+00 3.00000000e+00
Al 1.10000000e+01 7.00000000e+00 1.00000000e+00
Al 1.50000000e+01 3.00000000e+00 1.00000000e+00
Al 3.00000000e+00 1.50000000e+01 3.00000000e+00
Al 7.00000000e+00 1.10000000e+01 3.00000000e+00
Al 1.10000000e+01 1.50000000e+01 1.00000000e+00
Al 1.50000000e+01 1.10000000e+01 1.00000000e+00
Al 1.90000000e+01 7.00000000e+00 1.00000000e+00
Al 2.30000000e+01 3.00000000e+00 3.00000000e+00
Al 2.70000000e+01 7.00000000e+00 3.00000000e+00
Al 3.10000000e+01 3.00000000e+00 3.00000000e+00
Al 1.90000000e+01 1.50000000e+01 1.00000000e+00

Al 2.30000000e+01 1.10000000e+01 3.00000000e+00
Al 2.70000000e+01 1.50000000e+01 3.00000000e+00
Al 3.10000000e+01 1.10000000e+01 3.00000000e+00
Al 5.00000000e+00 1.00000000e+00 5.00000000e+00
Al 1.00000000e+00 5.00000000e+00 5.00000000e+00
Al 1.30000000e+01 1.00000000e+00 7.00000000e+00
Al 9.00000000e+00 5.00000000e+00 7.00000000e+00
Al 5.00000000e+00 9.00000000e+00 5.00000000e+00
Al 1.00000000e+00 1.30000000e+01 5.00000000e+00
Al 1.30000000e+01 9.00000000e+00 7.00000000e+00
Al 9.00000000e+00 1.30000000e+01 7.00000000e+00
Al 2.10000000e+01 1.00000000e+00 7.00000000e+00
Al 1.70000000e+01 5.00000000e+00 7.00000000e+00
Al 2.90000000e+01 1.00000000e+00 5.00000000e+00
Al 2.50000000e+01 5.00000000e+00 5.00000000e+00
Al 2.10000000e+01 9.00000000e+00 7.00000000e+00
Al 1.70000000e+01 1.30000000e+01 7.00000000e+00
Al 2.90000000e+01 9.00000000e+00 5.00000000e+00
Al 2.50000000e+01 1.30000000e+01 5.00000000e+00
Al 7.00000000e+00 7.00000000e+00 7.00000000e+00
Al 3.00000000e+00 3.00000000e+00 7.00000000e+00
Al 1.50000000e+01 7.00000000e+00 5.00000000e+00
Al 1.10000000e+01 3.00000000e+00 5.00000000e+00
Al 7.00000000e+00 1.50000000e+01 7.00000000e+00
Al 3.00000000e+00 1.10000000e+01 7.00000000e+00
Al 1.50000000e+01 1.50000000e+01 5.00000000e+00
Al 1.10000000e+01 1.10000000e+01 5.00000000e+00
Al 2.30000000e+01 7.00000000e+00 7.00000000e+00
Al 1.90000000e+01 3.00000000e+00 5.00000000e+00
Al 3.10000000e+01 7.00000000e+00 7.00000000e+00
Al 2.70000000e+01 3.00000000e+00 7.00000000e+00
Al 2.30000000e+01 1.50000000e+01 7.00000000e+00
Al 1.90000000e+01 1.10000000e+01 5.00000000e+00
Al 3.10000000e+01 1.50000000e+01 7.00000000e+00
Al 2.70000000e+01 1.10000000e+01 7.00000000e+00

List of Symbols, Abbreviations, and Acronyms

γ	gamma
2D	two-dimensional
3D	three-dimensional
DFT	density functional theory
LAMMPS	Large-scale Atomic/Molecular Massively Parallel Simulator
PBC	periodic boundary condition
SAED	selected area electron diffraction
XRD	X-ray diffraction

1 DEFENSE TECHNICAL
(PDF) INFORMATION CTR
DTIC OCA

1 DEVCOM ARL
(PDF) FCDD RLB CI
TECH LIB

1 DEVCOM ARL
(PDF) FCDD RLA WA
W MATTSON