Chapter 3

Methodology

3.1 Introduction

All simulations carried out in this work are based on the classical Born model of ionic solids [99] in which the lattice is constructed from a periodic array of charged spherical ions. The interaction between ions is partitioned into two components. The first is a long range electrostatic interaction (Φ_{LR}) and the second is a short range interaction (Φ_{SR}) that acts mainly as a repulsive force that prevents two oppositely charged ions from colliding to form a singularity but it may also contain an attractive component. These combine as per Equation 3.1 to contribute to the total energy (Φ). Figure 3.1 shows a graphical representation of the short range, long range and total interaction energy between a pair of ions oppositely charged ions.

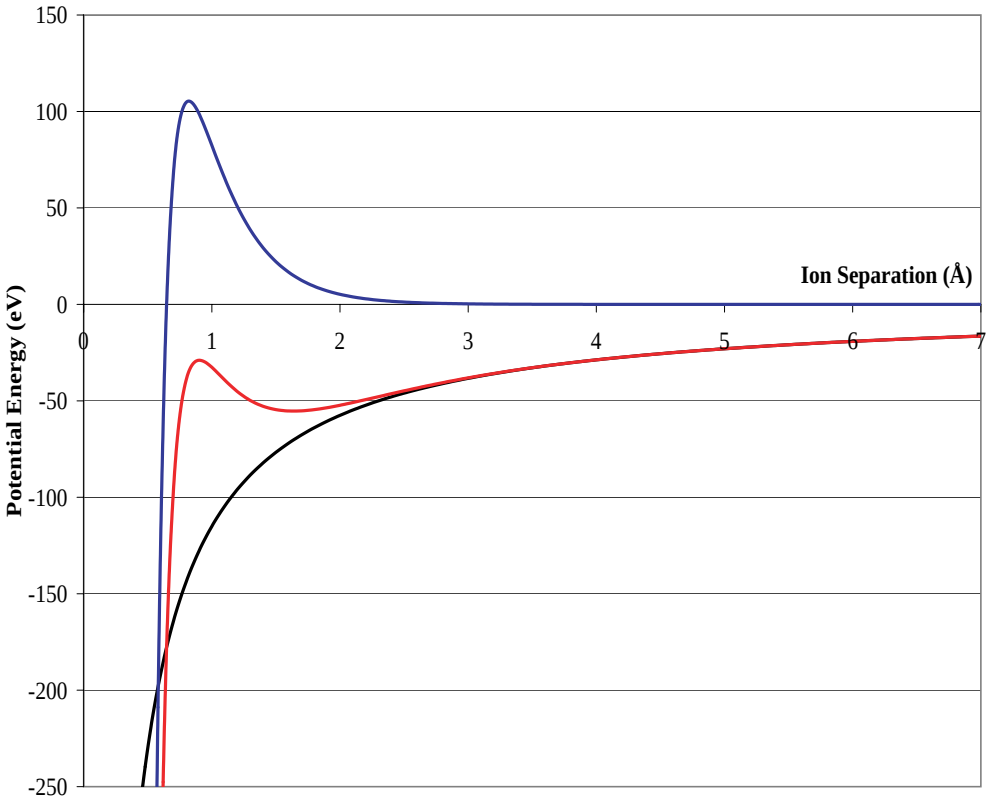

Figure 3.1: Graph of the interaction energy between two oppositely charged ions showing the contribution from the short range interaction (Blue), the long range Coulombic interaction (Black) and the total interaction (Red).

$$\Phi = \Phi_{SR} + \Phi_{LR} \tag{3.1}$$

The general interaction between all the ions in a solid containing many ions can be very complicated but it can be written simply as a series of summations involving increasing numbers of ions as per equation 3.2 i.e. two-body, three-body and n-body terms.

$$\Phi = \Phi_0 + \sum_{ij} \Phi_{ij} + \sum_{ijk} \Phi_{ijk} + \sum_{ijkl} \Phi_{ijkl} + \dots \tag{3.2}$$

where ij represents interactions between pairs of ions, ijk triplets, and higher order terms. Φ_0 is a function of the local environment and defines the zero energy. It is generally ignored as differences in energy are usually the important factor.

In an ionic solid (including all of the compositions studied) the interaction between ion pairs dominate and all the other terms can be omitted. This is known as the pair-wise approximation and is used throughout this work.

3.2 Atomic interactions Using Pair Potentials

The Long Range Interaction

By simplifying all the interactions to include only the pairwise, the long range interaction can be defined relatively simply. As mentioned earlier, it is essentially the electrostatic interaction between the ion pair and as such can either be attractive, in the case of oppositely charged ions, or repulsive if they are of like charge. Equation 3.3 shows the long range interaction energy(Φ_{LR}) of two ions (ion i and ion j) with charges q_i and q_j at a separation of r_{ij} (ε_0 is the permittivity of free space).

$$\Phi_{LR}(r_{ij}) = \frac{q_i q_j}{4\pi\varepsilon_0 r_{ij}} \tag{3.3}$$

While the definition of the long range interaction is simple, calculating it explicitly would be computationally expensive due to the slow rate at which contributions to the interaction energy fall off with distance $\left(i.e.\frac{1}{r_{ij}}\right)$. This means that for each ion, a huge number of interactions must be considered in order to compute this interaction with any accuracy. Fortunately, Ewald [100] developed an approximate method of calculating this summation which involves splitting the calculation into a short range real space component and a long range reciprocal space component [101]. The mathematics behind this are quite in depth and beyond the scope of this thesis. The result is that this allows the interaction to be calculated much more rapidly. This method is implemented in both molecular statics codes used during the course of this work (CASCADE [102] and GULP [103]). A simplified explanation of this method is given in Appendix B of "Introduction to Solid State Physics" by Charles Kittel [101].

3.2.1 The Short Range Interaction

Whereas the idea behind the long range interaction is very simple to understand, the origins of the short range interaction are more complicated. It is important to understand where this term originates from and how it is described as it is this contribution to the interaction that has the greatest effect on the accuracy of the simulations run in this study.

The short range interaction can be either completely repulsive or have a combination of repulsive and attractive components dependent on the ions that are interacting. The repulsive interaction originates from the overlap

of the electron clouds of both atoms at very small ionic separations. This is partly due to the Pauli exclusion principal [104], with the rest of the contribution due to nuclear-nuclear interactions. The generalised version of the Pauli exclusion principal states that no two fermions can share the same quantum state. When the electron clouds overlap the Pauli exclusion principal forces the ground state charge distribution of the electrons to have a higher energy. This increase in energy is manifested as a repulsive force that increases in strength as the ions are moved closer together.

At slightly larger separations, an overall small attractive force can exist which is generally known as the van der Waals - London interaction. This attraction is due to the spontaneous formation of instantaneous dipoles on each of the interacting ions. London was able to determine a general expression for this interaction by describing dipole formation as a correlated motion of the electrons in both atoms [105–107]. London found that for the case of pair of identical ions i and j, this force varies as a function of r_{ij}^{-6} where r_{ij} is the separation of the ions. Clearly this force will be stronger between polarisable ions.

The short range interaction is clearly complex and consequently it is possible to approximate it using a number of functional forms. The potentials used in this study and some of the other possible approximations will now be discussed.

Potential Forms

The earliest attempt to couple a short range repulsive potential with the Long range Coulomb interaction was attempted by Born and Landé [108] and the short range component was described by the following equation.

$$\Phi_{ij} = \frac{b}{r_{ij}^{n}} \tag{3.4}$$

where b and n are variables chosen to reproduce the equilibrium interionic separation; early work took n = 9. This model was later expanded when quantum mechanical calculations proved that, while it was a useful approximation, it was not correct for all materials. With this in mind, Born and Mayer [109] developed a short range potential of the form:

$$\Phi_{ij} = Ae^{-\frac{r_{ij}}{\rho}} \tag{3.5}$$

where A and ρ are variable parameters. So far, none of the potentials have considered the van der Waals interaction. The Lennard-Jones potential [110] combines the repulsive form from Equation 3.4 with an attractive van der Waals potential using the r_{ij}^{-6} dependence determined by London. The strength of this effect can be adjusted by varying the parameter C.

$$\Phi_{ij} = \frac{b}{r_{ij}^{n}} - \frac{C}{r_{ij}^{6}} \tag{3.6}$$

In modern calculations n is normally 12. This potential is often used to model non-bonded interactions such as those in liquids and gasses. If the

$\left(\frac{b}{r_{ij}^n}\right)$ term from the Lennard-Jones potential is removed and replaced with the more flexible two parameter exponential version from the Born-Mayer potential (Equation 3.5) then the result is the Buckingham potential [111].

$$\Phi_{ij} = Ae^{-\frac{r_{ij}}{\rho}} - \frac{C}{r_{ij}^6} \tag{3.7}$$

where, as before, A, ρ and C are parameters which are varied in order to reproduce experimental data. This is the potential form employed for many of the interatomic potentials used in this study although, due to the obvious case when C=0, some of the potentials used are of the Born-Mayer type. It should be noted that the short range interaction shown in figure 3.1 used the Buckingham potential model. Tables listing the Buckingham potential parameters used in this study are given in the respective chapters to avoid the confusion of which potentials are used in which circumstance.

Due to the short range nature of these potentials, a cut-off value beyond which they are no longer evaluated is used. The purpose of this is to reduce computation times since beyond a few lattice spacings, the contribution to the interaction is negligible (see figure 3.1). The value of this cut-off is normally determined by running a series of identical calculations with increasing values of the short range cut-off and plotting this against lattice energy. When this value reaches a plateau, further increases to the short range cut-off do not contribute to the energy and thus the cut-off value is chosen.

3.2.2 Derivation of Short Range Potentials

The potentials in chapter 4 were modified from Minervini *et al.* [112] in order to more accurately reproduce the lattice parameters of the pyrochlore and the δ-phase (see chapter 4). This was performed by a purely empirical approach.

Empirical potential fitting is an iterative process whereby the potential parameters are varied in order to minimise the discrepancies between the simulation result and experimental data. In this study, only the parameters for chapter 4 were newly derived, and each potential was fitted to its basic oxide and any pyrochlore or δ-phase compound for which experimental data was available. In this system efforts were made to make sure that the sum of the deviations from the experimental lattice parameter was less than 1%.

3.3 Ionic Polarisability

The response of an ion's electric charge density to an electric field is simulated via the shell model of Dick and Overhauser [113] (note: only for those systems where it is required and feasible to implement i.e. not for molecular dynamics). The shell model is an addition to the Born model of the lattice that provides a method for predicting the effects of ionic polarisation. The addition of shells to ions in the lattice increases the computational time and memory requirements for the simulation as it adds many-body terms to the calculation. This means that it is not reasonable (and nor is it required) to add shells to all species. In order to decide whether or not an ion requires

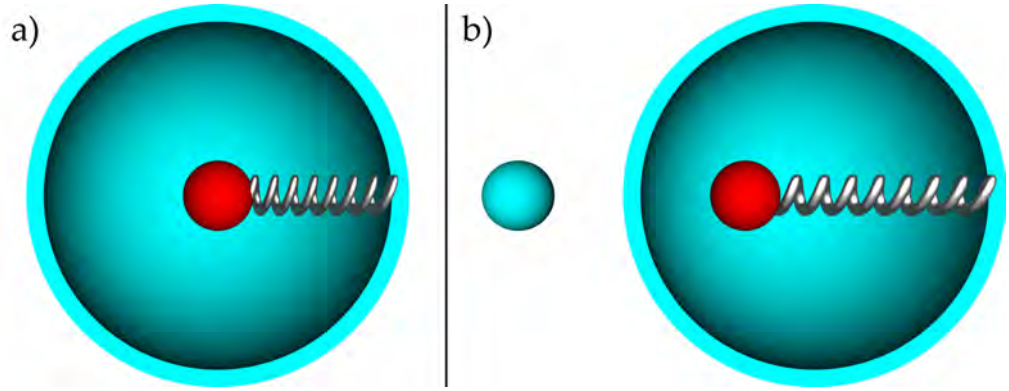

Figure 3.2: Graphical representation of the shell model showing a) no displacement, unpolarised b) displacement of the core from the centre, polarisation. In both figures, the red sphere represents a core with charge $+Xe$, the surrounding hemisphere represents the massless shell with charge $-Ye$ and the spring is representative of the isotropic force constant k. In b) the light blue sphere represents a negatively charged ion.

a shell, the electronic polarisability on the ion must be taken into account. Generally it is found that shells are required for anions, $O^{2=}$ for example, but not necessarily for cations, e.g. Mo^{4+}.

The Dick and Overhauser shell model for ionic polarisability [113] is used throughout. This model describes an ion as having a central core with charge Xe, representing the nucleus, and a shell of charge Ye, representing the electrons. These two charges are balanced so that the sum of $(X + Y)\, e$ is the same as the valence state of the ion. Generally the core is given a slightly positive charge and the shell is negative to balance this. These are coupled together via a harmonic force constant k (see figure 3.2). This allows the shell to move with respect to the core, thus simulating a dielectric polarisation. Using this description, the polarisability of an isolated ion, α, is given by equation 3.8:

$$\alpha = \frac{1}{4\pi\varepsilon_0}\left(\frac{Y^2}{k}\right) \tag{3.8}$$

Where Y is the charge on the shell, α is the polarisability (Å^3), k is the harmonic force constant (eVÅ^{-2})

Although no new shell model parameters were defined during the course of this study, it is still important to understand how such parameters were developed. Shell parameters are added to reproduce polarisability and therefore are fitted against the dielectric and elastic properties of a crystal. Since the polarisability has a particularly strong influence on the high frequency dielectric constant, it is generally the main observable parameter used to fit X, Y and k.

This model allows the coupling of the short range interaction to the polarisability and in this work it is assumed that (where present) short range interactions only interact between shells. This increases the complexity of the model as the position of the cores is now affected by a many-body interaction based on the electrostatic interactions with the shells and cores within the system, the two-body interactions of the shells and the force constant k. This added complexity allows the simulation of systems with greater accuracy than via a rigid-ion model.

This implementation of the shell model is limited as it cannot reproduce materials that violate the Cauchy relationship [114]. This relationship indicates that (for cubic crystals) the elastic constants c_{12} and c_{44} are equal. Unfortunately real materials can violate this and these two values can be quite different. This can be fixed to allow for Cauchy violations where $c_{12} < c_{44}$ by assigning a real radius to the shell and allowing it to change depending on the conditions, this model is termed the breathing shell [115]. Further modifications are also possible, for example, the elliptical breathing shell model [116] in which the shell can distort elliptically as well as change radius. This allows

deviations of the form $c_{12} > c_{44}$. Unfortunately, these breathing shell parameters are notoriously difficult to parameterise and it has not been possible to implement here.

3.4 Static Calculations

3.4.1 Energy Minimisation

As the potential model used in the simulations is an approximation to reality, the force field predicted by the model does not exactly reproduce that determined by experiment. This means that even though the ions are placed in the ideal configuration as dictated by experiment, they may not be in the lowest energy state with respect to the approximated interionic potentials. Before a defect calculation is run or even the lattice energy is determined, it is important to allow the ions to relax to their lowest energy configuration with respect to the potentials used (otherwise any subsequent changes to ion positions may include a term associated with the movement of ions to their equilibrium positions). Obviously great care is taken to ensure that discrepancies with experimental data are kept to a minimum during the fitting process. This is done to ensure the greatest possible agreement between the simulated lattice and reality.

The simulation codes used minimise the lattice iteratively. During this process, the forces on each ion are calculated and then the ion is shifted slightly in proportion to the force acting on it. This continues until the forces acting

on all the ions are zero.

It is possible to minimise the lattice at a constant volume or a constant pressure. Under constant volume conditions, the edges of the cell are held fixed and the ions can only move by varying their locations within the cell relative to the strains experience by each ion in the cell. Under constant pressure minimisation, the unit cell dimensions are also allowed to change such that the strains on the cell are minimised as well as those for the ions within it (clearly, in general this affords a more realistic simulation). As there are fewer degrees of freedom available for constant volume calculations, they take less time and as such many of the early energy minimisation calculations used this method. Due to advances in computer technology over the years, this optimisation is no longer required and all of the energy minimisation calculations in this thesis were performed under constant pressure conditions (interestingly many quantum mechanical simulations still use the constant volume approximation).

As mentioned earlier, the aim of the minimisation process is:

$$F = \frac{\partial U_L}{\partial r} = 0 \tag{3.9}$$

where F is the force on the system, U_L is the lattice energy and r is the coordinate system.

If the lattice energy of a system of N ions with coordinates, r, is $U_L(r)$ then the lattice energy at a new set of coordinates r' is:

$$U_L(r') = U_L(r) + \vec{g^T} \cdot \vec{\delta} + \frac{1}{2}\vec{\delta^T} \cdot \mathbf{W} \cdot \vec{\delta} \tag{3.10}$$

where $\vec{\delta}$ is a generalised strain vector with $3N$ displacement components, $\vec{\delta r}$, and 6 bulk strain components, $\vec{\delta\varepsilon}$;

$$\vec{\delta} = \left(\vec{\delta r}, \vec{\delta\varepsilon}\right) \tag{3.11}$$

$\vec{g}$ corresponds to the first derivatives of the lattice energy with respect to displacement and strain;

$$\vec{g} = \left(\frac{\partial U_L}{\partial r}, \frac{\partial U_L}{\partial \varepsilon}\right) \tag{3.12}$$

and $\mathbf{W}$ is a matrix that contains the corresponding second derivatives;

$$\mathbf{W} = \begin{pmatrix} \frac{\partial^2 U_L}{\partial r \partial r} & \frac{\partial^2 U_L}{\partial r \partial \varepsilon} \\ \frac{\partial^2 U_L}{\partial \varepsilon \partial r} & \frac{\partial^2 U_L}{\partial \varepsilon \partial \varepsilon} \end{pmatrix} \tag{3.13}$$

the new coordinates are related to the old coordinates by

$$r' = \Delta\varepsilon \cdot (r + \delta r) \tag{3.14}$$

where $\Delta\varepsilon$ is the Voight matrix representation of $\vec{\partial\varepsilon}$

$$\Delta\varepsilon = \vec{\partial\varepsilon} = \begin{pmatrix} \delta\varepsilon_1 & \frac{1}{2}\delta\varepsilon_6 & \frac{1}{2}\delta\varepsilon_5 \\ \frac{1}{2}\delta\varepsilon_6 & \delta\varepsilon_2 & \frac{1}{2}\delta\varepsilon_4 \\ \frac{1}{2}\delta\varepsilon_5 & \frac{1}{2}\delta\varepsilon_4 & \delta\varepsilon_3 \end{pmatrix} \tag{3.15}$$

In the simulation codes used, the simulation iterates closer and closer to the goal of zero force on the ions until the energy change is less than a predetermined value.

3.4.2 Defect Calculations

Once the lattice energy has been minimised with respect to ion positions, defects can be introduced, for example an ion can be removed to create a vacancy. The lattice will respond to the presence of this defect by further relaxation around this defect and it is assumed that the majority of this relaxation occurs close to the defect and that the extent of relaxation decreases with distance. This makes it reasonable to split the lattice up into a series of different concentric spherical regions in which progressively more approximate methods are used to calculate the lattice response to the defect with increasing distance (figure 3.3 shows this).

The region containing and immediately surrounding the defect is called Region I, this extends from the centre of the defect site to a predetermined radius. In this region, the interactions are calculated explicitly with the ions being allowed to relax to zero strain. Beyond this boundary, the lattice relax-

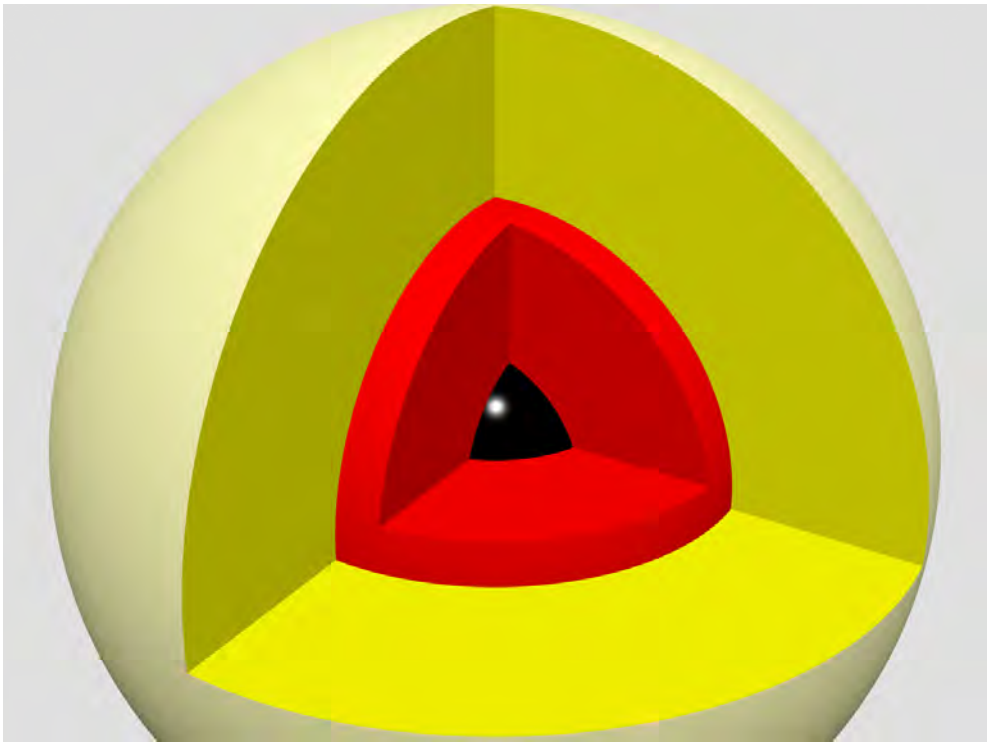

Figure 3.3: Graphical representation of the multi region approach for defect energy calculations. The defect(s) are coloured black, region I is red and region IIa is yellow. Region IIb is represented by the grey background and extends to infinity from the outside surface of region IIa (pale yellow).

ation of the defect is much smaller and as such, a more approximate method can be used. This outer region, termed region II, extends from the edge of this region to infinity.

Region II is further subdivided into two regions, IIa and IIb, where region IIa acts as a buffer layer between region I and region IIb. In region IIa, ion displacements are inferred from the Mott-Littleton approximation [117] but interactions with region I are calculated by explicit summation. This means that region IIa must always be at least the size of region I plus the short range cut-off, such that the ions at the edge of region I can interact with the

ions in region IIa. The response of the ions in region IIa, due to a defect at a distance, r, is due to the polarisation, P, calculated using the Mott-Littleton approximation [117]. For large distances

$$P = \frac{q\vec{r}}{4\pi r^3}\left(1 - \frac{1}{\varepsilon}\right) \tag{3.16}$$

where q is the charge of the defect and ε is the static dielectric constant of the crystal.

The total energy introduced to the lattice via the incorporation of a defect, E_d, is the sum of the energies in each region

$$E_d = E_I(\vec{r}) + E_{I-II}(\vec{r}, \vec{\zeta}) + E_{II}(\vec{\zeta}) \tag{3.17}$$

where E_I is the energy due to the interactions of the ions in region I with their displacements given by the displacement vector $\vec{r}$; E_{II} is the energy of region II with an ion displacement vector $\vec{\zeta}$; and E_{I-II} is the energy due to the interaction between region I and II ions. It is not possible to calculate this explicitly since region II extends to infinity and thus would require the sum of an infinite number of displacements. However, since the displacements in region II are very small a quasi harmonic approximations is valid such that:

$$E_{II} = \frac{1}{2}\vec{\zeta} \cdot \mathbf{A} \cdot \vec{\zeta} \tag{3.18}$$

where $\mathbf{A}$ is the force constant matrix. Substituting equation 3.18 into 3.17 and differentiating with respect to $\vec{\zeta}$, the equilibrium displacements in region

II are given by:

$$\frac{\delta E}{\delta \zeta} = \left.\frac{\delta E_{I-II}(\vec{r},\vec{\zeta})}{\delta \vec{\zeta}}\right|_{\vec{\zeta}=\vec{\zeta_e}} + \mathbf{A}\vec{\zeta_e} \tag{3.19}$$

where $\vec{\zeta_e}$ is the equilibrium value of $\vec{\zeta}$ corresponding to $\vec{r}$. When this is substituted into equation 3.18 and then back into equation 3.17, the defect energy dependence of the energy of region II is removed as shown in Equation 3.20

$$E_d = E_I(\vec{r}) + E_{I-II}(\vec{r},\vec{\zeta}) - \frac{1}{2}\left.\frac{\delta E_{I-II}(\vec{r},\vec{\zeta})}{\delta \vec{\zeta}}\right|_{\vec{\zeta}=\vec{\zeta_e}} \cdot \vec{\zeta_e}\vec{\zeta_e} \tag{3.20}$$

3.5 Molecular Dynamics Calculations

3.5.1 Background

Molecular dynamics simulations are fundamentally different from the energy minimisation process discussed previously. Energy minimisation processes relax the system to a local energy minimum with respect to the interatomic potentials used. While valuable, it cannot give any information about how such a system can vary as a function of time. This is where molecular dynamics (MD) becomes useful.

MD is a computer simulation technique where a set of interacting ions is allowed to evolve in time by integrating the laws of motion. In a MD simulation containing N ions, each ion, i, is assumed to obey Newton's laws of motion and, most importantly, Newton's first law such that:

$$F_i = m_i \mathbf{a}_i \tag{3.21}$$

where F_i is the force act on each individual ion due to interactions with all the other ions, m_i is the mass of that ion and $\mathbf{a}_i$ is equal to $\frac{d^2\mathbf{r}_i}{dt^2}$ (the acceleration of that ion).

There are many similarities between MD and the static calculations described previously in that the forces between the ions are based on a pair potential approximation where the interaction energy, Φ, for any two ions, i and j, with positions, $\mathbf{r}_i$ and $\mathbf{r}_j$, can be expressed in the form:

$$\Phi(\mathbf{r}) = \sum_j \sum_{j>i} \phi(|\mathbf{r}_i - \mathbf{r}_j|) \tag{3.22}$$

where ϕ is a combination of long range and short range interactions as per Φ in Equation 3.1 and the $j > i$ term in the second summation means each interaction is only considered once.

3.5.2 Time and System Size Limits

The size of the simulation is limited by the amount of memory available to the computer and while technically there is no limit to the amount of time that can be simulated, a practical limit is imposed by the short time-steps required for an accurate simulation (on the order of 10^{-15} s). Simulations of metal systems (where only 1 or 2 types of atom exist and only very short range forces need to be considered) have been made in which of millions of atoms have been simulated for nanoseconds. Simulations of ceramic systems where the short range interaction acts over 5-10 Å and the even longer range Coulombic interaction needs to be considered, make it necessary to limit

the system sizes in order to simulate any meaningful period of time. The simulations run during the course of this work consist of individual runs containing about 98000 ions that were simulated for 6 ps.

3.5.3 Potentials Models for Molecular Dynamics Simulations

Short Range Interactions

While the Buckingham potential form works well for simulations reasonably near equilibrium conditions, it encounters problems with simulations where ions are likely to reach very small separations, such as simulations of collision cascade events. This is due to the fact that at very small separations, the Buckingham potential (with a non zero C-term) tends towards $-\infty$ (see figure 3.4). Even for Born-Mayer type potentials, the short range interaction is overly attractive at short range and therefore at very small separations a screened Coulomb potential (in this case the Ziegler-Biersack-Littmark or ZBL potential) was used [118]. In order for there to be a smooth transition between the short range potential and the ZBL potential, a spline was used to connect these such that the first and second derivatives of both potentials

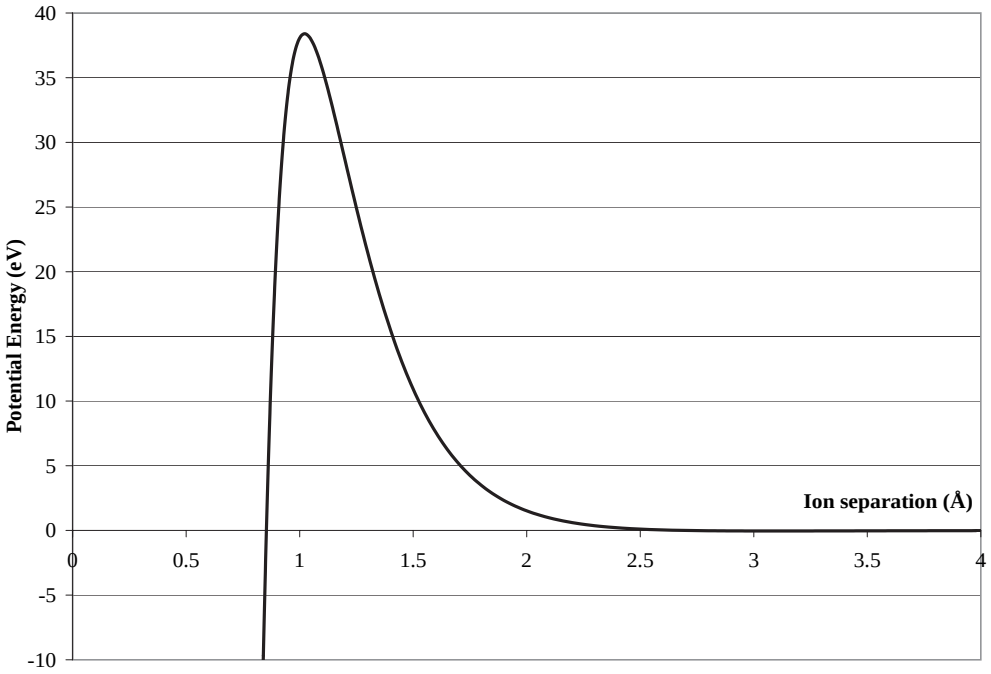

Figure 3.4: Graph showing the tendency for the Buckingham potential to tend to $-\infty$ at small ion separations when using a non-zero C-term.

were preserved. The general breakdown of the interactions is:

$$\Phi(r) = \begin{cases} \text{ZBL} & r < r_0; \\ g(r) & r_0 \leqslant r < r_1; \\ Ae^{-\frac{r}{\rho}} - \frac{C}{r^6} + V(r) & r_1 \leqslant r < r_2; \\ V(r) & r \geqslant r_2. \end{cases} \tag{3.23}$$

where V(r) is the Coulombic interaction and $g(r)$ is the splining function which is given by

$$g(r) = e^{\left(f_1 + f_2 r + f_3 r^2 + f_4^3 r + f_5 r^4\right)} \tag{3.24}$$

It should be noted that for cation-cation interactions no short range Buck-

ingham potential is used, therefore the short range term in equation 3.23 reduces to just the repulsive Coulomb interaction between r_1 and r_2 and this is then splined to the ZBL potential.

Table 3.1 shows the parameters used to spline the short range interaction to the ZBL potential, table 3.2 gives the radial cut-offs for all the species in the system. The Buckingham potential parameters used in the MD simulations will be reorted in chapter 6. These parameters were previously used to study collision cascades in spinel [119].

Table 3.1: The parameters for the spline to the ZBL interaction.

	f_1	f_2	f_3	f_4	f_5	f_6
Mg-Mg	14.4507	-47.7977	122.4210	-175.9999	125.1304	-34.1480
Mg-Al	10.6531	-17.0984	27.9501	-36.8467	28.3526	-8.5504
Mg-O	11.0765	-31.3878	118.9207	-302.6137	367.5647	-167.4741
Al-Al	10.3578	-112.8183	9.4511	-1.2318	-1.0544	0.1604
Al-O	11.0284	-28.5714	95.3424	-217.4885	242.7988	-107.0122
O-O	9.9306	-17.4669	25.3341	-19.0023	6.6210	-0.8289

Table 3.2: Cut-off parameters used in MD simulations of pure and Al doped MgO (r_2 was 7.2Å for all interactions).

	r_0 (Å)	r_1 (Å)
Mg-Mg	0.5	1.05
Mg-Al	0.5	1.05
Mg-O	0.15	0.80
Al-Al	0.3	1.05
Al-O	0.15	0.8
O-O	0.2	1.05

Long Range Interactions

While many MD simulation packages use the Ewald sum (like the energy minimisation packages mentioned earlier) to calculate the long range Coulombic interactions, the MD code used in this thesis employs the Distributed Parallel version of the fast Multipole Tree Algorithm (DPMTA) library by Rankin [120]. This method uses multipole expansions to reduce the scaling from a system containing N particles from N^2 to $N \log N$, or for special situations, N [120]. A mathematical breakdown of how this method works will not be given here as it is beyond the scope of this thesis and detailed information can be found in [120] and [121]. A simple but brief introduction into the workings of the fast multipole algorithm is also given in [122].

Multipole expansions take advantage of the fact that at sufficiently great distances a group of charged particles can be represented as a single multipole expansion. As shown in figure 3.5, the many individual interactions between a single distant particle with all the particles in a group may be represented by a single interaction of the distant particle with the multipole expansion for the entire group.

Even though this method is only exact in a system containing an infinite number of particles, it is known to converge quite rapidly [120]. This allows these expansions to be truncated at a relatively small size whilst still maintaining a small error. Interactions between ions with small separations are calculated directly and it is for large separations that multipole expansions replace the direct calculation.

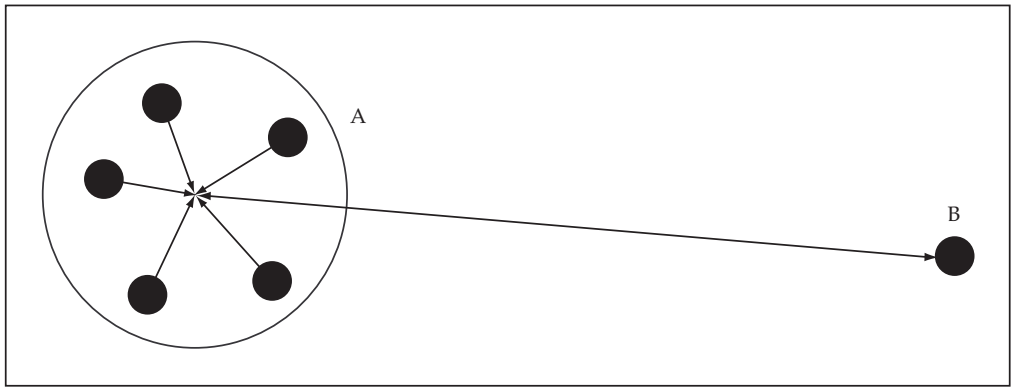

Figure 3.5: A multipole representation of a group of particles (A) interacting with a distant particle (B). Reproduced from [120].

3.5.4 Time Integration Algorithm

The core of any MD code is the time integration algorithm. This is required to integrate the equations of motion for the interacting ions and follow their paths. The solutions to this algorithm are calculated using finite difference methods in which time exists in discrete increments. By knowing the positions and some of the derivatives (see equations 3.25 and 3.26) at time, t, it is possible to integrate these to calculate the values at a time $t + \Delta t$. By making Δt small enough, the error is reduced, but never completely removed (the Verlet algorithm, for example, has an error associated with the timestep proportional to Δt^4). Another source of error is due to the rounding error associated with the finite accuracy of the processor. In this work, 64 bit representations of the properties are used in order to minimise these errors. In the current implementation of the simulation code, the timestep is a variable and scales with respect to the fastest moving ion in the system. This allows a much smaller timestep (sometimes as low as 0.08 fs) to be used at the start of the simulation so that accuracy is maintained when a few ions

have very high kinetic energies (peaking at 1 or 2 keV), whilst allowing the timestep to grow (to a maximum of 1.05 fs) once the ions in the system slow down. Consequently the simulation completes faster and time is not wasted calculating more timesteps than necessary.

Verlet Algorithm

The Verlet algorithm [123] is one of the most common integration algorithms used for MD simulations. The basic concept is to construct two, third order Taylor series expansions for the positions, $\mathbf{r}(t)$, one forward and one backwards in time. Calling the velocity $\mathbf{v}$, the acceleration $\mathbf{a}$, and the third derivatives of $\mathbf{r}$ with respect to t, $\mathbf{b}$ gives:

$$\mathbf{r}(t+\Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{1}{2}\mathbf{a}(t)\Delta t^2 + \frac{1}{6}\mathbf{b}(t)\Delta t^3 + O(\Delta t^4) \tag{3.25}$$

$$\mathbf{r}(t-\Delta t) = \mathbf{r}(t) - \mathbf{v}(t)\Delta t + \frac{1}{2}\mathbf{a}(t)\Delta t^2 - \frac{1}{6}\mathbf{b}(t)\Delta t^3 + O(\Delta t^4) \tag{3.26}$$

These equations are then added together and rearranged to yield:

$$\mathbf{r}(t+\Delta t) = 2\mathbf{r}(t) - \mathbf{r}(t-\Delta t) + \mathbf{a}(t)\Delta t^2 + O(\Delta t^4) \tag{3.27}$$

which is the basic form of the Verlet equation. Since Newtonian mechanics are assumed, $\mathbf{a}(t)$ is just the force divided by the mass and the force is a function of the ion positions $\mathbf{r}(t)$:

$$\mathbf{a}(t) = -\left(\frac{\nabla\Psi(\mathbf{r}(t))}{m}\right) \tag{3.28}$$

The popularity of this algorithm is due to the simplicity of implementation, its accuracy and stability. One of the problems is that it is often necessary to generate ion velocities, this information is often subsequently inferred in order to calculate the kinetic energy of the ions. In fact, this is commonly used to test that conservation of energy is being maintained as this is a good indication that the step size is small enough.

While it is possible to calculate the velocities using

$$\mathbf{v}(t) = \frac{\mathbf{r}(t+\Delta t) - \mathbf{r}(t-\Delta t)}{2\Delta t} \tag{3.29}$$

the errors as a function of step size are significantly larger than those for positions (proportional to Δt^2 rather than Δt^4). In order to deal with this, a better implementation of this algorithm has been developed which is used in this work and is termed the velocity-Verlet algorithm. In this method, the positions, accelerations and velocities are calculated at time $t + \Delta t$ using the following equations:

$$\mathbf{r}(t+\Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{1}{2}\mathbf{a}(t)\Delta t^2 \tag{3.30}$$

$$\mathbf{v}\left(t + \frac{\Delta t}{2}\right) = \mathbf{v}(t) + \frac{1}{2}\mathbf{a}(t)\Delta t \tag{3.31}$$

$$\mathbf{a}(t) = -\left(\frac{\nabla\Psi(\mathbf{r}(t+\Delta t))}{m}\right) \tag{3.32}$$

$$\mathbf{v}\left(t+\Delta t\right) = \mathbf{v}\left(t + \frac{\Delta t}{2}\right) + \frac{1}{2}\mathbf{a}(t+\Delta t)\Delta t \tag{3.33}$$

3.6 Simulation Codes Used

Three simulation codes were used in total, two static energy minimisation codes and one molecular dynamics code. The energy minimisation codes were; CASCADE (Cray Automated System for the CAlculation of Defect Energies) [102] and GULP (General Utility Lattice Program) [103]. CASCADE was developed at the Daresbury Laboratory for the CRAY computer and was based on HADES (Harwell Automatic Defect Examination System) code [124]. The molecular Dynamics code is called LBOMD [125] and was developed at Loughborough University.