School of Applied Chemistry

Computer Modelling of Gibbsite Crystallization

Sean David Fleming

This thesis is presented as part of the requirements for the award of the Degree of Doctor of Philosophy

at

Curtin University of Technology

July 1999

Acknowledgements

I would like to thank to my supervisors, Dr. Andrew Rohl and Prof. Gordon Parkinson, for their invaluable assistance throughout the course of this work. Dr. Rohl is especially to be congratulated for managing to juggle the demands of his first baby as well as the demands of his first Ph.D. student.

A large amount of thanks must also go to Dr. Julian Gale, who has helped me on far too many occasions to mention individually. I am also grateful to Prof. Steve Parker for his assistance with the defect studies presented in Chapter 5. Furthermore, I would like to recognize the tolerance of my colleague Mike Wilson, and also his thoughtfulness in providing the essential sugar and caffeine based foodstuffs for maintaining productivity.

And, of course, special thanks must go to my family for their amazing level of forbearance over the last few years. This work would never have been finished without their understanding. In particular, I would like to acknowledge the encouragement and support of my parents.

Finally, this project was made possible by the funding of the A.J. Parker C.R.C. for Hydrometallurgy. My appreciation goes to them for their generous financial support.

ABSTRACT

This thesis documents the development and application of a computer model for gibbsite, an aluminium tri-hydroxide polymorph. In particular, the work has emphasized the idea of computer modelling techniques combining with experimental observations to provide greater insight than either could separately. Chapter One provides an overview and introduction to the fields of solid state chemistry, crystallization and computer modelling. These ideas are extended in Chapter Two to include a more detailed discussion of the theoretical principles behind the modelling in this project. The development of transferable oxalate and hydroxide potential models, intended primarily for sodium oxalate and gibbsite, is described in Chapter Three. Both *ab initio* hypersurface fitting and lattice fitting techniques were utilized, with an average structural fitting error of under two percent. In addition, the potentials were used to successfully reproduce several (related) crystal structures, thus establishing the quality of the model. In Chapter Four, the model for gibbsite was employed in generating equilibrium and growth morphologies. The equilibrium morphology was found to give excellent agreement with experiment, with all observed faces present. However, the importance of the prismatic planes is underestimated. Also discussed in the chapter is a method for predicting the phenomenon of crystalline twinning. This technique was successfully applied to a number of systems, including gibbsite and sodium oxalate. In Chapter Five, the equilibrium morphology calculations performed earlier were extended by probing the effects of cation incorporation on the habit of gibbsite. This study was conducted in order to provide a first step in estimating the role of the crystallizing solution. Calculations of the change in surface energy caused by the replacement of a surface proton with a cation from solution were made. Different crystal habits were constructed by applying a range of defect surface coverage values to each of the faces appearing in the morphology. The resulting defect morphologies were in excellent agreement with crystal habits commonly observed by experimentalists. Also, the work provided an explanation for the

earlier underestimation of the prismatic faces. Chapter Six documents molecular simulations of solutions containing the major species known to be present in industrial and experimental Bayer liquors. The structuring in two solutions, one containing sodium hydroxide and the other potassium hydroxide, was probed by constructing graphs of the radial distribution functions. These plots indicated that a significant degree of ion pairing was occurring between the alkali metal cations (Na^+ and K^+) and the aluminate monomer ($[Al(OH)_4]^-$). Furthermore, these cations were found to be acting as 'bridges' which stabilize multiple aluminate monomers, allowing them to form clusters. This data was used to assist in explaining vibrational spectra, and to postulate that clustering may be the origin of the fine particle suspensions noted during the induction period.

Table of Symbols

K	kinetic energy
U	potential energy
G	Gibbs free energy
R	ideal gas constant
T	temperature
S	dimensionless supersaturation ratio
μ	chemical potential
γ	activity coefficient
a	activity constant
ϵ_0	dielectric constant of free space
m_e	electronic mass
e	electronic charge
$\hbar$	Planck's constant (in atomic units)
n	principal quantum number
n^*	effective principle quantum number
ξ	nuclear screening constant
ℓ	angular quantum number
m_ℓ	azimuthal quantum number
r	distance
θ	angle
p	momentum
L	Lagrangian
H	Hamiltonian
ψ	Hamiltonian eigenfunction
J	direct coulomb functional
ϵ_{xc}	exchange-correlation functional
M	Madelung constant
$\overrightarrow{D}$	dipole moment
$D(\omega)$	phonon density of states

d_{hkl}	interplanar spacing
E_{surf}	surface energy
E_{att}	attachment energy
χ	fractional surface coverage
∇	gradient operator

Contents

Chapter 1

Introduction

Crystalline materials have found countless application in today's society. Interest in crystals can be found in a wide range of fields such as medicine, industrial refining and the development of semiconductor and optical devices. The distinguishing property of these solids is the well defined patterns in which the constituent atoms and molecules are arrayed [1]. As observed by Kepler in 1611 and Hooke in 1665, it is this high degree of internal ordering that is responsible for the regular crystalline polyhedra that are observed in Nature [2][3]. These early speculators also showed that the shapes of observed crystals could be obtained by packing together spheres of identical radii. Further development on this theme was made by Haüy (1782) who explained the occurrence of various crystal faces by packing together parallelepipeds; which closely resemble the lattice or unit cells in current usage [4]. In 1912 Laue suggested that if crystals were as regular as supposed, then X-rays ought to diffract from them [5]. He then developed an atomic diffraction theory, which was experimentally verified by Friedrich and Knipping [6]. This important finding quickly led to the first crystal structure determination (of NaCl) which was performed by W.H. and W.L. Bragg in 1913 [7].

Today, interest in crystals falls broadly into one of three different categories. Firstly, work focused on the characteristics and properties of the crystalline material generally falls within the scope of the material sciences. Secondly, the determination and classification of the internal structure of a crystal may be designated as crystallography. Finally, the production of the crystalline material and how its external properties, such as size and shape, may be affected under different applied conditions may be labelled as crystal growth or, more broadly, crystallization. In practice, these fields often overlap. For example, interest in crystalline properties is often directed towards the manufacture of crystals that

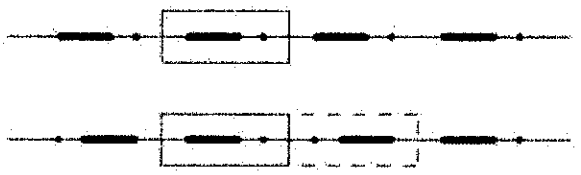

Figure 1.1: The two possibilities for pattern repetition (motif enclosed in solid box) in one dimensional space.

conform to certain standards (*e.g.* purity, size, shape). Conversely, in the field of crystal production, a knowledge of the ordered properties of the crystal is required to understand and model the mechanisms of growth. The latter area is the primary focus of this work.

1.1 Crystal Science

1.1.1 Crystallography

In general, crystallography is concerned with how a motif is repeated throughout space. For the case of a purely linear motif, there are two unique ways to fill one dimensional space (Figure 1.1). The first method involves the use of simple translations, while the second method employs a reflection operation. Since the uppermost pattern in Figure 1.1 has been produced using translation only, the motif enclosed in the solid rectangle is also called a (one dimensional) unit cell. However, the unit cell for the lower pattern needs to be twice as large to include the original motif (solid rectangle) and also one reflection (dashed area). This combination is necessary so that the application of only translation operations to the unit cell will fill all one dimensional space. Since the latter unit cell is comprised of two motifs that are symmetry related, it is possible to characterize a unit cell with details of the unique motif, plus any internal symmetry operations required to create a full unit cell. A cell that has been symmetry reduced in this fashion is known as the asymmetric unit.

Proceeding to two and three dimensional patterns, it can be shown that there is a maximum of 17 and 230 unique ways to fill their respective spaces [8]. The derivation of the 230 possible three dimensional patterns was independently conducted by Fedorov, Barlow and Schoenflies in the late nineteenth century. Schoen-

flies, who employed group theory, is responsible for usage of the term 'space group' to describe these patterns [4]. However, due to the complexity of the three dimensional case, it is often convenient to create an analogy between the three dimensional ordering in crystals and the patterns that may be drawn on a flat sheet of paper, via the projection from a higher dimension approach [9]. Two dimensional patterns or tilings are useful, as they are often easier to visualize (and much easier to draw) than their three dimensional counterparts.

One of the basic rules of crystallography requires that translational repetition of a crystal's unit cell must fill all space. This may be regarded as a stability issue, as repeated gaps in unit cell packing are inherently unstable and unlikely to appear in Nature. This does not mean that there are no holes in the matter that comprises a crystal, only that copies of the unit cell are required to pack perfectly together in a regular fashion. An apparent exception to this rule was the discovery of quasi-crystalline material by Shechtman *et al.* [10] which exhibited icosahedral symmetry, yet gave the characteristic sharp diffraction peaks of a crystal. This symmetry is forbidden in true crystals as it is impossible to completely tile a flat surface with non-interleaving pentagons. Initially, there was a considerable degree of debate over the explanation of this phenomenon. Some researchers suggested that quasi-crystals possessed multiple unit cells that are semi-disordered at short range, but ordered at longer range. This is is usually represented in the form of Penrose tiling [6][11]. In contrast, confusion arose when certain crystals, apparently exhibiting this forbidden symmetry, were explained in terms of crystalline twinning [12][13]. However, this is a microcrystalline state, and is distinguishable from the true quasi-crystalline material [14].

It is conventional to use a unit cell that has straight edges and flat terminating surfaces. This does not remove any generality, and simplifies matters enormously. With this in mind, a set of constraints to which a unit cell must conform may readily be postulated. Consider an infinite block of homogeneous matter which is to be divided into identical finite units that can be packed together *ad infinitum* to reproduce the original block of matter. If a series of parallel cuts is applied, all of which are separated by a distance a (cut1), we obtain slices of material that are infinite in two dimensions. If a different series of equally spaced (separation b) parallel cuts are applied (cut 2), which are not parallel to the first, then fingers of material are created that are infinite in one dimension. Finally, if a third series of parallel cuts (cut 3) are applied, each separated by c and not parallel to cut 1 or cut 2, the goal of finite elements that reproduce the whole has been achieved (Figure 1.2). These unit cells are parallelepipeds with unique side lengths of (a,

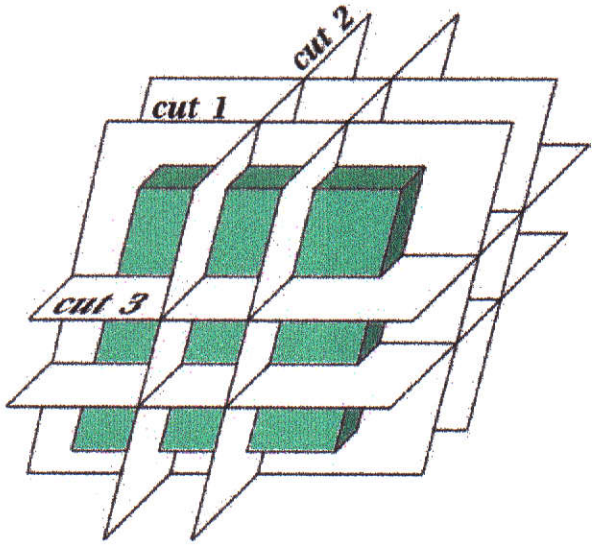

Figure 1.2: Decomposition of a block of homogeneous matter using parallel planar cuts.

b, c) and angles given by (α, β, γ) as shown in Figure 1.3. For a block of matter, built from such units, the points at which the vertices of adjacent blocks touch are known as lattice points. A system that possesses only one such lattice point per building block is termed a primitive lattice. In the illustration, the lattice points are not unique and could, for example, be placed at the center of each fundamental building block. In addition, taking only representative combinations of (a, b, c) and (α, β, γ), there are a total of seven unique crystal classes. While each class contains a single primitive or P-type lattice [15], some possess additional (non-primitive) lattice types as shown in Table 1.1. The letters used to indicate the type of non-primitive lattice are: I = body centered, F = face centered and C = single face centered. Note that the latter centering type implies that the c axis is not in the plane of the centered face. Thus, A and B centering are alternate possible notations that also appear in the literature.

In the previous discussion of lattices, it has been assumed that matter is homogeneous, and lacking in any internal order. This is of course unrealistic, as crystals have atoms and molecules located at or around the lattice points. In the simplest possible case, a crystal may have lattice points with only a single atom located at each, such as in metallic lattices. This internal order has some consequences on the choice of a, b and c. By convention, the smallest possible lengths for the three axes are chosen. However, many materials have multiple

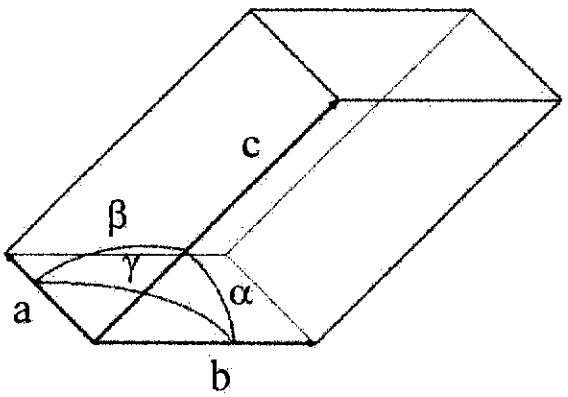

Figure 1.3: Unit cell with side lengths a, b and c and angles α, β, γ.

Crystal class	Lattice types (centering)	Unit cell restrictions
Triclinic	Triclinic parallelepiped (P)	$a \neq b \neq c$ $\alpha \neq \beta \neq \gamma$
Monoclinic	Monoclinic parallelepiped (P) Clinorhombic prism (C)	$a \neq b \neq c$ $\alpha = \gamma = 90° \neq \beta$
Orthorombic	Rectangular prism (P) Face-centered rectangular prism (F) Rhombic prism (C) Body centered prism (I)	$a \neq b \neq c$ $\alpha = \beta = \gamma = 90°$
Tetragonal	Square prism (P) Body centered square prism (I)	$a = b \neq c$ $\alpha = \beta = \gamma = 90°$
Cubic	Cubic (P) Body centered cubic (I) Face centered cubic (F)	$a = b = c$ $\alpha = \beta = \gamma = 90°$
Trigonal	Rhombohedron (P)	$a = b = c$ $\alpha = \beta = \gamma \neq 90°, < 120°$
Hexagonal	Hexagonal prism (P)	$a = b \neq c$ $\alpha = \beta = 90°$, $\gamma = 120°$

Table 1.1: The 14 Bravais lattices types. For unit cell lengths, the usage of $\neq$ means only that equality is not required by symmetry.

arrangements of atoms present, which may or may not possess some additional symmetry. Classification is achieved by sorting this internal symmetry into one of 32 possible point groups. The name point group derives from fact that they produce all space (i.e. all possible packing arrangements) about a lattice point, leaving the point unmoved [16]. Point groups are comprised of a set of symmetry operations that may include the following (or any combination thereof): rotations, reflections and inversions. When applied to lattices, translational operations must also be considered. Thus, combinations of these operators, such as a translation and reflection (glide plane) and a translation and rotation (screw axis) also occur. Combining the 14 lattices (arrangements of lattice points) with the 32 point groups (arrangement about a lattice point) we obtain 230 unique possibilities, or space groups, to which any crystal may be designated.

Morphology

In Nature, crystals are obviously not infinite, but are terminated by surfaces. The shape of single crystals may be broadly classified according to the dimension of their surfaces. Those crystals with a surface dimension of two are polyhedra, being terminated by macroscopically flat surfaces. Such surfaces are usually classified using integer multiples (h, k, l) of the inverses of their intercepts on the three crystallographic axes. These are the Miller indices, which describe a family of parallel planes as shown in Figure 1.4. These planes are equally spaced, with the separation denoted as d_{hkl}. Crystals with a non-integer surface dimension are termed dendrites [17], which include bifurcated structures such as snow flakes. These forms are also known as fractals.

Morphology prediction is concerned with the shapes a given crystal may take, and the factors influencing such phenomena. In this work, the focus is on predictions for polyhedra only. Around the mid-1600's it was observed that identical substances sometimes formed crystals of differing size and shape. However, Steno [18] observed that in these cases the angles between adjacent pairs of faces remained the same. This led Haüy to remark that crystal faces could be referenced with rational indices [19]. However, the biggest advancement in morphology prediction came when Bravais noted that the importance of a face was often proportional to its interplanar spacing d_{hkl}. This enables the determination of a crystal's shape from information about its unit cell only. In practical terms, this approach is based on the idea that faces with the largest interplanar spacing grow the slowest, and thereby dominate the morphology. Later modifications of this method were made by Freidel, Donnay and Harker [20][21][22] in order to

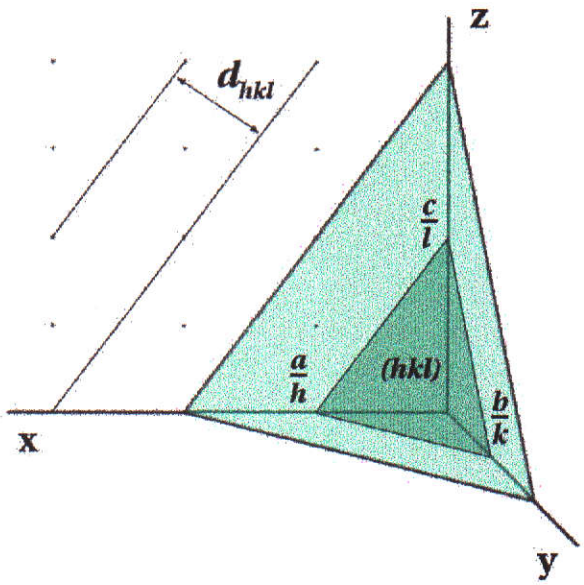

Figure 1.4: The Miller indices (hkl) reference a family of planes *via* the inverse intercepts of the crystallographic directions $\vec{a}$, $\vec{b}$ and $\vec{c}$.

account for internal symmetry within the interplanar spacings. This method of morphology prediction is known as the BFDH technique. While enjoying reasonable success [23], this approach has a significant flaw in that it ignores the nature of the atomic species that comprise the crystal.

The relative growth rates of crystal faces is at the heart of several morphology methods, and can be probed in more realistic ways than a BFDH analysis. In 1955 Hartman and Perdok introduced the concept of a periodic bond chain (PBC), and an alternative scheme for the morphological ranking of faces [24][25][26]. These researchers defined a PBC as an uninterrupted network of growth units connected by strong bonds. Faces that contained no such PBCs parallel to the orientation of the face are termed K-faces. Faces with only one parallel PBC are termed S faces. Finally, those faces parallel to two or more intersecting PBCs are the F-faces. According to definition, there must be a finite number of PBCs for a given crystal. This suggests that growth of the K faces will be most favoured due to extension of the PBCs as growth units are incorporated onto the surface. Thus, the order of morphological importance is rated as F > S > K, which is the same as the ranking of numbers of PBCs parallel to the face.

More recently, Hartman and Bennema [27] found that a quantity called the attachment energy could be employed to predict morphology. The attachment energy, which may be readily calculated from computer simulations, is the energy

released when a complete layer of growth units precipitates on a crystal face. The shape constructed from the values of the attachment energy for the different crystal faces is known as the growth morphology.

Equilibrium morphology predictions are based on the earlier idea of Gibbs, who stated that a crystal of given volume will assume a habit that minimizes its surface free energy [28]. The surface free energy is equal to the surface energy at 0K, and is frequently approximated by the surface energy in morphology predictions. This is generally accepted to be the habit achieved by a crystal in equilibrium with its surroundings.

Non-Ideal Crystals

In practice, crystals are rarely the perfectly ordered structures described above, due to the presence of imperfections or defects. The driving force for the formation of such defects is the gain in entropy when the order of a lattice is perturbed in some way [1]. Thus, defect concentrations tend to increase as the temperature is raised. Such defects are not always undesirable. For example, some defects in metals can increase strength, as they act to block the motions of dislocations (due to some applied force) through the bulk of the material. In other conditions, such as the manufacture of crystals to some standard of purity, it is obviously desirable to limit the presence of defects as much as possible.

The first general class of imperfection is the point defect [29]. This may occur as either a vacancy in a lattice (Schottky defect), an interstitial impurity (Frenkel defect) or an impurity substitution. An additional set of defects sometimes occurs when the lattice is damaged at multiple points, either by heat or the bombardment of X-rays, gamma rays, electrons or neutrons. These defects arise when an electron becomes bound at an anionic vacancy, in order to compensate for an imbalance in charge. This is one variety of the so-called colour centre defect, which gains its name from the resulting change in crystal colour; for example, diamonds can appear blue after electron bombardment.

The second class of defects is the line defects [1], which are also known as dislocations. These defects occur when lattices translate or slip relative to each other, and they appear as either an edge or a screw dislocation, or a combination of both. The edge defect involves a lateral translation (unequal to the lattice repeat vector) of planes of lattice points relative to one another, as shown in Figure 1.5. This defect is also described in terms of an insertion of an extra half plane of atoms above the slip plane of the dislocation. The other form of dislocation, the screw defect, acts as a bridge between two adjacent layers as

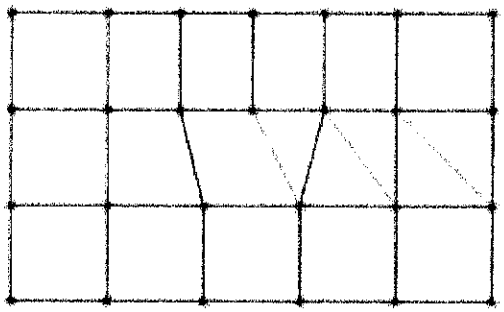

Figure 1.5: Edge dislocation in a lattice.

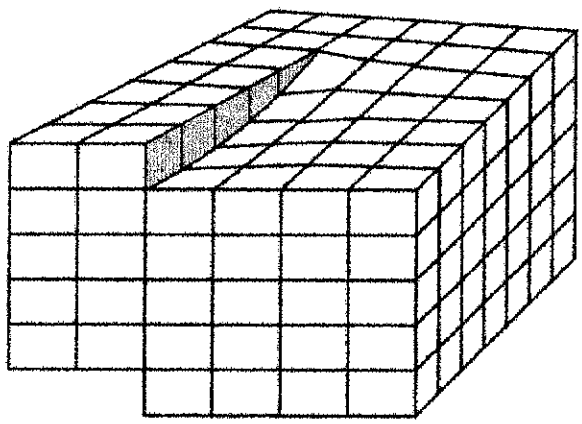

Figure 1.6: Screw dislocation.

illustrated in Figure 1.6. This defect essentially turns two distinct parallel planes into a single plane using a spiral ramp.

The final class of defects are the planar defects. Here, crystals are subdivided into sections or grains that possess different lattice orientations. The planar areas of mismatch separating each region are called grain boundaries. In practice, the structure of these boundaries may be quite complex. However, if the deviation in orientation between adjacent grains is low, the boundaries can be described relatively easily in terms of dislocations [1]. For example, stacking faults are planar regions of mis-register, bounded on either side by partial dislocations of opposite sign. Twin boundaries, may be regarded as a special case of this type of defect, where the different lattice orientations of the constituent grains are related by a symmetry operation. This form of defect is sometimes manifested in the form

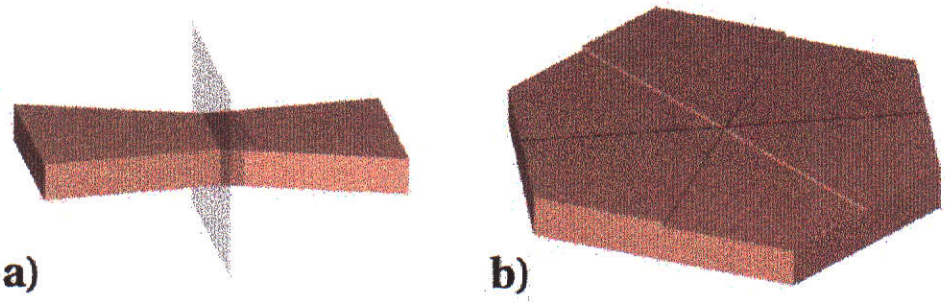

Figure 1.7: a) A simple crystal twin, with a mirror plane relation, b) mimetic twin containing six constituent crystals, and giving the appearance of hexagonal symmetry.

of twinned crystals. These are two (or more) crystals of the same material that are either intergrown or in contact *via* a mutual facet. Characteristically, the constituent crystals are related by a symmetry operation such as a mirror plane or a rotation axis (Figure 1.7a). Multiply twinned crystals, such as the mimetic twin, also occur. These are repeated twins that give the appearance of a higher order symmetry than is actually present, as shown in Figure 1.7b. Polysynthetic twinning, where regular twinning occurs at, or close to, the unit cell level can lead to new crystal structures and superlattices.

1.1.2 Crystallization

Overview

There are many different forms of crystallization, such as vapour growth, melt growth, solution growth and solid state recrystallization. However, all branches of crystal growth are primarily concerned with the mechanisms by which dispersed particles in a liquid or gas phase solidify into a highly ordered crystalline array. The individual particles that condense to form the crystal are known as the growth units. In some cases the growth units may be a full unit cell, or they may be simply the smallest stoichiometric piece of the unit cell. However, the situation can be more complicated than this, as there may be several ways solute particles can be combined to form growth units. In addition, environmental parameters, such as temperature, the local surface structure, and the nature of the growth units might all affect a given system's growth mechanism.

Under typical industrial conditions, where the rate of crystallization is usually as high as possible, the term precipitation is often used instead of crystallization. This distinguishes it from natural formation of crystals; which nearly always

occurs on a much longer time scale. Industrial precipitation is a very old industry, with evidence of chemically manufactured alum produced by the Egyptians in 1500 B.C. [30]. It is also a very diverse field, with a wide range of processes such as those in industrial ore purification plants, drug manufacturing, and industrial gemstone production. Crystallization is not limited to industrial concerns. In nature, there are also a broad range of crystallization processes, such as the growth of kidney stones, bones, and teeth which are of obvious medical importance.

Crystallization Driving Force

Although certain basic principles apply to all forms of crystal growth, this work is concerned with growth from solution. This may be written as a dissolution-reprecipitation equilibrium reaction.

$$A_{aqueous} \rightleftharpoons A_{solid}$$

In order to bias the reaction a particular way, for example to favour precipitation, a net driving force is required. This may be represented by the thermodynamic driving force $\triangle\mu$ [19], which is defined as the difference in chemical potential between initial and final phases,

$$\triangle\mu = \mu_{initial} - \mu_{final} \tag{1.1}$$

where

$$\mu = \mu_0 + RT \ln a \tag{1.2}$$

here a is the activity constant, defined as $a = \gamma C$ with γ the activity coefficient and C the solute concentration. However, for practical reasons the driving force is commonly represented in terms of the supersaturation ratio; which is a dimensionless quantity written as,

$$S = \frac{C}{C_{equilibrium}} \tag{1.3}$$

or,

$$S = \exp(\frac{\triangle\mu}{RT}) \tag{1.4}$$

Nucleation

The initial phase of crystallization, when the first microscopic crystalline material (or nuclei) begin to appear is known as nucleation. There are two main types of nucleation, known as primary and secondary nucleation. Any nucleation that occurs without the aid of pre-formed crystalline material is called primary nucleation. Primary nucleation can be either homogeneous (spontaneous formation of nuclei, dominant at high supersaturations) or heterogeneous (occurs when suitably configured impurities act as templates for the formation of nuclei). Alternatively, any nucleation that occurs due to the assistance of pre-crystallized material is termed secondary nucleation.

Fundamental to an understanding of the process of primary nucleation is the definition of a critical nucleus size [19]. This quantity may be understood to be that size for which all smaller particles are unstable and tend to re-dissolve, whilst all larger particles tend to be stable and able to grow. For the case of homogeneous nucleation, working against the driving force is the difficulty faced by solute particles in having to group together and become orientated into some uniform lattice. This energetic barrier may be expressed as an accompanying gain in surface and volume free energy of a solute nuclei ($\triangle G$) possessing surface area A and volume V, compared to the solute particles in solution. That is,

$$\triangle G = A.\gamma + V.\triangle G_v \tag{1.5}$$

where γ is the interfacial tension per unit surface area, and $\triangle G_v$ is the free energy change for the phase transition per unit volume. If the particle is assumed to be a sphere of radius r, then there is some critical radius r_{crit} for which $\triangle G$ is minimum. Solving gives,

$$r_{crit} = \frac{-2\gamma}{\triangle G_v} \tag{1.6}$$

and

$$\triangle G_{crit} = \frac{16\pi\gamma^3}{3\triangle G_v^2} \tag{1.7}$$

For the case of heterogeneous nucleation, the case is not so simple and greatly depends on the details of the type of impurity acting as a template. In general, the most important term is considered to be the change in the behaviour of the interfacial tension. This may be decomposed into the surface tension between: the nuclei and the solution γ_{ns}, the nuclei and the template γ_{nt}, and between the template and the solution γ_{ts}. These three quantities lessen the gain in free

energy for heterogenous ΔG^* over the homogeneous case ΔG so that,

$$\Delta G^* = \sigma(\gamma_{ns}, \gamma_{nt}, \gamma_{ts}).\Delta G, \qquad \sigma < 1 \tag{1.8}$$

Secondary nucleation, resulting from the presence of small fragments of crystalline material, may occur as a result of internal or external factors. The different sources of these nuclei allow them to be labeled in terms of the so-called "breeding mechanisms", as described by Strickland-Constable [31]. If crystalline seeds are manually introduced into the system, this is labelled as initial breeding. Seeds may also be generated through collision of particles in solution (attrition breeding) or through the hydrodynamic flow of the parent phase (fluid shear breeding). High supersaturations may also result in the formation of fragile dendrites which are dispersed by solution flow and can act as nucleation sites (dendritic breeding). An additional form of secondary nucleation due to semi-ordered solution in the vicinity of the crystal, known as catalytic breeding, is also thought to occur [32].

Crystal Growth

For solution growth, the two fundamental processes of interest are: transport mechanisms within the solvent, and the actual incorporation of growth units. If the rate at which precipitation proceeds is dominated by the first step, then crystallization is said to be diffusion controlled. In this range, the principles of fluid mechanics are employed. However, if growth is dominated by the incorporation step, the process is said to be integration controlled. Integration control is a function of the attraction between the crystal and the growth units. It is affected by the local surface structure and solution parameters, such as temperature and supersaturation.

The attachment of large numbers of growth units to a surface was first treated statistically in the works of Kossel, Volmer and several other workers. The product of these early studies is still employed in some work today in the form of what is known as the Kossel model [33]. This is a simplification of the more general Ising (lattice gas) and Tempkin (multilayer) models [34]. The Kossel model treats a crystallization system as a large assembly of cubic blocks which are either solid particles, or parcels of fluid. Each block interacts only with its nearest neighbours, of which there are 6, according to some appropriate potential function. All solid particles must sit above another solid particle with no defects, gaps or overhangs allowed. This results in a surface that possesses several different adsorption sites. The three most important types are the kink (K), step (S), and surface (F) sites,

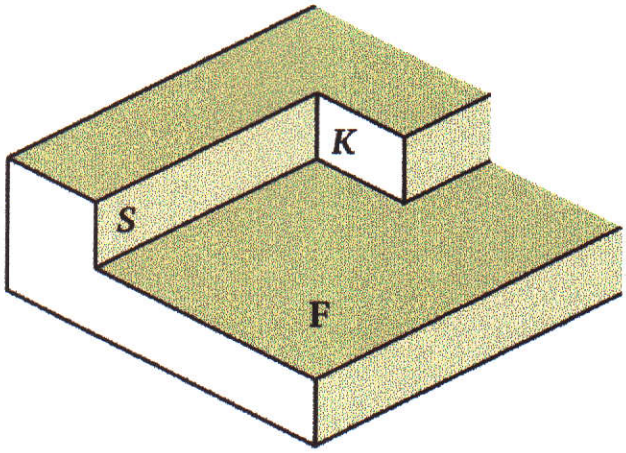

Figure 1.8: Kossel surface showing the three most important sites.

as shown in Figure 1.8.

This model has been used to study growth rates and transitions between smooth and rough growth. Smooth growth is defined as the growth mechanism by which the building blocks of a crystal attach only at the step or kink sites, causing lateral growth until the layer is completely filled. This type of growth is also called layer growth or birth and spread. In this model, the steps and kinks that arise are a product of some initial two dimensional nucleus that has grown sufficiently large. The high energy barrier for the formation of this initial nucleus means that it is the rate determining step for growth. The second form, known as rough growth, is the result of indiscriminate adhering of growth units to the surface of the crystal. This occurs when the step free energy is relatively small. Rough growth may occur if the driving force for crystallization is especially high, when this is known as kinetic roughening [35]. Alternatively, rough growth may also occur when the temperature is sufficiently high. Thermal rough growth and smooth growth are separated by a quantity known as the roughening temperature, as studied in both analytical and computer methods [36][37][38]. Below the roughening temperature smooth growth occurs, while at higher temperatures rough growth dominates.

While successful in some cases, this model has been unable to explain the growth of crystals when there is an insufficient driving force to cause the formation of solid particles on a flat surface. According to the model, an inability to produce 2D nuclei should result in termination of crystal growth whenever a flat surface is formed. This is at variance with experimental findings. In 1951, Burton, Cabrera and Frank [39] described a mechanism by which the barrier of 2D nucleation could be overcome, or in fact bypassed. They showed that the propagation of screw

dislocations produced self-perpetuating step sites (see Figure 1.6) where growth units could incorporate.

Crystal size may also be increased by other means. An important process, known as aggregation, occurs when multiple crystals join to form large clumps. Aggregation, combined with the further growth of the crystalline cluster is termed agglomeration. Agglomeration may be divided into two basic events; the probability of particle collision and the probability of particle cementation. In complex systems, the latter step is sometimes broken into an association function, based on interparticle attraction, and a sticking or consolidation function.

Modelling these complex solution processes revolves around the solution of differential equations describing the system. All processes in nature must obey the conservation equations describing mass and heat balance. However, manufacture of crystals is usually regulated by the desire to obtain a product that conforms to a certain crystal size distribution (CSD). Thus modelling is additionally concerned with solving the time dependent population balance equation [32] for $n(x,t)$, which is the number of particles in size range $[x,\ x+dx]$ at time t,

$$V\{\frac{\partial n(x,t)}{\partial t} + \frac{\partial G(x,t).n(x,t)}{\partial x} + C(x,t)\} = Q(x,t) \tag{1.9}$$

with the volume V assumed to be constant. In this expression, G is the growth and C is the net flux of crystals out of a particular size range $[x,\ x+dx]$, due to internal processes. Q is the volumetric net flux of crystals out of the system attributed to physical input and extraction feeds. Note that C may be further broken down into birth and death expressions, distinguished from the normal crystal growth G, as they are employed to model phenomenon such as agglomeration and attrition.

Solution of the population balance equation is difficult, and assumptions generally have to be made. One of the simplest is the mixed suspension, mixed product removal (MSMPR) scheme. Here, the system is considered to be perfectly mixed, so that each portion of fluid is assumed to have an equal probability of leaving the crystallizer at any given time. Although useful, this is not representative of the complex fluid dynamics within an industrial scale crystallizer [40], and additional studies have focused on developing more realistic treatments. In particular, the types of mixing and non-uniform concentration distributions are of interest because of the large effect upon the resulting CSD [41][42][43].

Non-ideal Crystal Growth

An important aspect is the effect that impurities in solution may have on growth. In examining such impurities, the effect on solution species is of primary importance. Principally, concern focuses on whether complexing takes place between the impurity and the growth unit. If this occurs then nucleation and/or growth rates are retarded. There are also two further aspects to consider. Firstly, is the impurity structurally compatible (and incorporation is energetically favourable) with one or more surfaces of the crystal? Secondly, is the defect repellent to further growth units? If only the first is satisfied, then incorporation will occur, with possibly some defects forming depending on the magnitude of the strain the presence of the impurity causes within the crystal lattice. If both incorporation and repulsion occur, then the defect will cause face blocking and thus an overall change to the crystal morphology.

The above can be considered controllable aspects that, once their workings are fully understood, could be eliminated with application of certain techniques. However there is another broad group of factors that cause deviations in growth rate and crystal morphology that are non-repeatable, and thus not directly controllable. The source of these anomalies is generally attributed to dislocations, adsorbed impurities, and other forms of surface damage to the crystal [35][44]. One or more of these factors may affect the growth rates of individual crystals differently, and collectively lead to the phenomenon known as growth rate dispersion.

1.2 Gibbsite

Gibbsite is an intermediate compound produced during commercial alumina production. This is an important Western Australian industry, with almost eight and a half million tonnes of alumina produced in 1997, contributing approximately 2.1 billion dollars to the economy [45]. As gibbsite production is also the slowest and most difficult stage of the alumina production process, it is naturally the focus of considerable research interest.

1.2.1 Structure and Morphology

Not all crystalline substances with the same chemical formula possess identical internal symmetry. For cases where this occurs, the different crystal structures are known as polymorphs. Gibbsite is one of the four known polymorphs of $Al(OH)_3$.

	Saalfeld			Gale		
Atom	x	y	z	x	y	z
H1	0.101	0.152	0.876	0.077262	0.136549	0.874464
H2	0.595	0.573	0.902	0.574642	0.552033	0.897494
H3	0.503	0.137	0.810	0.494353	0.110970	0.795476
H4	0.971	0.801	0.893	0.951158	0.814745	0.886939
H5	0.293	0.724	0.804	0.295768	0.716748	0.794078
H6	0.815	0.160	0.810	0.805904	0.161311	0.797539

Table 1.2: Unique hydrogen positions in gibbsite, taken from X-ray analysis (Saalfeld) and from quantum mechanical calculations (Gale).

The other structures are: bayerite [46][47], nordstrandite [48] and doyleite [49]. Gibbsite has a monoclinic lattice and belongs to the space group $P\ 2_1/n$. The definitive structural determination was performed by Saalfeld [50] using X-ray diffraction. Unfortunately, low electronic density associated with the hydrogen atoms makes their positions difficult to determine using X-ray diffraction techniques. One solution would be to re-scale all the O-H bond lengths to a standard value in order to correct for the systematic underestimation. However, this is not entirely suitable as there will be some variation in the true material depending on the extent of hydrogen bonding that each site is involved in.

Work by Giese [51] attempted to resolve this by minimizing the electrostatic repulsion experienced by the hydrogens. However, his technique was limited to a fixed electrostatic point charge model. In this work, the hydrogen positions from a fully periodic quantum mechanical calculation, performed by Gale [52], were substituted for the X-ray values. The calculation was performed using the total energy pseudopotential method [53] with a planewave basis set expanded to a cutoff of 650 eV. Norm conserving pseudopotentials were employed for aluminium and oxygen, whilst a bare Coulomb potential was utilized for hydrogen. The Brillouin zone was sampled using just the gamma point which was found to be adequate, based upon tests for higher numbers of K points. The calculations were performed using the GGA gradient corrected density functional of Perdew and Wang [54]. A restricted optimization was performed in which the unit cell parameters, and Al and O coordinates were frozen at the experimental values [50] with only the hydrogen atom positions allowed to vary. These results are presented in Table 1.2 along with Saalfeld's original values.

In crystalline gibbsite, the aluminium atoms are octahedrally coordinated and in layers of pseudo hexagonal rings. These layers are stacked above one another (bearing a similarity to graphite) and held together by relatively strong hydrogen

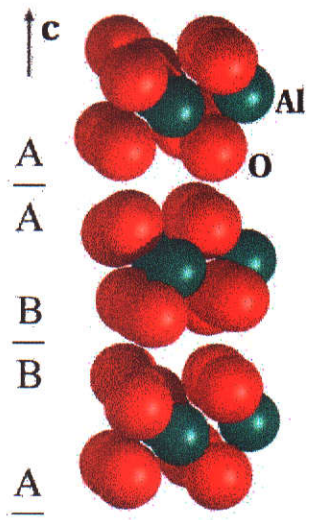

Figure 1.9: Stacking in the gibbsite lattice.

bonding. An illustration of the alternate double layers of oxygen atoms AA and BB, each separated by mirror planes is shown in Figure 1.9. In gibbsite, single hydrogen bonds occur for all H atoms, apart from H2 which participates in a second weaker hydrogen bond [51]. All O-H pairs in gibbsite are either approximately perpendicular or parallel to the plane of the layers (see Figure 1.10). This is distinguished from bayerite which has a more complicated hydrogen bonding pattern with a third more oblique orientation of O-H pairs between the layers (Figure 1.11). For both these structures, the complex hydrogen bonding is a source of some difficulty when attempting to create an accurate interatomic potential model.

In the absence of any environmental considerations, bayerite is thermodynamically more stable than gibbsite [55]. In alkaline environments, gibbsite tends to be more stable than bayerite [56]. In addition, gibbsite is more abundant in nature and typical morphologies vary from hexagonal platelets, to the more common concretionary, massive, clay-like, and stalactitic forms [57][58]. Under industrial conditions, gibbsite usually crystallizes as either hexagonal tablets or diamond prisms [55]. These crystals often grow into mixed agglomerates of both hexagonal and diamond shaped crystals. Elongated hexagonal prisms can also be formed from sodium aluminate solutions at low supersaturations and high temperatures under industrial type conditions. This elongation occurs along the c-axis.

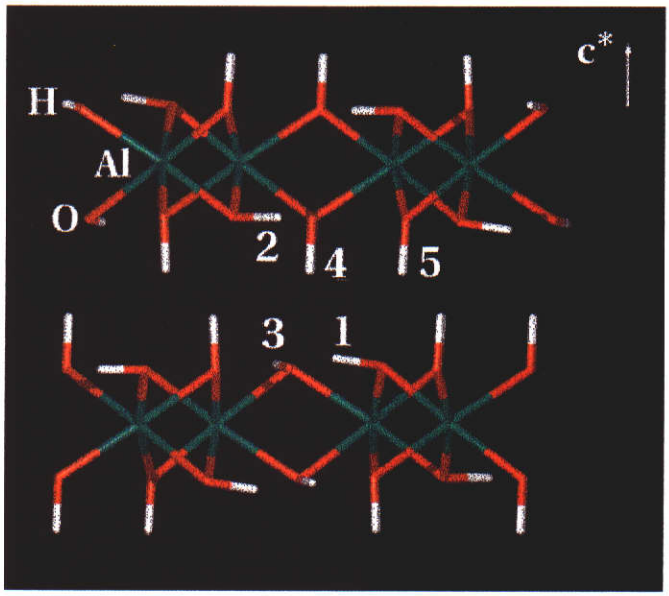

Figure 1.10: The hydroxide double layer configuration in gibbsite, with unique hydrogen atoms labelled.

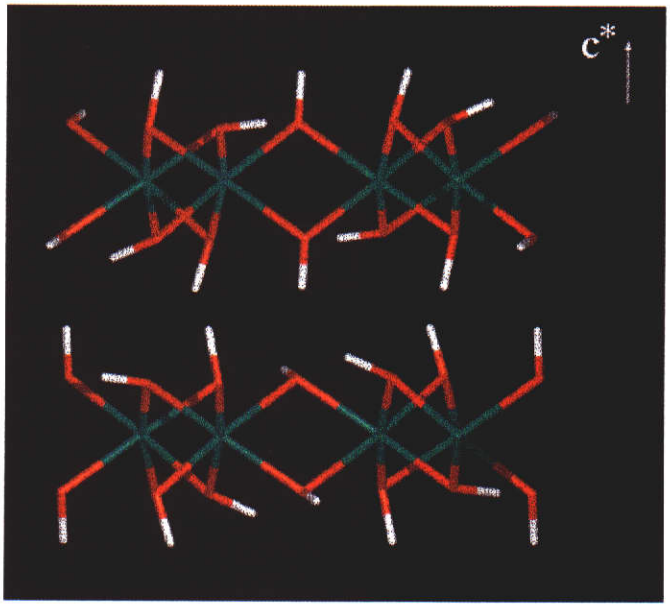

Figure 1.11: The complex double hydroxide layer configuration in bayerite.

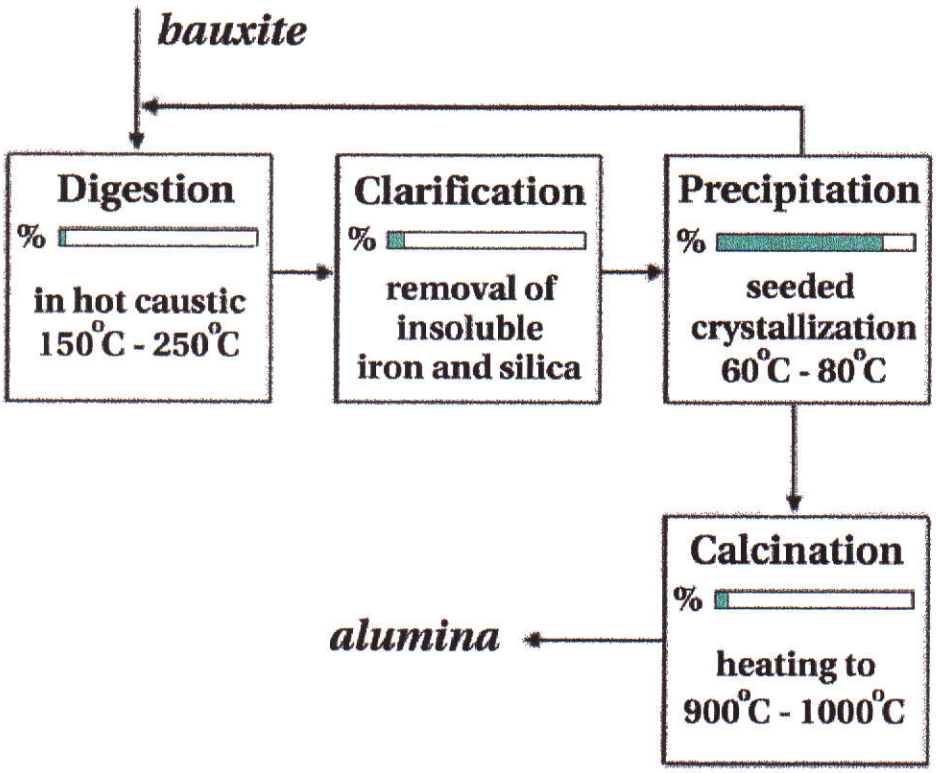

Figure 1.12: A flowchart illustrating the four main stages of the Bayer process. The approximate time required, as a percentage of the overall processing time, is graphically represented for each of the stages.

1.2.2 Precipitation

Overview

Industrially, the precipitation of gibbsite is accomplished using the Bayer process. This method has survived largely unchanged for over a hundred years since it was first formulated by Karl Bayer in 1888. An idealistic four step scheme describing the Bayer process is presented in Figure 1.12. The precipitation step is by far the longest, taking almost 90% of the overall processing time. It may be written in terms of a dissolution-reprecipitation reaction,

$$Al(OH)^-_{4\,(aq)} \rightleftharpoons Al(OH)_{3\,(s)} + OH^-_{(aq)}$$

However, as noted previously, gibbsite contains octahedrally coordinated aluminium atoms, whilst the aluminate monomer $Al(OH)^-_{4\,(aq)}$ is tetrahedrally coordinated. The mechanism by which the monomer undergoes the necessary change to become incorporated in gibbsite $Al(OH)_{3\,(s)}$ is not well understood. This de-

ficiency hinders the investigation of key issues such as: the slow rate of gibbsite growth, and the incorporation of cationic impurities.

Aluminate Solution Species

The identification of species present in Bayer liquors is an obvious stepping stone to elucidating the growth mechanism. Consequently, numerous researchers have conducted IR and Raman studies in an attempt to address this issue [59][60][61]. These studies have demonstrated that most of the aluminium in caustic aluminate solution has four-fold coordination, which is almost universally believed to be due to the aluminate monomer $Al(OH)_4^-$ [62][63]. Unfortunately, many of these studies are inconclusive regarding further species identification, due to the difficulty in interpreting the results [61].

Moolenaar *et al.* [59] proposed that the tetrahedral monomeric ions exist in equilibrium with the similarly coordinated dimeric hydroxo ion $[Al(OH)_3OAl(OH)_3]^{2-}$, with many researchers in agreement [60][64][65][66]. However, limiting the aluminate species distribution to the above two anions may be premature. Gibbsite, which precipitates spontaneously from pure, homogeneous caustic solutions supersaturated with aluminium [67], has a complex structure in which the octahedrally coordinated aluminium atoms are linked to one another by double hydroxo bridges [50]. Clearly, the spontaneous precipitation of gibbsite from solution must involve further changes in anionic structure (four-fold to six-fold aluminium coordination and oxo to hydroxo bridging) together with the formation of oligomeric intermediate species. In addition, there is some evidence to suggest that the IR and Raman data are not unambiguous. There is a difficulty in identifying the precise nature of Al to Al bond bridging in solution, as those frequencies previously attributed to Al-O-Al linkages have been observed in hydroxyaluminate compounds possessing no such single oxygen bridging [68][69].

In addition to experimental spectra measurements, computational methods have also been brought to bear on these problems. Most of these studies have centered around calculating the relative thermodynamic stabilities of the different species. However, there are problems inherent to this approach as well. Quantum chemical calculations [70][71] have been used to explain the ultraviolet spectra of sodium aluminate solutions in terms of three anions, $Al(OH)_4^-$, $Al_2O(OH)_6^{2-}$ and $Al(OH)_6^{3-}$. However, other authors [72] have shown that the UV spectrum of rigorously pure sodium aluminate solutions is largely featureless, and that any observed features can be ascribed to iron or vanadium contamination. The use of both *ab initio* and semi-empirical quantum mechanics by Honglin *et al.* [73]

indicated that the most stable species in sodium aluminate solutions, apart from the monomer, is the $[Al(OH)_3OAl(OH)_3]^{2-}$ dimer which is preferred over the $[Al(OH)_3(OH)_2Al(OH)_3]^{2-}$ dimer. This is consistent with Moolenar's conclusions. In contrast, semi-empirical findings of Gerson *et al.* [74] concluded that the double oxo-bridged dimer is more stable, under different applied conditions. These conflicting conclusions underline the importance of accurately accounting for the environmental conditions in order for any species predictions to be meaningfull.

Until recently, there have been few studies that have attempted to rigorously investigate solvated poly-aluminate species that may exist under basic conditions. This is largely due to the significant increase in computational expense when explicitly including a solvation shell. In recent work by Gale *et al.* [75], a number of aluminate species were modelled using a combination of explicit solvation and a continuum dielectric model. The authors obtained the expected result that the $Al(OH)_4^-$ monomer, typically coordinated with 4 water molecules, is the dominant solution species. The study also predicts that, at high sodium aluminate concentrations, the dimerization to the double hydroxy bridged dimer $[Al(OH)_3(OH)_2Al(OH)_3]^{2-}$is the most energetically favourable.

There is little doubt that geometries calculated *via* quantum mechanics, with the subsequent prediction of vibrational frequencies, have been useful in describing the S_4 symmetry of the $Al(OH)_4^-$monomeric ion [76][77]. Disparity in further results has led to lack of agreement concerning additional species, although there is consensus that there are small amounts of higher order polymeric species. The precise nature of these species is still debatable, with other studies attempting to link (in addition to dimers) higher order species such as tetramers [74] and hexamers [78] with gibbsite precipitation. Whilst increasingly accurate quantum mechanics calculations [75] are now becoming feasible, knowledge concerning polymeric species in concentrated aluminate solutions is still inconclusive.

Precipitation Mechanisms

Like most industrial crystallization processes, the Bayer process utilizes secondary nucleation. This is usually accomplished after the clarification stage by initial breeding with a continuous feed of seed nuclei. Enlargement of crystals occurs through crystal growth and agglomeration, although attrition can occur [79]. The average growth rate, as measured by Misra and White [80], is quite low (1-2 μm/h). There have been many attempts to explain the slowness of the precipitation mechanism, addressing the problem from both computational and

experimental standpoints. However, lack of conclusive data concerning the species present in solution, combined with the complexities of the process have thus far prevented any resolution of the problem. However, it is generally accepted that industrial gibbsite precipitation is integration controlled. This is supported by evidence of the relatively high activation energy of the reaction [81][82][83], the dependence on crystal surface area [84], and independence from agitation conditions [85].

Finely divided globular material on the surface of industrially produced gibbsite crystals has been observed by scanning electron microscopy [86][87]. This has led various researchers to suggest that the material is a precursory stage that eventually undergoes a phase transformation to gibbsite. However, there is some disagreement as to the exact identity of this precursory phase, in part due to the effects of different applied conditions [88]. Some researchers have suggested that the precursory phase material is amorphous $Al(OH)_3$ [89], pseudoboehmite [90], bayerite [74][91] and even a semi-ordered, gel-like layer [92][93].

Researchers have also attempted to characterize the growth mechanism by determining the growth rate dependence on the crystallization driving force. Measurements of growth rates in bulk experiments have shown that the rate is consistent with a spiral growth mechanism [80][81][82][94]. However, the dependence of growth rate on surface area and the negligible growth at low supersaturations suggests that the growth mechanism occurs through a 2D nucleation mechanism [84][95]. Recent atomic force microscope studies of the growth of the gibbsite basal plane (002) by Freij *et al.* [96] has suggested that a combination of these two mechanisms occurs. The authors observe that step growth is dominant at low supersaturations, while the birth and spread mechanism prevails at high supersaturations. Midway between these two extremes, growth *via* both mechanisms was observed.

1.3 Molecular modelling

1.3.1 Atomic Theory

The idea that matter is composed of atoms has formed the basis for much of modern science over the last 200 years. However, until recently, no one had actually seen an atom and even today the pictures at atomic resolution are still fuzzy at best [97]. It is therefore quite intriguing that the existence of atoms was first proposed more than two thousand years ago, by early Greek philosophers.

In the 4th century B.C., Democritus expounded his ideas that all materials could be subdivided into smaller and smaller constituents, but only up to a point. He believed that this indivisible and invisible unit of matter, constituted all visible matter. It was not until the early nineteenth century that scientists began devising experiments to validate and characterize these building blocks of matter [98]. The principles and concepts of modern chemical and molecular theory can be said to originate from the work of John Dalton, around 200 years ago. Dalton constructed wooden balls connected with sticks in order to represent molecules as atoms connected by some form of bonding [97]. In addition to Dalton, work by Avogadro, Faraday and Mendeleev pioneered the understanding of the types of atoms, the rules governing their combinations, and their systematic classification [98].

The idea that matter interacts *via* a force field can perhaps be traced back to Newton's landmark work [99] concerning the theory of gravitation. Although incorrectly performed, Newton even applied a repulsive inverse radial law to atomic interactions in order to explain Boyle's law [3]. However, it was not until the respective works of Boskovic, Clairaut and later Laplace and Gauss that the modern idea of atoms repelling at short range and attracting at long range emerged, and was put on a more rigorous mathematical footing [100][101][102]. Almost in parallel with these early theories of matter, the classical theory of light was developed. The first convincing argument that light was a wave was given by Huygens in 1678 [5]. But it was not until 1873 that the mathematically rigorous electromagnetic equations were written down by Maxwell.

Thus, up until the twentieth century, scientific understanding was largely entrenched in these two core laws of classical physics. The interactions between the atoms comprising solid matter was modelled using simple potential forms that depended only on geometric parameters, whilst light was treated as a continuous electromagnetic wave. However, Planck's work in 1900 on blackbody radiation, and further work by Einstein on the photoelectric effect in 1905, led to the conclusion that light was actually composed of particles (later called photons) [98][103]. Soon, with data from experimental techniques, a new understanding of matter was also achieved. Neils Bohr proposed a revolutionary new model of the atom in 1913. Here, electrons existed in stationary non-classical states centered on the nucleus. A transition between states was accompanied by a release or absorption of discrete amounts of energy. This model was able to predict hydrogen spectra to an accuracy of about 0.02% [5].

However, the Bohr model proved unsuccessful for any atom other than hydro-

gen. The next major progression came in 1924, when de Broglie suggested that as light has particle-like properties, so might matter have wave-like properties. Quantitatively, particles may be thought of as matter waves characterized by a wavelength $\lambda = \frac{h}{p}$, where h is Planck's constant and p is the momentum of the particle. Experimental evidence of electron diffraction soon proved this hypothesis. In 1925, Schroedinger developed the famous eigenvalue equation $E\psi = \widehat{H}\psi$, the solution of which is at the heart of most applied quantum mechanics. In this equation, the stationary energy states (E) of a given matter wave (ψ) are the eigenvalues of the system's Hamiltonian operator ($\widehat{H}$). Some debate ensued as to the precise interpretation of ψ [104], with the generally accepted interpretation eventually becoming that of Born [105]. This states that $|\psi|^2 d\tau$ is the probability of finding the matter wave (or particle) in some finite volume element $d\tau$. Hence, the classical idea of atoms behaving like tennis balls and molecular bonds as springs, was superseded as a representation of physical behaviour.

1.3.2 Computer Modelling

Overview

With the discussion of the previous section, it might be supposed that atom-atom potentials find little current application. However, classical potentials are still useful whenever large numbers of atoms are considered, provided they are not used to model any properties which depend explicitly on the electronic structure [106]. This includes the making or breaking of bonds, although there has been some work involving hybrid classical-quantum mechanics models which treat interactions as classical at long range and quantum mechanically at short range [107][108][109][110]. The use of potentials still has a large niche in computational chemistry, as their simplicity gives a certain flexibility and power as well as yielding considerable savings in computer time over the more rigorous quantum mechanical techniques. Even as computing power attains greater levels, empirical potentials are still likely to remain a tool for probing larger and larger numbers of atoms. There are several good general review articles on the subject of computer modelling [106][111][112][113] and also books detailing theory and methods [114][115].

Most problems of everyday interest are manifestly macroscopic, involving numbers of atoms that are of the order of 10^{23} and higher. Typical workstations today can deal explicitly with groups of atoms of the order of 10^4, and with current supercomputers a system with 10^6 atoms may be attempted [116].

Despite rapid progress, which will undoubtedly expand these limits, simulations with numbers of atoms approaching that of real systems are unlikely to be conducted for some time. This means that some form of approximate method must be sought. Some assistance can be found in the fact that crystals are very ordered structures, which can be reduced to a relatively small non-trivial part and a set of operators describing how this part is replicated through all space. By dividing a model into unique and repetitive parts, rapid convergence summation techniques can be employed to deal with the repetitions.

There are certain advantages to molecular modelling. Computational experiments can be performed with relatively little cost, and are readily repeatable. There is also an advantage in being able to directly monitor behaviour at the atomic level. In addition, once a good model has been constructed (which may not be a trivial task) it is possible to apply it in many different ways that would be too difficult or expensive to implement experimentally. Avoidance of some of the problems facing experimentalists is also enjoyed, as simulations of processes are *in situ* by definition, yet completely non-intrusive.

In general, the basic principles behind most forms of molecular modelling are the same. To begin, an initial configuration must be constructed with the appropriate boundary conditions. These conditions are usually either three dimensional periodic, two dimensional periodic, or isolated clusters. Then, with respect to defined variables, minimum energy configurations are sought. Energy evaluations can be broken into two parts: summation of the short range potentials, and summation of the electrostatic interactions. The summation of the Coulombic interactions in periodic structures merits special attention. This is because the charge interactions are slowly (and conditionally) convergent. The usual approach, known as the Ewald method [117], is to perform a partial transform into reciprocal space to aid the convergence. Although originally formulated for three dimensionally periodic structures, it may also be modified for summation in two dimensions [118].

Solution Studies

These investigations are concerned with generating a number of feasible states which are a representative sample of the entire range of possible system states. From these states, the quantities of interest are extracted in some form of average value. The first simulation of a liquid was accomplished by Metropolis *et al.* [119] who also laid the foundation for what is termed the Monte-Carlo simulation method. This process involves the random generation of possible configurations

for a given system, and is suitable for modelling stochastic processes such as mean atomic displacements in a liquid. The other important method is molecular dynamics, where states are generated by explicitly modelling the trajectories of individual atoms. The first dynamics experiments on a liquid (composed of hard spherical particles undergoing perfectly elastic collisions) was reported by Alder and Wainright [120][121][122]. Since then, simulations containing larger and larger polyatomic species have been investigated [115]. Molecular dynamics is suitable for any time dependent (non-equilibrium) processes, such as nucleation and growth rates, although there has been some recent work employing the Monte-Carlo technique to model rate processes, which have compared favourably to molecular dynamics [34].

Particularly popular in studying nucleation and crystallization processes is the Lennard-Jones fluid, with several studies applying either Monte-Carlo or molecular dynamics techniques to the problem [123][124][125][126]. Simulations by Wolde *et al.* [123] demonstrate a technique whereby the nucleation rate in a Lennard-Jones liquid may be calculated. Density functional theory has also been used to model the phase transformation during crystal and melt growth in atomic liquids by Shen and Oxtoby [124]. They found that a diffusive model is less representative than a collision-limited model. Studies of the crystallization of water have also been conducted [125], indicating that local electric fields may play an important role in the formation of polar crystals. While there has been considerable work concerned with crystallization from the melt, recent studies of crystallization in a simple binary (solute-solvent) solution have been attempted by Anwar and Boateng [126] and Huitema [34]. In both cases, the authors have shown how parameters such as the chemical potential (supersaturation) and the presence of impurities affect the mechanism of the crystallization process.

Bulk Studies

Here a model is sought for the crystal structure(s) of interest that reproduces physically measurable quantities. The most obvious is the atomic coordinates in the unit cell, and the dimensions and shape of the unit cell. If suitable data are available, reproducing other parameters such as elastic constants, vibrational data and relative permittivities should also be attempted [127][128][129][130]. Other means to this end have also been demonstrated. Work by Pavese *et al.* [131] centered on modelling the thermal dependence of the structural and elastic properties of calcite. They used the quasi-harmonic approach [132][133] in order to derive expressions for the change in these parameters with temperature to a

reasonable degree of accuracy. Alternatively, some workers have fitted multiple lattices in order to construct a model that is applicable to a large number of binary and ternary oxides [134]. Due to the fitting of different crystal lattices, this approach samples the values of potentials at a wider range of separations, and is therefore more likely to be representative of the true interatomic forces.

In general, the application of computer modelling to the solid state can be approached in several different ways. The first approach is the minimisation of lattice energy by attempting to reduce the gradients of the forces on individual atoms, which are governed by the interatomic potentials. This method has been used to solve for the crystal structure of complex materials, such as in the work by Gorman *et al.* [135]. Initially, the authors employed a grid search method to approximately locate the positions of cationic species in various microporous materials. The use of a geometric optimization technique then allowed a final refined structure to be determined. In addition, molecular mechanics studies have also been used to conduct investigations on aluminium site preference in MFI (high silicate content) zeolites [136]. Alternatively, Monte-Carlo type approaches have been used to find the minimum energy configuration using random perturbations in the atomic positions. An investigation by Teppen *et al.* [137] illustrates the use of this technique in deriving a potential model for several clay minerals. Molecular dynamics simulations have also been used, for example, in the work by Smirnov and van de Graaf [138]. The authors examined the electronegativity and induced dipole moments of adsorbed methane in MFI and MEL zeolites, which was found to yield important information on the active regions in the pore system of the host zeolite. Combinations of the above techniques have also been employed. The use of Molecular mechanics and Monte-Carlo techniques was demonstrated by Freeman and Catlow [112] in order to reproduce the crystal structure of rutile. To accomplish this, the authors used a Monte-Carlo driven simulated annealing method to refine a random starting point, which was then further refined with energy minimisation techniques.

Surface and Morphology Studies

Calculations involving surfaces are similar to bulk studies, but higher quality potentials are generally needed due to the boundary conditions of solvent or vacuum over a two dimensional periodic surface [113]. There may also be some doubt concerning the wisdom of employing identical interaction terms, regardless of whether the species involved are at the surface or in the bulk. Fortunately, this situation is improved by the fact that atoms at the surface often relax so

that the Madelung potential approximates that in the bulk [139]. However, this relaxation means that separations between atomic species at the surface will differ from their counterparts in the bulk. As a consequence, some degree of error may be introduced by employing potentials that have been explicitly derived to reproduce the bulk structure and properties [97]. Such effects may be partially overcome by fitting to a range of different crystal structures that contain the same chemical species. This will result in a potential model that is robust over a wider range of chemical environments and species separations.

Morphology predictive schemes have been applied to a range of materials from simple oxides [140][141] and organic materials [142], to more complex protein structures such as lysozyme [143]. Studies have even been conducted in order to predict the time dependent morphology of quartz using computer methods [144]. Many such studies are of interest and have been developed in order to provide insight into industrial problems. Studies of the crystal forms of the calcium carbonate polymorphs [145][146] are of considerable geological, chemical and biological importance [147]. There has also been considerable interest in modelling the effect of impurities on calcite morphology [148][149][150]. Work on modelling the morphology of barium sulphate morphology, an important compound in water treatment and off-shore oil platforms [151], has also been conducted [129].

1.3.3 Project Aims

It may be argued that industrial concerns are largely macroscopic and therefore molecular modelling is ill-suited as an investigative tool. As an example, the prediction of crystal size distributions using explicit molecular modelling of growth processes in an industrial crystallizer is beyond the capability of current computers. Thus, this problem must be left to mathematical analysis on a larger length scale. However, certain problems such as the growth mechanism and the effect of growth modifiers are in origin microscopic and amenable to atom-atom interaction studies.

In the literature, computational work that is of relevance to the Bayer process may be roughly divided into two main areas. The first is growth species and mechanism determination, which was discussed earlier in Section 1.2.2. As mentioned, these studies have been inconclusive. To provide further clues, some computational studies of synthetic Bayer liquors have been conducted as part of this thesis. Although it is close to impossible to model all the possible components in a Bayer liquor, the major species thought to be present have been included. These studies have primarily been concerned with investigating the possibility of

solution structuring.

The second area is concerned with the development and utilization of a model for the gibbsite crystal structure. Potential developmental work in this area, apart from studies by Lee *et al.* [152], have not directly been concerned with the Bayer process. In addition, all these works suffer from certain limitations that are discussed in Section 3.3.

Thus, there is a deficiency in potential model development, which will clearly affect subsequent modelling studies. This work attempts to address this problem by constructing a model that reproduces available gibbsite observables to a high degree of accuracy. The gibbsite model was applied to three problems. Firstly, both equilibrium and growth morphologies were calculated and compared to experiment. Secondly, the possibility of twinning was investigated. Thirdly, with the relaxed surfaces generated from these morphology studies, the effect of sodium and potassium incorporation was studied. This part of the project was aimed at obtaining a greater understanding of the known phenomenon of cation incorporation in gibbsite.

In summary, the aims of this project were to gain a better understanding of phenomena, such as solution structuring and final morphologies, that are relevant to the precipitation of gibbsite. Care has been taken to use experimental data with modelling techniques in order to produce a model that is as physically representative of real processes as possible. Thus, the derived model is of sufficient quality to be employed in related studies in the future.

Chapter 2

Principles of Computer Modelling

The investigation of systems comprising multiple interacting particles is a discipline that probes events at the atomic scale right through to processes occurring at the cosmic level. Independent of scale, or the types of forces acting between the particles, there is a singular problem in that no known exact solution exists. Indeed, there is no known general analytical solution to any system with three or more mutually interacting bodies [153]. Thus, numerical techniques provide the only means of solution. All of these numerical methods are based on solving a system of equations, where each equation represents the net work done on a particle by all others in the system. In molecular modelling, all structures are composed of various atomic species; with particle interactions dominated by electrostatic forces. This makes solution somewhat harder than in the gravitational N-body problem, as the particles can either attract or repel depending upon the sign of their charges. In general, methods of solution can be categorized according to whether the interacting particles comprising the system are treated as classical particles, or as quantum wave packets. It is currently believed that quantum mechanics offers the definitive description of how particles of matter interact at the atomic level. However, despite the various levels of approximation available, a quantum mechanical approach is usually far too time consuming in even modestly macroscopic simulations. Thus, a necessary part of computer modelling is selecting the most suitable method, from a range of different types and levels of approximation.

2.1 Quantum Mechanics

The formalism of quantum mechanics is markedly different from classical mechanics. In particular, certain physical quantities are expressed as expectation values

Observable quantity	Classical mechanics	Quantum mechanics
Position	$\vec{r}$	$\vec{r}$
Momentum	$\vec{p} = m\frac{d\vec{r}}{dt}$	$\frac{\hbar}{i}\nabla$
Kinetic energy	$\frac{p^2}{2m}$	$-\frac{\hbar^2}{2m}\nabla^2$
Potential energy (conservative field)	$V(\vec{r})$	$V(\vec{r})$
Total energy	$\frac{p^2}{2m} + V(\vec{r})$	$-\frac{\hbar^2}{2m}\nabla^2 + V(\vec{r})$

Table 2.1: Classical properties and their quantum mechanical counterparts. For some vector quantity $\vec{x}$, the magnitude is written as x ($\equiv |\vec{x}|$).

of an operator that acts on the system's wave function Ψ, and not as intrinsically measurable properties of the particle itself. Whilst some of these particle properties, such as position, remain the same in both classical and quantum descriptions, others are quite different. Table 2.1 indicates the relationship between classical and quantum mechanical expressions for some of the common particle properties. The momentum operator shown is a simplification of the more general time dependent form [105]. In both formalisms, the total energy of a system is the sum of the kinetic (K) and potential energy (U) components, and is also called the Hamiltonian (H).

In quantum mechanics, a particle is thought not to have an inherent value of momentum or kinetic energy, until it has been measured by an observer. This leads to the concept of describing physical quantities, such as the momentum of some particle j, as the expectation value of the (normalized) wavefunction Ψ. That is,

$$\left\langle \vec{p}_j \right\rangle = \int \Psi^* \left(\frac{\hbar}{i} \nabla_j \right) \Psi \, d\tau \tag{2.1}$$

Where the operator ∇_j acts with respect to the coordinates of the particle j. Expression 2.1 may also be written in Dirac notation as,

$$\left\langle \vec{p}_j \right\rangle = \left\langle \Psi \left| \frac{\hbar}{i} \nabla_j \right| \Psi \right\rangle \tag{2.2}$$

In practice, we are usually interested in applying quantum mechanics to determine the properties of an atomic or molecular system as a whole, rather than the individual motions of the particles that comprise it. The total energy E of the system is of particular interest, and can be found by taking the expected value of the Hamiltonian,

$$E = \langle \Psi | H | \Psi \rangle \tag{2.3}$$

The expression shown in 2.3 is also known as the time independent Schroedinger equation, and has only been solved analytically for simple problems. One well studied example is the spinless two-body hydrogen-like atom. Here, a single electron moves in a Coulomb field created by the nucleus which is assumed to be stationary at the origin. Due to the disparity in masses, the nuclear speed will usually be much less than that of the electron. Nuclear and electronic motions can thus be treated separately (Born-Oppenheimer approximation). Hence, for the electron,

$$U = -\frac{e^2}{4\pi\epsilon_0 r} \tag{2.4}$$

and

$$K = -\frac{\hbar^2}{2m_e}\nabla^2 \tag{2.5}$$

with the computed eigenfunctions Ψ_i and corresponding eigenvalues E_i describing the possible electronic orbitals and their energy levels respectively. The index i represents the particular state, which can also be written in a more informative way as three quantum numbers (n, ℓ and m_ℓ) that uniquely specify the stationary state (ignoring spin). Solutions are known to be separable into angular and radial components [154]. The angular parts comprise the well known spherical harmonic functions $Y_\ell^{m_\ell}$, while the radial components $R_{n\ell}$ can be decomposed into terms involving the Laguerre polynomials $L_{n+\ell}^{2\ell+1}$. Hence,

$$\Psi_{n,m,\ell} = Y_\ell^{m_\ell}.R_{n\ell} \tag{2.6}$$

with corresponding eigenvalues,

$$E_n = \frac{e^4 m_e}{(4\pi\epsilon_0)^2 2\hbar^2 n^2} \tag{2.7}$$

For many electron systems, the interaction terms make it impossible to obtain a general analytical solution. Common approximate solution methods include the so-called *ab initio* and semi-empirical approaches. The former has two main subdivisions: the Hartree-Fock method including correlation methods that employ exact exchange, and the Density Functional approach. The term *ab initio* is a slight misnomer; its usage arises from the fact that after the initial Hamiltonian is written down, the problem is solved with rigour and without recourse to experimental data [155]. Conversely, semi-empirical methods do employ experimental data to simplify the solution process, with some subsequent loss of generality.

2.1.1 Hamiltonian Approximations

Hartree Fock Theory

A common Hamiltonian approximation, the Hartree-Fock method, treats each electron as moving in some field v_{eff} that represents an average effect created by the other electrons. In this method, the approximate eigen-solutions (ϵ^{HF}, ψ) will satisfy,

$$\epsilon^{HF} = \langle\psi|\, h^{HF}\, |\psi\rangle \tag{2.8}$$

With the use of atomic units for notational convenience, the Hartree-Fock Hamiltonian (h^{HF}) can be separated into terms concerning single electrons,

$$h(\vec{r}_1) = -\frac{1}{2}\nabla_1^2 - \frac{Ze}{r_1} \tag{2.9}$$

and describing pairwise electronic interactions,

$$g(\vec{r}_1, \vec{r}_2) = \frac{1}{r_{12}} \tag{2.10}$$

Thus, the full Hamiltonian is written as [105],

$$h^{HF} = \langle i\,|h(\vec{r}_1)|\,j\rangle + \sum_{k,l}\left(\langle ij\,|g(\vec{r}_1, \vec{r}_2)|\,kl\rangle - \frac{1}{2}\langle ik\,|g(\vec{r}_1, \vec{r}_2)|\,jl\rangle\right) \tag{2.11}$$

The first term inside the two-electron summation can be interpreted as the Coulomb repulsion experienced by two charge clouds. The second term is the exchange energy, which arises from the indistinguishability of the electrons. Physically, it represents the correlation of electron motions with like spin.

Based on purely physical arguments, the true multi-electronic wavefunction Ψ is expected to show that electronic motions are not independent. That is, the mutual repulsion of electrons will result in some degree of correlation between their motions. As the Hartree-Fock method treats each electron as moving in an averaged field due to the other electrons, the correlation between individual electrons of opposite spin is not accounted for. Clearly, this will result in energies that are too high due to excessive repulsion. Typical energies computed by the Hartree-Fock method (called the Hartree-Fock limit) are good to within about one percent of the experimental energy [156]. Although this is a reasonable estimate in absolute terms, it is insufficient for most chemical purposes.

A common method to include electron correlation into the result achieved

by a Hartree-Fock calculation is to add it as a perturbation, so that the system Hamiltonian becomes

$$H = h^{HF} + \lambda H^{corr} \tag{2.12}$$

For the case where λ (a book-keeping parameter) is 0, the solution is the Hartree-Fock limit, and is assumed to have known solutions $(\epsilon_i^{HF}, \psi_i)$, whilst when $\lambda = 1$, the full electron correlation due to H^{corr} is introduced. It can be shown [105] that the change in energy due to the extra term can be written as

$$E_i = \epsilon_i^{HF} + \lambda \langle i | H^{corr} | i \rangle + \lambda^2 \sum_{i \neq j} \frac{\langle i | h^{HF} | j \rangle \langle j | h^{HF} | i \rangle}{\epsilon_i^{HF} - \epsilon_j^{HF}} + O(\lambda^3) \tag{2.13}$$

where the last term represents the third order and higher terms. A calculation that includes correction terms up to order λ^n is said to have been performed at the MPn (Møller Plesset) level.

Density Functional Theory

This method arose from the Hohenberg-Kohn theory [157], and is essentially based on the assumption that the ground state energy of a system is a functional of the calculated electron density ρ,

$$E[\rho] = K[\rho] + U[\rho] + J[\rho] + \epsilon_{xc}[\rho] \tag{2.14}$$

Whilst the single electron kinetic and nuclear interaction energy terms (K and U) remain comparable to the Hartree-Fock case [114], this formulation has some consequences for the two electron interaction terms. The direct Coulomb term can be written more explicitly as,

$$J[\rho] = \int \int \rho(\vec{r}_1) g(\vec{r}_1, \vec{r}_2) \rho(\vec{r}_2) d\vec{r}_1 d\vec{r}_2 \tag{2.15}$$

which is equivalent to pairs of electrons being treated as mutually repelling charge clouds. The non-classical term $\epsilon_{xc}[\rho]$ is included to account for the electron exchange and correlation contributions. The search for expressions describing the exchange and correlation energy is difficult, and has challenged researchers for many years [158]. The local density approximation (LDA) is a common method used to develop explicit expressions for these terms. In this scheme, it is assumed that $\rho(\vec{r})$ is slowly and smoothly varying throughout the molecule. Thus, the

bound electrons can be considered to behave somewhat like a uniform electron gas. However, with the use of density gradient expansions, functionals that take into account the inhomogeneity of the electron gas have also been developed. Some commonly used expressions are the PW (Perdew and Wang) and LYP (Lee, Yang and Parr) correlation functionals and the B (Becke) gradient corrected exchange functional [159].

Semi-Empirical Approach

As the name implies, the central idea behind these methods is to simplify the Hamiltonian by replacing integral evaluations with experimentally derived data. These methods usually assume the core electrons to be "frozen", and thus having only a modifying effect on the potential of the nucleus [155]. For the valence electrons, part of the one electron operator $h(r_1)$ (equation 2.9) can be written as terms involving orbitals centered on single atoms. These terms are not evaluated explicitly, but taken from experimental valence ionization energies or electron affinities. For molecular systems, the semi-empirical methods are usually based on the parameterization of single electron terms only. This is because all two electron contributions require different parameterizations for a given atom, depending upon the other atom type involved. However, much of the difficulty in solving for the wavefunctions and energies of a system lies in evaluating the two electron integral terms, $\langle ij\,|g(\vec{r}_1, \vec{r}_2)|\,kl\rangle$. The semi-empirical methods can be classified in part by how they treat these terms. A popular method, the modified neglect of differential overlap (MNDO) [160] keeps only terms where $i = j$ and $k = l$ and also those where all the orbitals are centered on the same atom. There are three main variations of this method: MNDO, AM1 [161] (second parameterization of MNDO) and PM3 [162] (third parameterization of MNDO). The latter represents perhaps the state of the art for these types of calculations [155].

2.1.2 Self Consistent Fields

There is a problem in finding the solutions E_i, as the analytical form of Ψ_i is unknown. The usual approach is to write a new wavefunction ψ_i in terms of a linear combination of known functions. This reduces the problem of determining an unknown function, to that of solving for unknown expansion coefficients. The functions used to describe the wavefunction should span the solution space in order to eventually reach the correct answer. Therefore, such a group of functions is also known as a basis set. By the variational principle [156] such a construction

places an upper bound on the true ground state energy,

$$E_i^{(0)} \leq \frac{\langle \psi_i | H | \psi_i \rangle}{\langle \psi_i | \psi_i \rangle} \tag{2.16}$$

There remains some difficulty, however, as the solutions (E_i, ψ_i) can only be computed after terms involving the wavefunction ψ_i are explicitly evaluated. In essence, this means that the solution is required in order to evaluate itself. The usual approach involves a feed-back loop, where an initial guess at the solution becomes refined through successive iterations. For some approximate eigenfunctions $\varphi^{(i)}$, obtained using the old set $\psi^{(i)}$, any new information in the solution can be expressed as

$$\triangle\varphi^{(i)} = \varphi^{(i)} - proj_{\psi^{(i)}}\varphi^{(i)} \tag{2.17}$$

where the second term evaluates the projection of the approximate solution onto the old solution. Thus, a new set of eigenfunctions can be written,

$$\psi^{(i+1)} = (a\psi^{(i)} + b\triangle\varphi^{(i)}) \tag{2.18}$$

with a and b the adjustable mixing parameters. When $\psi^{(i+1)} = \psi^{(i)}$ the solution is said to be self-consistent and equal to the best possible wavefunction within the current basis set. This is the self consistent field (SCF) procedure.

2.1.3 Basis Sets

For a multi-electron atomic system, a logical choice for basis functions might be the single electron solutions from the hydrogen-like atom (ϕ_i). Typically, products of these single particle functions are used to represent the system wavefunction. However, the total wavefunction is required to be antisymmetric with respect to the exchange of particle labels. Physically, this is a consequence of the Pauli exclusion principle which states that no two electrons can have identical quantum numbers. A convenient way to write such an n-electron wavefunction is the Slater determinant,

$$\psi(1...n) = \frac{1}{\sqrt{n!}} \begin{vmatrix} \phi_1(1) & \cdots & \phi_n(1) \\ \vdots & \ddots & \vdots \\ \phi_1(n) & \cdots & \phi_n(n) \end{vmatrix} \tag{2.19}$$

since a determinant changes sign when two rows are swapped. In molecular systems, the atomic orbitals are replaced by molecular orbitals (MOs). A common

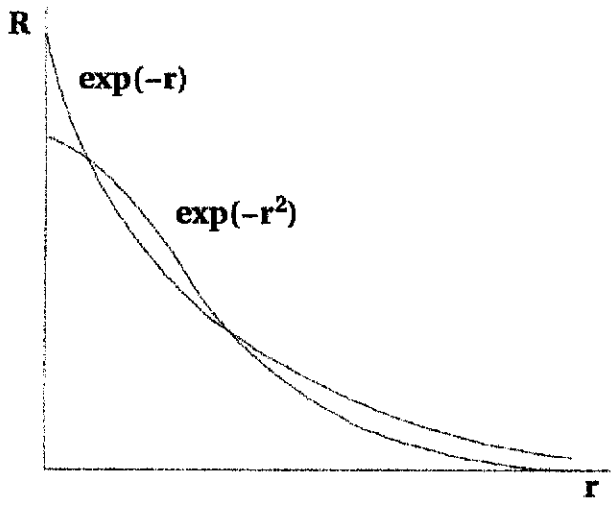

Figure 2.1: Exponential and Gaussian functions.

method of formulating a MO is to employ a linear combination of atomic orbitals, this is the LCAO method.

The radial part of the orbital expression R from equation 2.6 is often approximated with,

$$R = N.r^{(n^*-1)}e^{-\xi.r} \tag{2.20}$$

where n^* the effective quantum number and ξ allowing for the screening of the nucleus by the inner electrons. There are empirical rules given by Slater for determining n^* and ξ, and these functions are thus known as Slater-type orbitals (STOs) [163].

However, products of $exp(-r)$ are difficult functions to integrate, while Gaussian functions $exp(-r^2)$ are much easier. In addition, the product of two Gaussians is also a Gaussian, which makes the evaluation of many centered (molecular) overlap integrals much simpler. Unfortunately, Gaussian functions do not quite match the features of the exponential type functions, as shown in Figure 2.1. The illustration indicates how Gaussian functions underestimate the electron density near the origin due to the lack of an exponential cusp. Also, at longer range, Gaussians decay faster than exponential functions. One method of overcoming these deficiencies is to use more than one Gaussian function for a single atomic orbital. This enables a more accurate description of the electron density to be achieved. Such functions are denoted STO-nG, indicating that a linear combination of n Gaussians is used for each atomic orbital. The STO-3G is a commonly used minimal basis set, used when only a qualitative answer is desired or larger

basis sets are computationally prohibitive. In addition, this basis set has been used to determine geometries of compounds involving lighter elements quite effectively, in part due to a fortuitous cancellation of errors [164]. However, the computed energetics for this basis are generally much worse.

Another commonly used class of basis sets are the split valence $n-ij$G family. Here n is the number of Gaussians used for core orbitals, while the i and j indicate that a linear combination of i functions are used for the inner part of the valence orbitals and j functions for the outer part of the valence orbitals. This splitting of the valence orbital is also termed a double zeta basis. Other sets exist with triple zeta valence splitting. The split valence sets 3-21G up to 6-31G are something of a compromise set, providing reasonable geometric evaluations and progressively better energetics. The incremental cost is relatively small by today's computing standards. In general, more functions are used for the core electrons compared to the valence electrons because of their greater contribution to the electronic energy. However, when highly polarizable atoms, negatively charged species or some form of molecular dissociation is studied, higher quality basis sets than those mentioned above should be used.

To model polarization of an orbital with angular quantum number ℓ, basis functions from $\ell + 1$ are used. Thus, for the hydrogen atom, p functions are used. To denote such a basis set, the symbols ** are appended to indicate that polarization functions are included for all atomic species. In addition, since Gaussian functions vanish faster than exponential functions, it may become necessary to use diffuse functions. These sets have extra Gaussians that tend to vanish at longer than usual range, and are appended with ++ to indicate that diffuse functions are included for all atoms. For historic reasons, + and * are used to indicate that all basis sets excepting those for the hydrogen atom have these extra functions. With the 6-31G* set at the MP2 level an average level of agreement of bond lengths to 0.018Å has been achieved for a large number of closed shell electronic systems[165]. Larger basis sets such as 6-311G** have been shown to give agreement in the 0.006-0.013Å range [166].

2.2 Empirical Potentials

Despite the increasing processor power available for simulations, employing quantum mechanics is often far too computationally expensive. Indeed, the study of overall trends and properties of large numbers of atoms may not even demand the precision of a quantum mechanical calculation. An alternative method of describ-

ing atomic and molecular interactions is to use empirically fitted functions. Here, the central focus is on replacing complex quantum mechanical effects with some form of average interaction field. It is convenient to classify all such interactions as either electrostatic, intermolecular or intramolecular. However, the distinction between these types of interactions is not always clear cut. In this work, electrostatic interactions will refer to all monopole-monopole terms, and only in special cases to the dipole moment. Intermolecular functions will be used to describe the short range repulsive forces, as well as the longer range induced dipole interactions. Lastly, intramolecular potentials are utilized to describe the bonding, angle-bending and torsional energy within a molecule. The application of such empirical functions is frequently termed molecular mechanics in the literature.

2.2.1 Functional Forms

Electrostatic Interactions

The monopole interactions between charged species are evaluated *via* the usual coulombic term,

$$E_{ij} = \frac{q_i q_j}{4\pi\epsilon r_{ij}} \tag{2.21}$$

with suitable point charges allocated to the atoms. Assignment of point charges is frequently taken from the results of an electron population analysis of the molecule in question. Whilst this approach has been questioned by some workers, it has nonetheless proved reasonably successful [167][168]. An issue that has been raised in the literature, is the validity of using the dielectric constant of free space $\epsilon = \epsilon_0$. In order to account for the effect of neighbouring electronic charge clouds, it has been suggested that a distance dependent dielectric $\epsilon \propto r_{ij}$ be utilized [169][170]. However, this leads to much steeper gradients on the electrostatic interactions, rather than the desired uniform muting. Other researchers have suggested the usage of various fixed values that are greater than that of a vacuum [171][172].

In some cases, it may be undesirable to model the electrostatic interactions in terms of point charges alone. The polarizability of an atom can be modelled using the method described by Dick and Overhauser [173]. This technique involves coupling a shell (massless point charge) to the positive core of an atom, from which it is Coulombically screened.

$$E(r) = \frac{1}{2}kr^2 \tag{2.22}$$

Here, the displacement of the shell from the core is represented by r, with a corresponding restoring force k. Thus, the polarizability α can be written as,

$$\alpha = \frac{q^2}{k+f} \tag{2.23}$$

with q the shell charge and f the damping due to short range forces. Physically, the shell can be thought to represent the polarizable valence electrons, while the core charge is the nucleus plus inner shell electrons. In the literature, a dipolar shell model is frequently employed for oxygen in inorganic compounds [128][130][131][174], although the polarizability of some of the heavier metallic cations has also been modelled [134].

Intermolecular Functions

A rigorous explanation of van der Waals attraction was first attempted in 1912 by Keesom, and was based on the interactions between permanent dipoles. A refinement on this analysis by Debye enabled the effect of induced dipoles to be quantified. However, the attraction between neutral molecules was not accounted for until the quantum mechanical treatment of dispersion by London in 1930 [175]. In these works, it was found that the attractive forces approximately follow an inverse sixth power relationship with distance; this is reflected in the empirical functions.

A commonly occurring non-bonded form, known as the Lennard-Jones potential [176], can be written as

$$E(r) = \frac{A}{r^m} - \frac{B}{r^n} \tag{2.24}$$

with m and n chosen to reflect the repulsion and van der Waals forces respectively. For a value of $n = 6$, m has been shown to have an optimal value of 9 in some cases [114], and 12 in others [177]. Thus, given the coefficients A and B of the repulsive and attractive forces, the interaction energy at an arbitrary distance r can readily be evaluated. In comparable fashion, the Buckingham potential [178] is defined as,

$$E(r) = Ae^{-\frac{r}{\rho}} - \frac{C}{r^6} \tag{2.25}$$

where the interacting atom types are separated by a distance r. Thus, the pairwise energy between any two atoms may be evaluated if the short range repulsion parameters A and ρ, plus the coefficient C of the long range attraction force are

known. Quantum mechanical calculations on helium by Slater [179] and neon by Bleick and Meyer [180] have indicated that short range atomic repulsion is adequately represented by an exponential relation. However, as there is no positive inverse power in the Buckingham potential; the energy as $r \rightarrow 0$ experiences a maximum point past which the potential is no longer repulsive. This shortcoming is usually of little concern as it occurs at reasonably close atomic separations. Hence, it is unlikely to cause problems at equilibrium or near-equilibrium situations [181].

Intramolecular Functions

Bonded interactions between two atoms are sometimes described using the expansion,

$$E_{ab} = \frac{1}{2}k(r - r_0)^2 + \frac{1}{6}k'(r - r_0)^3 + ... \tag{2.26}$$

where r is the instantaneous separation of the bonded atoms and r_0 is the equilibrium bond distance. If the bond distance is not stretched significantly from equilibrium, the third order (and higher) terms may be set to zero so that the bond stretching is purely harmonic. An additional form of bonded potential, the Morse potential, may be written as

$$E_{ab} = D_e((1 - e^{-\alpha(r-r_0)})^2 - 1) \tag{2.27}$$

with D_e, α and r_0 completely defining the bonded interaction energy. For covalent bonding between three atoms, a harmonic angle bending potential may be used,

$$E_{abc} = \frac{1}{2}k(\theta - \theta_0)^2 \tag{2.28}$$

where the equilibrium bond angle is θ_0, the instantaneous angle θ, and the restoring force k. It is usual to employ no higher than a four body potential in a molecular mechanics system. The four body term is also known as the torsional potential, which may be expressed as,

$$E_{abcd} = \frac{1}{2}k(1 \pm cos(n\phi - \phi_0) \tag{2.29}$$

with n the periodicity, k the torsional constant, ϕ the torsional angle, and ϕ_0 the equilibrium torsional angle.

2.2.2 Fitting

Whilst suitable potential forms can be readily chosen for a given system, the parameters specific to the atomic species involved are generally not known. One of the most common method of obtaining these parameters is to refine an initial guess by the method of least squares. This approach assumes that there is a set of known observables $\{y_i^{obs}\}$ which the model must reproduce. The least squares method seeks to determine the unknown model parameters $\vec{x} = (x_1, x_2, ..., x_n)$ so that the calculated observables $\{y_i^{calc}\}$ minimize some error function [182],

$$f_{err} = \sum_{i=1}^{m} w_i (y_i^{obs} - y_i^{calc})^2 \tag{2.30}$$

with weighting parameters w_i, and the usual requirement that,

$$\frac{df_{err}}{d\vec{x}} = 0, \quad \frac{d^2 f_{err}}{d\vec{x}^2} > 0 \tag{2.31}$$

However, the results yielded by this method can be biased considerably by calculated properties with large deviations from the experimental value. This may not be desirable, particularly if the experimental values have an associated measurement error. Hence, judicious choice of weights (w_i) may be required in order that such points do not completely dominate the approximation.

There are many different methods for finding minima on a given n-dimensional hypersurface. These can generally be sub-divided into three groups according to whether the first derivatives, the second derivatives, or neither are used. The latter includes the simplex search method, which is discussed further in 4.3.1. The two main first order methods are the steepest decent and the conjugate gradient techniques [168]. The steepest decent approach involves successive movements on the hypersurface according to the direction of the gradient. However, this method can experience oscillations in an extremely narrow minimum. Hence, the conjugate gradient method is sometimes used as the direction of movement is computed from the both the gradient vector and the previous direction of motion, which avoids the oscillation problem [114]. The second derivative methods, which require the gradients as well, are typically based around the well known Newton-Raphson method. In this scheme, an iterative procedure is used that generates a new vector $\vec{x}_{i+1}$ from $\vec{x}_i$ that is closer to the minimum,

$$\vec{x}_{i+1} = \vec{x}_i - \frac{f'(\vec{x}_i)}{f''(\vec{x}_i)} \tag{2.32}$$

Refinements on this basic idea, such as the Quasi-Newton methods, revolve

around special techniques to estimate the second derivative terms, as the direct evaluation and subsequent matrix inversion constitute the most time consuming steps [114]. In general, studies employing second derivative methods have proven to be particularly efficient in lattice simulations [106][183].

Usually, when searching a hypersurface, the global minimum is sought. However, in reality there are frequently local minima present. All the above methods have no intrinsic way of distinguishing a local minimum from a global minimum. This can present a problem if not dealt with. One approach is to perform more than one minima search, with each starting from a different position on a grid that spans the hypersurface domain. By adjusting the mesh size of the grid, the absolute or global minimum can usually be located. In addition, incorporating a known atomic interaction function from another (suitable) system can also be useful. This is because, by holding this known potential fixed, the other potential functions are less likely to locate to some spurious minima.

Observables

In order to employ a potential model to calculate some unknown quantity with confidence, it is clear that as many data points as possible should be sampled in the fitting procedure. For crystal systems, the most common range of observables employed in fitting are: the lattice parameters, elastic constants, permittivities and phonon frequencies. However, the main sources of experimental data are the atomic positions and the lattice parameters. These are weighted higher in the fitting procedure as they dominate the properties subsequently calculated from the potential model. Multiple structures may also be fitted in order to increase the number of sample points for each potential. This assists in avoiding potential models that only reproduce the correct interaction energy for the few interatomic distances sampled by one or two structures. This is advisable when attempting to transfer the model to a different structure, relaxing a cleaved surface, or attempting to model defects, as large motions may occur and the potentials may well yield erroneous energies.

As previously discussed, in some applications it is not feasible to apply quantum mechanics. However, when experimental data is imprecise or unavailable, it may be convenient to use classical atom-atom potentials that approximate a quantum mechanically derived energy hypersurface. The dimension of this hypersurface corresponds to the number of degrees of freedom in the system of interest. Thus, energy calculations for a number of distorted configurations in the system of interest become the observables to be fitted by a potential model.

Unfortunately, generating a hypersurface for the solid state is computationally demanding, and calculations in the gas phase are frequently used instead. For gas phase calculations involving ionic materials, caution is required as ionicity may change from the gas to the solid state. However, with the application of suitable care, this approach has been shown to yield good results. In the work of Jentys *et al.* [184], *ab initio* LDA calculations on clusters of atoms were conducted. The resulting energy hypersurfaces were employed to generate interatomic potentials suitable for modelling CdO and CdS inclusions in zeolite. In addition, since no charge transfer was observed to occur between the inclusions and a zeolite fragment, the assumption of constant ionicity and the use of atom-atom potentials was justified.

2.3 The Crystal Lattice

Lattice Energy

The lattice energy of a crystal is defined as the energy required to change one mole of solid at 1 atmosphere and 0K into its gas phase constituents.

$$AB_{(s)} \quad \rightarrow \quad A^{+}_{(g)} + B^{-}_{(g)}$$

Although lattice energy is positive in this definition it is not uncommon to see negative lattice energies, which are defined in the reverse sense to the above reaction. As lattice energy is not directly measurable by experiment, Hess's law is often used to describe the disassociation process in terms of several intermediate reactions, the enthalpies of which are either known or can be determined. This principle may also be used to calculate an unknown reaction energy in an otherwise known system, with the use of the Born-Haber cycle [29]. As an example, to compare the relative stability of two crystal lattices $AB_{(s)}$ and $AC_{(s)}$, a loop may be constructed (Figure 2.2). Here, the components $C^{-}_{(g)}$ and $B^{-}_{(g)}$ are assumed to be isolated from everything else so that they contribute nothing to $\triangle E$. If the enthalpies of formation of the constituents, and the enthalpy change for some intermediate reaction paths (x and y) are known, then the quantity $\triangle$E can be determined.

In this work, all structures of interest are ionic compounds, with the lattice energy therefore dominated by the monopole interaction terms. The summation of the Coulombic interaction of the i^{th} species with all other species can be written as,

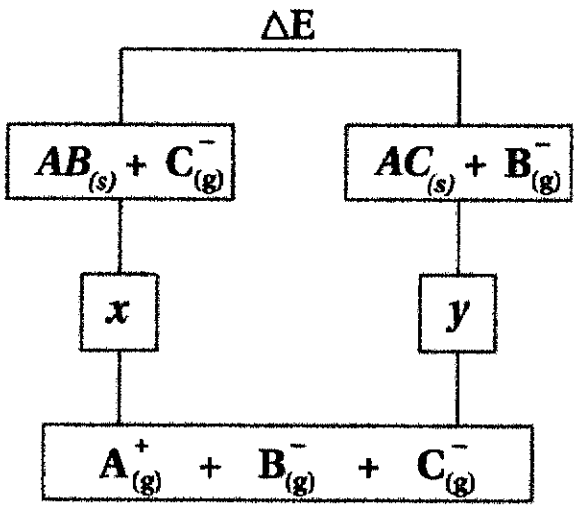

Figure 2.2: Simple Born-Haber cycle, with x and y representing known reaction pathways.

$$E_i^{mono} = \sum_j \frac{Z_i Z_j e^2}{r_{ij}} \tag{2.33}$$

With Ze being the charge of the species. In some cases, the symmetry of the lattice allows the total monopole contribution to the lattice energy to be written as,

$$\frac{1}{2}\sum_i E_i^{mono} = M\frac{Z^+Z^-e^2}{r_0} \tag{2.34}$$

where M represents an infinite summation of terms that converges to the Madelung constant for that particular ionic arrangement. Z^+e and Z^-e are the opposing charged species, and r_0 the minimum lattice separation. In more complex cases, the evaluation of the monopole energy must be performed explicitly using Equation 2.33. However, there is a difficulty in that the summation is slowly and conditionally convergent. The Ewald summation technique [117], which involves performing a partial transformation into reciprocal space, is a common method employed to accelerate the convergence.

The non-Coulombic interactions acting upon the i^{th} atom, usually determined by the potential model, can be written as the summation of the n-body terms,

$$E_i^{non-Coulombic} = \sum_i \phi_{ij}(\vec{r}_i, \vec{r}_j) + \sum_{j,k} \phi_{ijk}(\vec{r}_i, \vec{r}_j, \vec{r}_k) + ... \tag{2.35}$$

To calculate the net lattice energy it is necessary to sum over the entire number of atoms, with multiplication by factors of $\frac{1}{2}$ for two-body terms ($\frac{1}{6}$ for three body, and so on) to compensate for duplicate counting,

$$E^{tot} = \frac{1}{2}\sum_i \left(E_i^{mono} + E_i^{non-Coulombic}\right) \tag{2.36}$$

In most of the simulations in this work, the static lattice approximation is employed. This means that, unless otherwise stated, all calculations are performed with no inclusion of thermal effects.

Lattice Vibrations

Lattice types were discussed in section 1.1.1 these are also called the direct lattices. The periodic nature of crystalline materials means that analysis in Fourier space is common. It is therefore more convenient to work in reciprocal space. The unit cell vectors ($\vec{a}$, $\vec{b}$, $\vec{c}$) may be mapped from real space to reciprocal space using the following transformation [1],

$$\vec{k}_1 = 2\pi\frac{\vec{b}\times\vec{c}}{\vec{a}\cdot\vec{b}\times\vec{c}} \qquad \vec{k}_2 = 2\pi\frac{\vec{c}\times\vec{a}}{\vec{a}\cdot\vec{b}\times\vec{c}} \qquad \vec{k}_3 = 2\pi\frac{\vec{a}\times\vec{b}}{\vec{a}\cdot\vec{b}\times\vec{c}} \tag{2.37}$$

The vibrations in a crystal lattice may be treated as oscillations arising from the motions of atoms. As such, a quantum treatment of these oscillators shows that vibrations are discrete, with elastic energy levels $E = (n + \frac{1}{2})\hbar\omega$ [185]. Here, ω is the angular frequency and n is the quantum number of the vibrational mode. In a crystal, the wave vector $\vec{k}$ (corresponding to the mode of vibration) must satisfy the periodic boundary conditions. For a cubic lattice of volume $V = L^3$, the number of modes at a given frequency (ω) is given by the density of states,

$$D(\omega) = \left(\frac{L}{2\pi}\right)^3 \int_\omega d^3k \tag{2.38}$$

where the integral is over the surface of constant ω in Fourier or reciprocal space [186]. For a given potential model describing the interactions in a crystal lattice, the density of states function can be computed for a range of frequencies. The resulting phonon spectrum can then be compared to experimental vibrational data to verify the atomic and molecular interaction terms in the model.

2.4 Surfaces and Morphology

2.4.1 Morphology Prediction

A surface may be modelled as a stack of interacting layers which are classified as belonging to one of two different region types. The uppermost layers are designated as region 1, and here the atoms are allowed to move until the net force acting upon them converges to zero. Region 2 comprises the layers beneath region 1, and here all the atoms are held in fixed positions. Thus, when both region types are constructed with sufficient depth, region 2 simulates the effect of the bulk crystal on the surface layers.

Extending this concept of constructing a surface from layers, it is useful to define the attachment energy of a plane (hkl),

$$E_{att}^{hkl} = \sum_{i=1}^{\infty} E_i \tag{2.39}$$

with E_i being the interaction energy (per molecule) between a growth slice of thickness d_{hkl} and the i^{th} layer [141]. Thus, the attachment energy corresponds to the energy released when a complete growth slice precipitates on the crystal surface. For each face, the higher the absolute value of the corresponding E_{att} the faster this surface will grow. By ranking each face according to the inverse of its growth rate, the growth morphology may be constructed [27].

The surface energy for a plane (hkl) can be written as:

$$E_{surf}^{hkl} = \sum_{i=1}^{n} \frac{E_{slice}^{i} - E_{bulk}}{A^{hkl}} \tag{2.40}$$

where E_{slice}^{i} is the energy of the i^{th} slice in region 1, E_{bulk} is the energy of an equivalent slice in the bulk and A^{hkl} is the surface area. The equilibrium morphology is the habit a crystal of given volume takes according to the minimization of its net surface free energy [28]. The surface energy is typically used in place of the surface free energy.

The visualization of a morphology prediction is typically accomplished by means of the Wulff construction [187], which assumes that the shortest distance from the centre of a crystal to each of its bounding facets is proportional to the rate of growth (or surface energy) of that face.

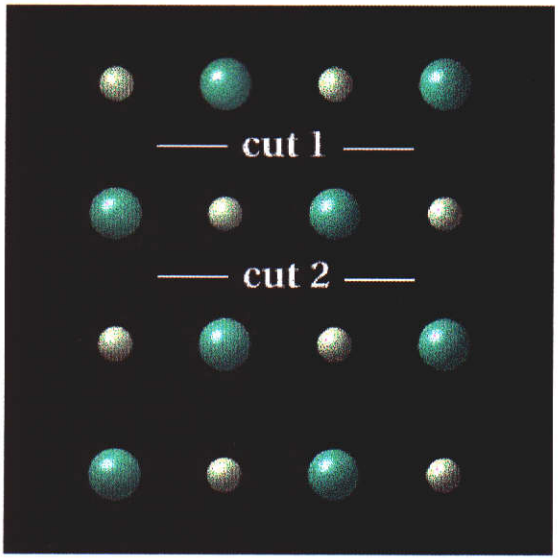

Figure 2.3: Type I surface, all possible cuts are valid.

2.4.2 Surface Types

The most common means to generate a surface is to cleave the bulk using a planar cut, defined by the Miller index (hkl) of that face. However, the Miller index alone cannot be used to define a crystal surface. This is because there may be more than one unique surface configuration depending upon the perpendicular location of the cleavage plane. Thus, in order to completely define a surface, a shift parameter must also be specified. This is commonly written as a fraction of the d_{hkl} value. The choice of shifts is limited to those which do not have a dipole moment normal to the newly created surface, as discussed below.

For ionic materials, there are three different types of faces. These surfaces may be classified according to the complexity of the layer structure. Type I faces are the simplest, the arrangement of ions means that no matter where the bulk is cleaved the surface will not be polar (see Figure 2.3). However, some surfaces can be polar or non-polar depending on where the bulk is cleaved, these are the type II faces (Figure 2.4). A type III surface is composed of alternate, evenly spaced layers of oppositely charged species (Figure 2.5). This results in a polar surface no matter where the bulk is cleaved. It can be shown that a surface with a perpendicular dipole moment has an infinite surface energy [188], which is not physically realistic. This does not mean that these surfaces do not occur in nature, but that some neutralization mechanism is necessary. For a computer simulation, polar surfaces are usually neutralized by mechanical means.

A common technique to accomplish this neutralization can be demonstrated by writing a type III surface as a stack of $2n$ alternating layers of oppositely

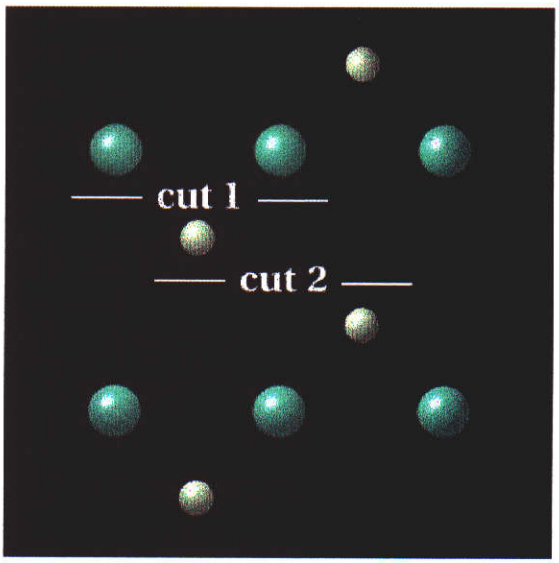

Figure 2.4: Type II surface, cut 1 is invalid, but cut 2 is valid.

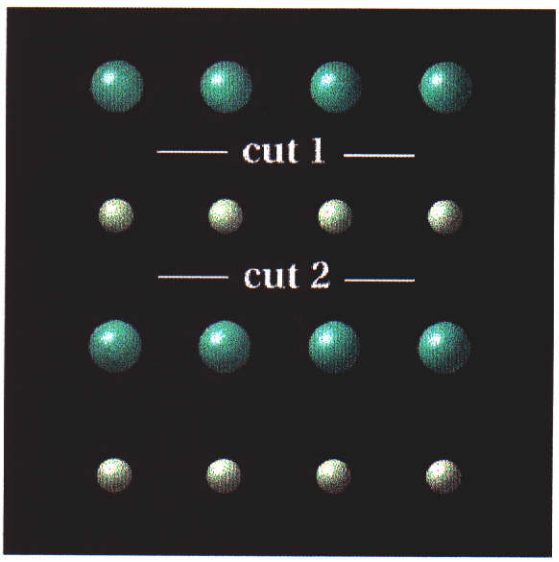

Figure 2.5: Type III surface, no cuts are valid.

charged species (A and B). Let the distance between layers be $\triangle z$. As the $2n$ layers must have a net charge of zero, the total charge on a row of A (Q_A) is equal and opposite to the total charge on a row of B (Q_B) so that $Q_A = -Q_B$. Also, the dipole moment is independent of origin, and will be chosen to coincide with the bottom of the surface region, which is the first row of A. Thus, the contribution to the dipole moment perpendicular to the surface by all A atoms is given by,

$$\begin{aligned} \overrightarrow{D}_A.\hat{z} &= \textstyle\sum_{i=1}^{n} Q_A(i-1)\triangle z \\ &= Q_A\triangle z \textstyle\sum_{i=1}^{n}(i-1) \end{aligned} \tag{2.41}$$

while the contribution from all B atoms is,

$$\begin{aligned} \overrightarrow{D}_B.\hat{z} &= \textstyle\sum_{i=1}^{n} Q_B[(i-\frac{1}{2})\triangle z] \\ &= Q_B\frac{n\triangle z}{2} + Q_B\triangle z \textstyle\sum_{i=1}^{n}(i-1) \\ &= Q_B\frac{n\triangle z}{2} - \overrightarrow{D}_A \cdot \hat{z} \end{aligned} \tag{2.42}$$

with the total perpendicular dipole moment,

$$\overrightarrow{D}.\hat{z} = \overrightarrow{D}_A.\hat{z} + \overrightarrow{D}_B.\hat{z} \tag{2.43}$$

Comparing the result of 2.42 to equation 2.43 it is apparent that the total perpendicular dipole moment can never be zero despite the choice of Q_A, Q_B, n or $\triangle z$, as the contribution from layers of B is too large by $Q_B\frac{n\triangle z}{2}$. However, if half of the uppermost (n^{th}) layer of B were removed, then the dipole would be reduced by a factor of $\frac{1}{2}Q_B(n-\frac{1}{2})\triangle z$, which is close to what is desired. Since the net charge must be conserved, this half layer can only be relocated elsewhere in the system and not destroyed. If the new location is some layer j then the contribution to the perpendicular dipole moment from all B becomes,

$$\begin{aligned} \overrightarrow{D}_B.\hat{z} &= Q_B\triangle z\,(\tfrac{n}{2} - \tfrac{1}{2}(n-\tfrac{1}{2}) + \tfrac{1}{2}(j-\tfrac{1}{2})) + Q_B\triangle z \textstyle\sum_{i=1}^{n}(i-1) \\ &= Q_B\triangle z\,(\tfrac{j}{2}) + Q_B\triangle z \textstyle\sum_{i=1}^{n}(i-1) \end{aligned} \tag{2.44}$$

Hence, setting $\overrightarrow{D}.\hat{z} = 0$ requires that $j = 0$. Thus the half plane of species from the top layer should be moved to the layer directly below row 1, which is the bottom of the surface region.

2.5 Dynamics

One of the earliest applications of computational chemistry was the simulation of a simple liquid composed of hard spheres [115]. Today, computing power has increased to the point where it is possible to simulate the behaviour of solutions containing thousands of atoms and molecules. Although this represents only a very small sample of liquid, the use of repeated images (periodic boundary conditions) can approximate the effect of a surrounding bulk liquid on a smaller explicit simulation cell. Techniques in setting up and running a dynamics simulation differ in several ways from the solid and surface modelling cases discussed previously. Most importantly, the rapid motions and complex structuring of components in a fluid necessitate additional care when setting up the simulation. This is to ensure that any data extraction, which will be statistical in nature, yields an unbiased and representative sample of 'true' behaviour.

2.5.1 Initialization

Before any liquid simulation and subsequent data extraction can be performed, the initial placements of all the atomic and molecular solution species must be assigned. Typically, no data is available in order to generate some reasonable initial system state. This means an improbable choice of starting position may trap the simulation and limit the number of solution configurations that could possibly occur. This is generally avoided by performing multiple simulations from different starting configurations, and comparing the results. Algorithms for creating an initial configuration are usually variants of the lattice approach. Here components are placed at evenly spaced lattice points in a box. The lattice separation is chosen to avoid overlap between neighbouring molecules. Clearly, this is an unrealistic state for any liquid and a period of equilibration is needed, the length of which is dependent on two chief aspects. Firstly, the ordering in the initialization lattice must have vanished, and secondly quantities such as the pressure and potential energy should have reached some steady state value [115].

2.5.2 Equations of Motion

Dynamics is principally concerned with how the positions, velocity and acceleration of a group of N particles changes, given some initial state. Thus, the problem can be reduced to solving for $x_i^{(j)}(t + \triangle t)$ given $x_i^{(j)}(t)$, with $i = 1...N$ and the derivatives j. Many such methods are built around the Taylor expansion [182],

$$x(t+\triangle t) = x(t) + \frac{\triangle t}{1!}x'(t) + \frac{\triangle t^2}{2!}x''(t) + ... \tag{2.45}$$

An example of such a numerical method is the Runge-Kutta technique, which is very robust and often used as a starter for the faster but more sensitive algorithms such as the Adams-Bashforth-Moulton predictor-corrector [153] or the Gear method [189]. These latter methods are more efficient as they employ extrapolation techniques using previous data points. However, starter methods are still required in order to supply the necessary information for initializing the first cycle of the more complex methods. In addition, the robust starting integrator may also be useful when the functions that are to be integrated are rapidly varying and cause numerical instability in the predictor-corrector cycle [182].

An important consideration in any simulation is the choice of step size, which is a compromise between accuracy and speed. Typically, a step size that is approximately a tenth of the period of the smallest event is chosen; giving a resolution of around 10 data points. For flexible molecules, a time step of about 1 fs is frequently used [114]. However, this is complicated by the fact that the events requiring the small time scale may be quite rare; such as molecular collisions in a low density environment. This can lead to long and inefficient simulations, for which the step size cannot be shortened without invalidating the results. Some methods, such as the Runge-Kutta-Fehlberg method include techniques to adjust the step size when necessary [182].

In the case of molecular systems, some coordinates are redundant, due to atoms that are connected to each other. For example, instead of 3N coordinates describing a rigid molecule that has fixed bond lengths, only 3 coordinates describing the position of the center of mass and 2 describing orientation are needed. Thus, a transformation to generalized coordinates q_i is performed to eliminate the dependent coordinates,

$$q_i = f(x_1, ..., x_n) \tag{2.46}$$

which also have generalized associated forces Q_i,

$$Q_i = \sum_{j=1}^{n} F\frac{dx_j}{dq_i} \tag{2.47}$$

Classical mechanics employs the Lagrangian equation [190] in order to formulate a system of simultaneous equations,

$$\frac{d}{dt}\left(\frac{\partial L}{\partial q_i'}\right) - \left(\frac{\partial L}{\partial q_i}\right) = 0 \tag{2.48}$$

with q_i' the associated time derivative of q_i, and L is the Lagrangian function defined as the kinetic energy minus the potential energy, $L = K - U$.

2.5.3 Simulation Sampling

After equilibration, measurements of some quantity of interest are typically made at periodic intervals. Clearly, there are an infinite number of possible states $\Phi(t)$ for a system comprising N particles, each with an associated position, velocity and acceleration vector. As any observation is entirely dependent upon the particular state the N particles are currently in, it is usual to determine the time average of the desired observable R,

$$R = \lim_{T\to\infty} \frac{1}{T} \int_{t=0}^{T} \Re[\Phi(t)]dt \tag{2.49}$$

Where $\Re$ is the functional corresponding to the observable of interest. In practice, sampling is performed from a group, or ensemble, that is representative of the external conditions. For example, almost all simulations impose the restriction that the number of particles must remain constant. Ensemble types are denoted by whichever quantities are held fixed. Some of the most common ensembles used are: NVT, NVE, NPT and the NPH, with N the number of particles in the system, V the volume, T the temperature, E the energy, P the pressure and H the enthalpy.

Statistical mechanics allows certain properties to be calculated from the partition function Q (also called the sum over states), such as the internal energy,

$$U = kT^2 \left(\frac{\partial lnQ}{\partial T}\right)_V \tag{2.50}$$

the enthalpy,

$$H = kT^2 \left(\frac{\partial lnQ}{\partial T}\right)_T + kTV \left(\frac{\partial lnQ}{\partial V}\right)_T \tag{2.51}$$

and the Gibbs free energy,

$$G = -kTlnQ + kTV \left(\frac{\partial lnQ}{\partial V}\right)_T \tag{2.52}$$

These results represent what a real experiment would be expected to yield under

the same ensemble conditions. However, one of the basic principles of statistical mechanics, the ergodic hypothesis, states that the ensemble average is the same as the time average [114]. As equation 2.49 is readily evaluated, it is usual to compute the time average of some quantity in a simulation and assume that this approaches the ensemble average as the sampling time is increased.

2.6 Computer Software

2.6.1 Computational

All computer codes were run on UNIX machines. Primarily this included several Silicon Graphics workstations, and also some Intel Linux machines. *Ab initio* Hartree Fock calculations were performed using the CADPAC code from Cambridge [191], while density functional calculations involved the DMOL package from MSI [192]. Both these codes were only executable on Silicon Graphics machines. The semi empirical code employed was MOPAC [193], which was run on both Silicon Graphics and Linux machines. Dynamics studies were accomplished on the Silicon Graphics machines, using the DISCOVER95 code from MSI. Whilst there are a host of free Molecular Dynamics codes that run on Linux platforms [193], many do not have the flexibility and extended range of features found in the commercial MSI package. Solid state molecular mechanics calculations were performed on both Intel and Silicon Graphics machines with GULP [194] and MARVIN [141] programs. With the exception of the MSI and CADPAC codes, all the above packages are freely available for academic use.

2.6.2 Data Analysis

The computations employed in this work often yielded large amounts of raw data that required some processing for visualization and interpretation. The Perl (practical extraction and report language [195]) proved to be a useful tool in parsing and processing large data files. Typically, codes were written in order to generate information that could be easily displayed in (for example) a typical spreadsheet program. Perl may be regarded as offering a sophisticated array of text parsing operators, useful in automating the process of sifting through large numbers of data files for only a few values. It also offers adequate mathematical manipulation tools, albeit with interpreted speed performance. As Perl may be interfaced with C, the relative slowness of Perl under computationally demanding conditions is not a major problem. In addition, there are Perl compilers available,

however they are relatively new and at present not quite as stable as their interpretive counterparts. Perl is freely available and runs on a variety of platforms, including Macintosh, MSDOS, OS/2, VMS and various UNIX platforms. While Perl's control structures are very similar to those in C, it is much more suited to manipulating strings containing both text and numerical data. Most of the code written as part of this project was in Perl, with one important exception, the molecular dynamics initialization (MDI) program, where the higher speed of compiled C code in dealing with mathematical manipulation of arrays was thought to be important. These codes are found in the Appendix, and described in more detail in later sections.

2.6.3 Visualization

All images created for this work were made on a Pentium 200MHz machine, running RedHat Linux 5.0 [196]. A Perl script was written to employ the POVRAY [197] rendering code to generate the molecular images, whilst the GIMP [198] paint program was utilized to create all other pictures. Morphology visualization was accomplished with the JSHAPE Java code [199]. All these packages are freely available.

Chapter 3

Empirical Potential Fitting

3.1 Hydroxide Compounds

3.1.1 Background

Some important areas of modelling that are applicable to the crystal growth industry include, morphology predictions, impurity effects and solution structuring. For many such studies, the relevant processes occur on far too macroscopic a level to involve quantum mechanics as a predictive tool. The development of a simplified model for gibbsite and related compounds is therefore of primary importance. Throughout this project, the construction of an accurate interatomic potential model for gibbsite has occupied a substantial portion of the total effort. Much of the work involved feed-back stages of model development, followed by subsequent correction as a result of experimental input. The final gibbsite model is presented in this chapter.

In the literature, three prior efforts to develop a potential model for gibbsite exist. The model by Baram and Parker [200] was derived from earlier studies by Lewis and Catlow [127] and was primarily constructed to examine hydroxyl defects at silicate surfaces and within the bulk. However, it was found that the model works quite well for gibbsite and bayerite, but in both cases the c lattice parameter is significantly larger than the experimental crystal structure. Molecular dynamics studies by Teppen *et al.* [137] were accomplished with a gibbsite potential that reproduces the lattice constants to a high degree of accuracy. However, the authors indicate that the hydrogen bonding pattern between the layers deviates significantly from the experimental gibbsite crystal structure. This produces a monoclinic angle that is too large. Lee *et al.* [152] also developed a potential set for gibbsite that was found to be sufficient for qualitative morpho-

logical studies. However, more detailed work with this model is not feasible as it produces a significant error in the monoclinic angle.

3.1.2 Method

As the potential models in the literature all fail to accurately reproduce the lattice parameters of gibbsite, a new model was developed. In addition to gibbsite and bayerite, the crystal structures of sodium, potassium, calcium and magnesium hydroxides were also fitted. These structures provided a means to test the robustness of the hydroxide model developed for the two polymorphs of $Al(OH)_3$. In order to fit these structures, the use of a formal charge for aluminium was mandatory. This follows from the fact that it is difficult to transfer potentials between ionic metal hydroxides if the net charge on the OH group is not -1. However, the usage of partial charges within the hydroxyl group was adopted by necessity, as the hydrogen bonding interactions are quite complex in the $Al(OH)_3$ polymorphs.

3.1.3 Results

Initially, the potential model was developed in order to reproduce the gibbsite and bayerite crystal lattices, and their respective hydroxide stretching frequencies. Presented in Table 3.1 are the potentials developed for gibbsite and bayerite. Note that the Morse bonding term in the hydroxide ion acts between both the oxygen and hydrogen cores, whereas all other oxygen potentials (including the oxygen-hydrogen intermolecular Buckingham term) have the conventional behaviour of acting upon the shell. This approach was found to give better results when attempting to maintain hydroxide bond lengths and stretching frequencies. Also, as maximum transferability of the model is desirable, an aluminium oxy-hydroxide structure (diaspore) was also fitted.

Due to the choice of charge model, it was relatively straightforward to extend the $Al(OH)_3$ potential model to alkali metal hydroxides. The hydroxide terms and charges have been taken from the above results and directly substituted into the additional structures, so that only the cation terms were fitted. The final potentials are shown in Table 3.2. Cation to hydrogen potentials were found to be necessary for this model. This is most likely a compensation due to the partial charge model used for the hydrogen, resulting in a lower Coulombic repulsion with the cationic species than actually occurs. Cation to cation potentials were in general not required, with sufficient repulsion experienced due to the formal

Atom types	Interaction parameters		
	Buckingham		
	A (eV)	ρ (Å)	C (eV Å^6)
Al-O	1342.86	0.2944	0.0
O-O	9999.97	0.1490	17.0
O-H	235.00	0.2500	0.0
	Morse		
	D_e (eV)	α (Å^{-1})	r_0 (Å)
O-H	5.4246	2.2682	0.95
	Spring constant k (eV rad^{-2})		
O	60.1		
	Coulombic q		
Al core	3.000		
O shell	-2.366		
O core	0.948		
H core	0.418		

Table 3.1: Potential parameters from the gibbsite and bayerite dual fitting results.

Atom types	Buckingham parameters		
	A (eV)	ρ (Å)	C (eV.Å^6)
Na-O	939.77	0.2978	0.0
Na-H	244.06	0.2843	0.0
Mg-O	967.90	0.3017	0.0
Mg-H	480.42	0.3048	0.0
K-O	3209.90	0.2810	0.0
K-K	738.33	0.3867	0.0
K-H	404.67	0.2041	0.0
Ca-O	1356.84	0.3208	0.0
Ca-H	323.44	0.3016	0.0

Table 3.2: Buckingham potential parameters for other hydroxides.

	experimental values				calculated values			
	a (Å)	b (Å)	c (Å)	β (°)	a (Å)	b (Å)	c (Å)	β (°)
gibbsite	8.68	5.08	9.74	94.54	8.81	4.99	9.79	95.95
bayerite	5.06	8.67	9.42	90.26	5.06	8.81	9.49	90.97
diaspore	4.40	9.42	2.84	-	4.34	9.38	2.92	-
NaOH	3.40	11.38	3.40	-	3.43	11.40	3.36	-
$Mg(OH)_2$	3.13	3.13	4.71	-	3.10	3.10	4.76	-
KOH	5.89	3.94	7.72	110.30	5.97	3.95	7.65	110.60
$Ca(OH)_2$	3.59	3.59	4.91	-	3.57	3.57	4.94	-

Table 3.3: Error in computed lattice parameters using the common hydroxide model. Values not shown are zero by virtue of lattice symmetry.

	experimental value (cm^{-1})	calculated value (cm^{-1})
gibbsite	3617	3576
bayerite	3534	3579

Table 3.4: Hydroxide stretching frequencies (cm^{-1}).

charge model. The one exception was potassium, a fact which may be attributed to the larger radius of the ion.

These potentials reproduce the lattice parameters of the fitted crystal structures to a high degree of accuracy. In Table 3.3, the computed lattice parameters are compared to the experimental values. In almost all cases, the error is no more than 2% of the measured values. In addition, the deviations in lattice parameters were generally sought to be of both positive and negative sign; as this aids in maintaining the unit cell volume. Of considerable importance is the ability of the gibbsite model to maintain the monoclinic angle at nearly the correct value. As previously mentioned, the complexity of the interlayer hydrogen bonding has previously led to considerable difficulty in preventing excessive sliding of the double OH layers, which in turn causes an artificial increase in β. The calculated frequencies of the hydroxide stretching in gibbsite and bayerite were also computed and compared to experimentally measured values taken from [201]. These are shown in Table 3.4. Although the frequencies are not an exact match, they are of similar magnitude.

3.1.4 Discussion

The initial parameters for the gibbsite fitting were taken from gas phase calculations between pairs of atoms. *Ab initio* Hartree-Fock calculations of the energy at various separations of AL-O, O-O and O-H atom pairs were performed. Several

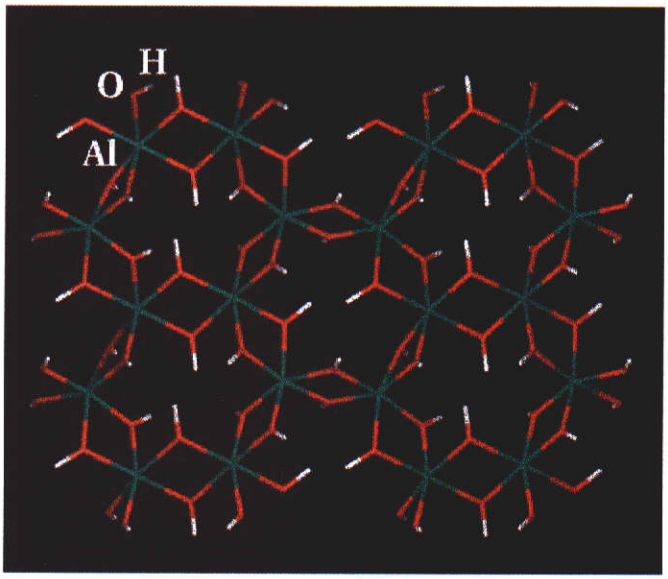

Figure 3.1: The calculated bulk structure of gibbsite, viewed down the c axis.

basis sets were examined, with the final results taken from the 6-31G** basis set with MP2 correction. These values were fitted using the GULP program in order to make a first guess at suitable parameters for the potential model. It was expected, due to the simplicity of the approach, that these values would not closely resemble the actual pairwise interactions in the gibbsite lattice. Consequently, significant changes were introduced to the potential model during the fitting process.

To examine how well the model reproduces the structure of gibbsite, the final optimized configuration was examined. The generated crystal structure is shown in Figure 3.1, whilst the experimental structure is given in Figure 3.2. The aluminium and oxygen pseudo-hexagonal framework has been reproduced quite well. The only significant deviation is the slight distortion of the pseudo-hexagonal rings. This has some impact on later studies involving cleavage of the bulk structure, particularly on planes that are orthogonal to the basal (002) face.

The O-H bond distances in the experimental gibbsite structure (with quantum mechanically derived proton positions) were compared with the computed structure, as shown in Table 3.5. Clearly, the calculated values are all slightly too high. However, tightening the Morse potential to compensate for this deficiency has the consequence of adversely affecting the stretching frequencies as well. A similar form of behaviour also occurs in the calculated lattice parameters. That is, adjusting a potential parameter to improve one lattice constant will increase the error in another. This suggests that a local minimum in the total deviation (sum of squares) has been achieved. Further refinement may be possible,

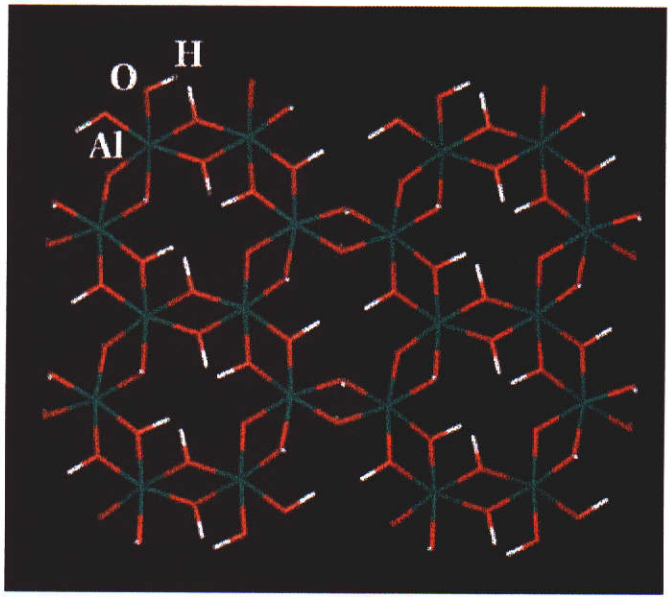

Figure 3.2: The experimental bulk structure of gibbsite, viewed down the c axis.

Hydroxide bond distances	
experimental (Å)	calculated (Å)
0.9674	0.9927
0.9755	0.9976
0.9782	0.9969
0.9740	0.9986
0.9798	0.9940
0.9826	0.9976

Table 3.5: Bond length comparisons for hydroxyl groups in gibbsite. The experimental values were taken from Saalfeld's structure determination [50], with quantum mechanically corrected proton positions calculated by Gale [52].

although it is likely that the goodness of fit is close to the achievable limit, given the classical interaction model used.

3.2 Oxalate Compounds

3.2.1 Background

Although a principal aim of this project was the development of a potential model for gibbsite, other compounds are of interest due to their effect on gibbsite precipitation in the Bayer process. Sodium oxalate can coprecipitate with gibbsite in alumina refining, degrading the product and resulting in higher production of undesirable fine material. To avoid this problem the refineries are run at far from optimum conditions, reducing alumina yield. A separate oxalate removal process is required, representing a huge cost in lost production and increased refining costs for the alumina industry [202]. In addition to the industrial importance of sodium oxalate, latter studies were conducted involving a successful predictive model utilizing both gibbsite and oxalate potentials (Section 4.3). The theme of transferability and parallel development is a desirable step in the direction of future effort. Hence, although oxalate and gibbsite models were not directly used in a joint simulation, this will be attempted in the future.

Consistent with the principle of fitting to as many observables as possible, the sodium oxalate potential model was developed in order to reproduce the crystal structures of calcium and potassium oxalates as well. Oxalate salts are also of great biological, medical and industrial importance. Calcium oxalate is the major component of more than 70% of kidney stones [203]. In 1981, 1% of all hospital admissions in the USA were for kidney stone problems. This represents an enormous health care cost and extensive studies into calcium oxalate crystallization and its inhibition have been conducted. An understanding of all aspects of the underlying mechanisms of stone formation would provide invaluable information in efforts to prevent its occurrence [204].

The generation of an accurate, transferable computer model of the oxalate species and its interaction with metal ions is a vital step to modelling crystal growth in different environments for many different oxalate salts. Modelling the growth mechanisms of oxalate salts and how they are modified in the presence of foreign molecules and ions will lead to the design of novel crystal growth modifiers. The technique of generating interatomic potentials from high quality *ab initio* calculations rather than fitting to experimental data is becoming more

widely utilized. In recent years a lot of work has been conducted on zeolites in an attempt to predict the structure and properties of as yet unsynthesized zeolites. Ermoskin *et al.* [205] used *ab initio* calculations to derive a potential for hydroxyl groups. Hill and Sauer [206] derived a consistent force field for the simulation of protonated alumina silicates on the basis of *ab initio* calculations. Hartree-Fock calculations have been used to obtain potential parameters for alumina which are superior to empirical parameterizations [207].

Derivation of potentials for organic molecules has also been a major field of study. The structure of tri-tert-butylmethane was finally resolved with the aid of quantum mechanically derived potentials [208]. Development of an *ab initio* force field for polycarbonates occurred after preliminary quantum mechanical calculations on several smaller compounds: carbonic acid, methyl and dimethyl carbonates, phenyl carbonate and 2,2-diphenylpropane [209]. Development of a force field from the *ab initio* potential energy surfaces has been used for molecular simulations of alkanes [210].

3.2.2 Method

The problems encountered in modelling molecular monoanions are well documented [211]. However there have been few publications addressing the inherent problems in modelling molecular dianions [212]. During the investigation of the oxalate dianion ($[C_2O_4]^{2-}$), it was found that the techniques used for monoanions failed to produce a stable oxalate dianion. The principal difficulty lies in stabilizing the excess negative charge on the ion. In oxalate compounds, this instability is compensated for by the presence of a surrounding lattice. It has been suggested that a point charge array can be utilized to mimic the electrostatic influence of the crystal lattice [106]. The choice of point charge configuration is therefore made so that the molecular geometry approximately matches that in the observed crystal structures. In addition, careful examination of the orbital energy levels was required to monitor the excess negative charge on the oxalate dianion.

It was decided to employ several quantum mechanical models to study the energetics of the oxalate dianion. Consequently, an assessment of the most suitable method for intramolecular potential generation was made. Three quantum mechanical techniques were used to reproduce the geometry of the oxalate dianion. These were the Hartree-Fock, density functional and semi-empirical approaches. The respective packages employed were CADPAC, DMOL, and MOPAC. It was decided to investigate the 6-31G basis set for the Hartree-Fock, and the corre-

Structure	C-C Bondlength (Å)	C-O Bondlength (Å)	Reference
sodium oxalate	1.568	1.253/1.265	[213]
potassium oxalate	1.581	1.256/1.262	[214]
calcium oxalate	1.563	1.254/1.257	[215]
averaged oxalate	1.564	1.251	[216]

Table 3.6: Experimental geometry of oxalate compounds.

sponding DN basis set for density functional techniques. In addition, the effect of including polarization functions and MP2 correlation correction was investigated. Diffuse basis sets were also examined, but discarded due to poor oxalate geometry. However, diffuse functions were better at accommodating the excess negative charge on the oxalate ion as would be expected.

After the determination of a suitable point charge array, the generation of energy hypersurfaces for the oxalate dianion was accomplished. Molecular distortions, including bond stretching, angle bending and molecular twisting (torsion) were performed in order to construct a suitable set of intramolecular potentials. These were then transferred as fixed potentials into further fitting of several oxalate structures, namely; sodium oxalate, potassium oxalate monohydrate and calcium oxalate monohydrate. The quality of the resulting interatomic potentials was assessed on the basis of how well they reproduced the crystal structures.

3.2.3 Results

Vacuum Geometries

Experimental oxalate geometries were examined in order to establish an estimation of the desired quantum mechanical geometries. The experimentally determined bond lengths for the oxalate dianion in sodium oxalate, potassium oxalate monohydrate, calcium oxalate monohydrate and as an average from nine crystal structures are given in Table 3.6. These data suggest that a typical carbon-carbon bond length might be expected to be 1.57Å, with the carbon-oxygen bond length around 1.25Å or 1.26Å.

Table 3.7 gives the oxalate geometries calculated using the three methods in vacuum. Both the Hartree-Fock and density functional methods overestimate the length of the carbon-carbon bond, whilst the semi-empirical method gives an underestimate. In addition, the carbon-oxygen bond length is overestimated, except for the Hartree-Fock calculations omitting the MP2 correction. As the errors are quite large for a quantum mechanical approach, this suggests that further effort is needed before intramolecular potentials describing the energetics

Method used	Basis set	C-C distance (Å)	C-O distance (Å)
Semi-empirical (AM1)		1.5405	1.2755
Hartree-Fock	6-31G	1.5817	1.2662
	6-31G MP2	1.6045	1.3106
	6-31G*	1.6032	1.2427
	6-31G* MP2	1.6072	1.2740
Density Functional	DN	1.5821	1.2906
	DNP	1.5855	1.2738

Table 3.7: Oxalate geometries, calculated *in vacuo*.

are sought.

In the absence of other charges, Coulomb repulsion between the ends of the oxalate dianion becomes the dominant force, causing this stretching of the molecular dianion. However, the worst problem is with the Highest Occupied Molecular Orbital (HOMO) energies. A previous study involving dianions [212] has shown electrons can be ejected from the HOMO. Our calculations of the energy levels of the oxalate dianion indicates a similar instability, as the HOMO is positive. Ejection of an electron will reduce the overall charge on the dianion and reduce the bond strain, increasing the stability in vacuum.

Point Charge Arrays

The oxalate dianion, like O^{2-}, is not stable in vacuum, but requires the stabilizing presence of a surrounding solution or crystal structure for molecular integrity. To simulate this a point charge array was placed around the oxalate dianion. The electrostatic potential experienced by an oxalate dianion in a crystal lattice was calculated using GULP. An infinite periodic sodium oxalate crystal with one missing oxalate dianion was constructed. The electrostatic topography in the region of the missing oxalate dianion was then calculated. The point charge array was constructed to qualitatively reproduce the electrostatic potential that an oxalate anion would experience in a crystal structure. The point charge array was altered manually until the HOMO energy was negative and the calculated bond lengths matched those observed experimentally. The electrostatic topography around the oxalate dianion in the point charge array was found to reasonably match that obtained using GULP in the crystal lattice.

The construction of the point charge array required the use of fractional point charges to give a sufficiently accurate array whilst balancing the double negative charge associated with the oxalate dianion. Unfortunately MOPAC, the semi-empirical package, does not have facilities for fractional point charges and

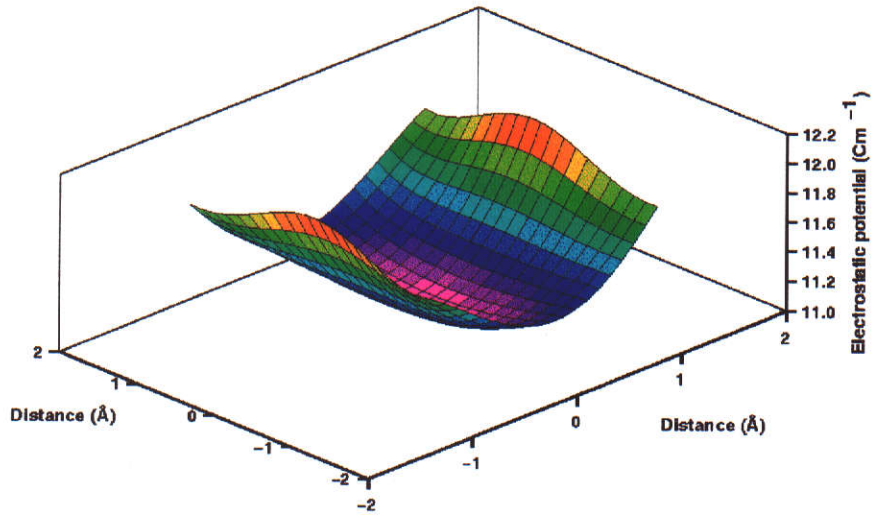

Figure 3.3: Crystalline electrostatic topography in the plane of a removed oxalate dianion (sodium oxalate).

therefore the point charge array could not be constructed using MOPAC. The electrostatic topography around a removed oxalate dianion in crystalline sodium oxalate is shown in Figure 3.3. In addition, the out of plane structure was examined by constructing contour maps at right angles to the above plane (Figures 3.4 and 3.5).

The geometry of the oxalate anion was recalculated in the presence of the point charge array using both Hartree-Fock and density functional methods. Six point charges, possessing the same symmetry configuration as the oxalate dianion, were situated as shown in Figure 3.6. To compare the effect of the point charges with a potential felt by oxalate in the sodium oxalate lattice, the electrostatic topography within the array was calculated. In Figure 3.7 the in-plane electrostatic field is shown, whilst Figures 3.8 and 3.9 illustrate the out-of-plane electrostatic variations. Note that the point charge array does not precisely mimic the calculated topography and contour maps in the sodium oxalate structure. However, this is desirable as the local electrostatic potential will change between different oxalate structures. Hence, precise reproduction of the electrostatic field would hinder the development of a transferable oxalate potential.

The distortion in contour maps of the 'real' system from the more symmetric 'idealized' electronic system results from additional oxalate species in the real lattice being sufficiently close to impact on the topography. This is difficult to simulate in the point array case, as charges that are too close may artificially remove some of the electronic charge density from the dianion. Indeed, this factor also contributes to the success of the smaller basis sets compared to the larger

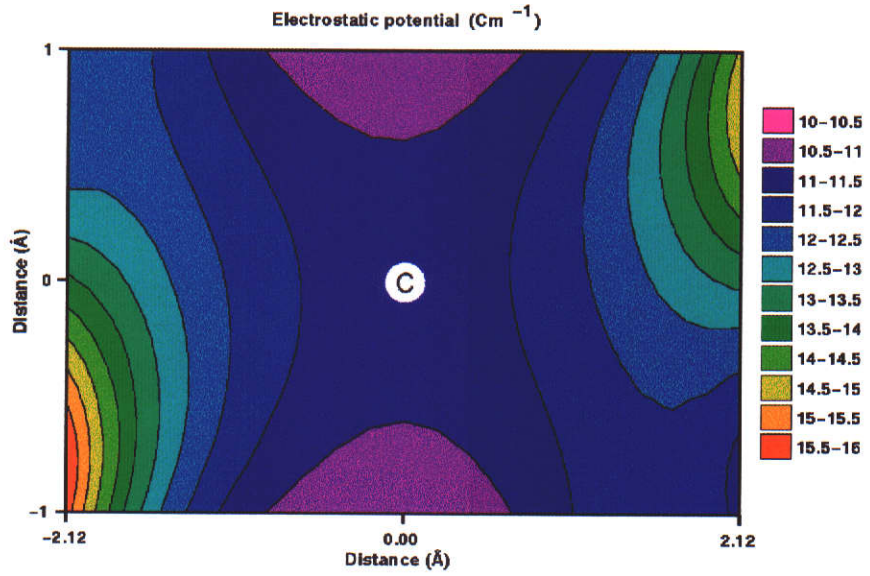

Figure 3.4: Electrostatic contour map at right angles to the plane of the oxalate site; sliced through a carbon atom.

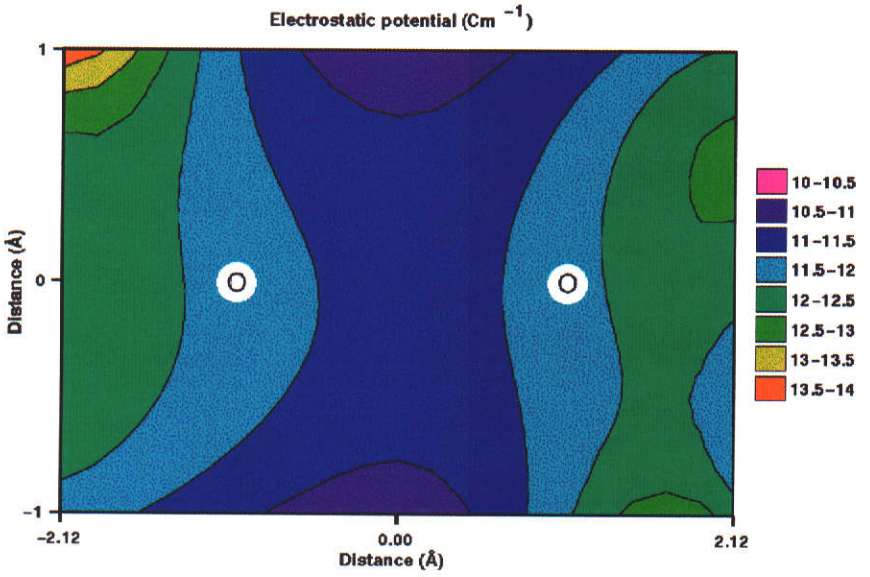

Figure 3.5: Electrostatic contour map at right angles to the plane of the oxalate site; sliced through two oxygen atoms (attached to the same carbon atom).

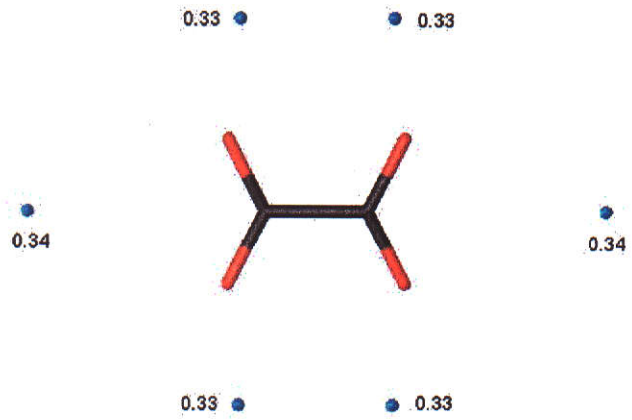

Figure 3.6: Oxalate dianion in a fractional point charge array.

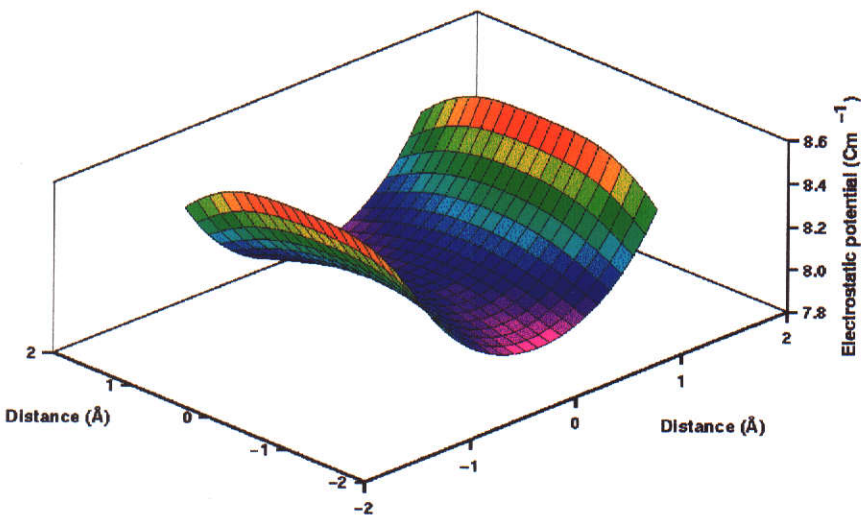

Figure 3.7: Electrostatic topography within the point charge array.

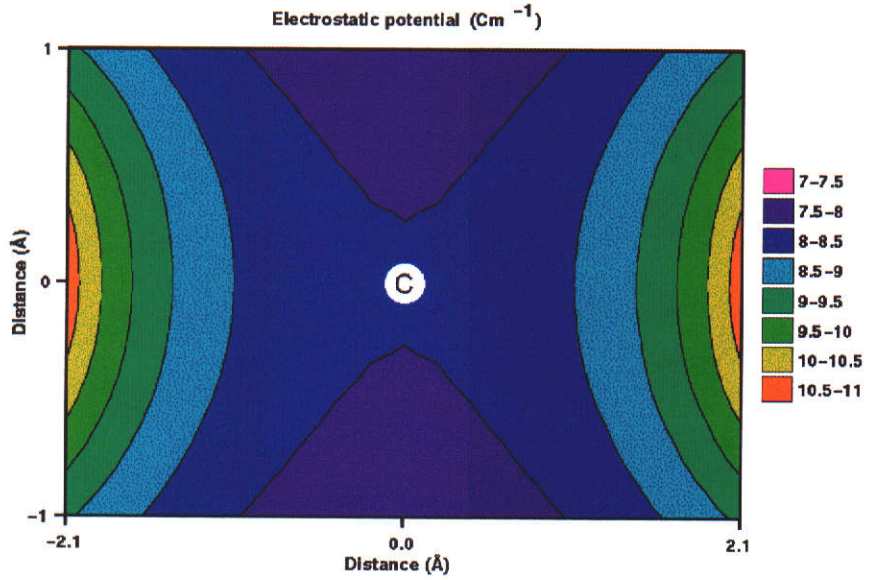

Figure 3.8: Point charge electrostatic contour map at right angles to the oxalate site; taken through the plane of a carbon atom.

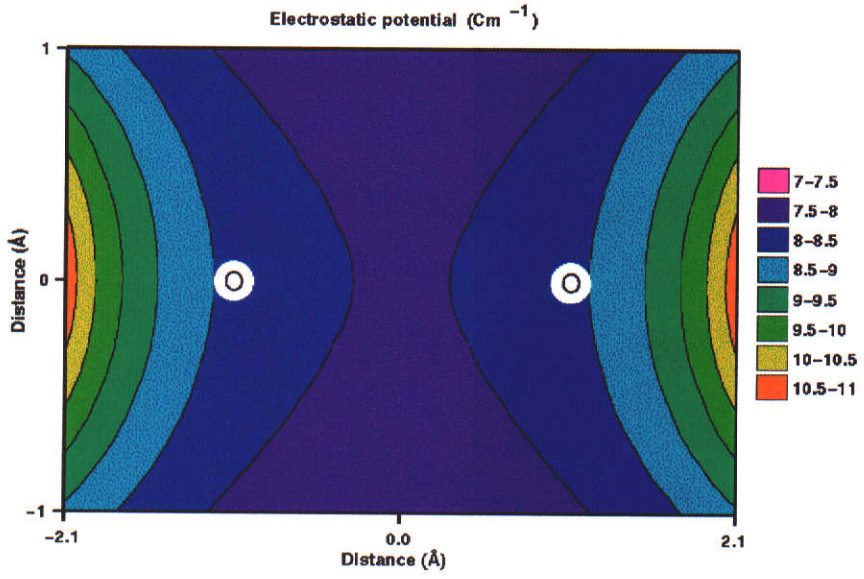

Figure 3.9: Point charge electrostatic contour map at right angles to the oxalate site; taken through the plane of two oxygen atoms (attached to the same carbon).

Method	Basis	C-C bondlength (Å)	C-O bondlength (Å)
HF *ab initio*	6-31G	1.5710	1.2434
	6-31G MP2	1.5975	1.2923
	6-31G*	1.5947	1.2271
	6-31G* MP2	1.6040	1.2602
Density functional	DN	1.5826	1.2905
	DNP	1.5903	1.2731

Table 3.8: Oxalate geometries in a point charge array.

more diffuse ones. The results from the minimization of the oxalate dianion in the point charge array are given in Table 3.8. In all cases the HOMO energy was found to be negative, indicating the formation of a stable molecular bond. The bond lengths calculated using the 6-31G basis set without the MP2 correction are close to those observed experimentally. The carbon-carbon bond length is only slightly longer and the carbon-oxygen bond length only slightly shorter than the average. The density functional bond lengths for the DN basis set are not as close, but are still in good agreement.

The 6-31G basis set without MP2 adjustment most accurately reproduced the experimental oxalate geometry. The 6-31G* basis set increased the carbon-carbon bond length and decreased the carbon-oxygen bond length farther from their respective experimental values. The MP2 correction worsened the calculated model, increasing both the carbon-carbon and carbon-oxygen bond lengths. For the density functional calculations, the DN basis set was more accurate for the C-C bond length, but the DNP basis set more accurate for the C-O bond length. The polarization functions failed to improve the calculated oxalate geometry. This may be due to the approximate nature of the electrostatic topography produced with the point charge array, and nearness of the point charges adversely affecting the polarization contributions.

Empirical Potential Derivation

Having calculated the optimized geometries using point charge arrays, it was necessary to generate a potential energy hypersurface to which intramolecular potentials could be fitted. Hypersurface generation involved randomly perturbing the positions of the C and O atoms in oxalate by up to 2.5% and computing the resulting energy. Two separate hypersurfaces were generated from 60 perturbations of the oxalate dianion. The corresponding energy of each perturbation was determined using the 6-31G basis set (without MP2 adjustment) for the Hartree-Fock hypersurface, while the DN basis was set used to calculate the density functional

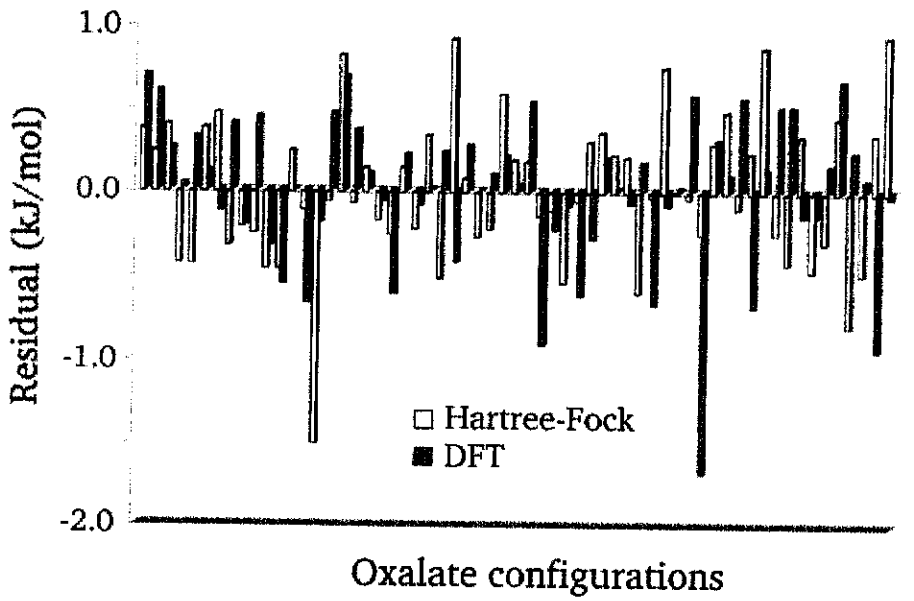

Figure 3.10: Energy residuals for the fitted potential models, compared against the respective quantum mechanical calculations.

hypersurface.

With the use of the GULP program, intramolecular potentials were derived from both hypersurfaces. Harmonic terms were found to be successful in describing the two body bond and three body terms. As a measure of the goodness of fit of each potential model to the respective hypersurface, a plot illustrating the residual energy for all oxalate perturbations is displayed in Figure 3.10. The intramolecular potentials reproduce the respective hypersurfaces to good agreement, with the RMS value of the error being 0.45 kJ/mol in both cases. The derived potentials are given in Table 3.9.

The above intramolecular potentials were then employed to accurately reproduce the crystal structures of three oxalate salts. Multiple related structures were fitted in order to test the robustness of both intramolecular models. Thus, a transferable potential model for sodium, potassium and calcium oxalates was developed by the method of simultaneous fitting. This generated a potential model for oxalate which could be used for other oxalate systems, but which may not necessarily have been the best possible model for each individual oxalate compound. However, the benefits of producing a transferable potential heavily outweigh this loss of accuracy.

Atom types	Hartree-Fock		density functional	
	harmonic parameters			
	k (eV Å^{-2})	r_0 (Å)	k (eV Å^{-2})	r_0 (Å)
C-O	60.74	1.24	52.72	1.28
C-C	26.47	1.59	29.29	1.56
	three body parameters			
	k (eV deg.$^{-2}$)	θ_0 (deg.)	k (eV deg.$^{-2}$)	θ_0 (deg.)
O-C-O	7.17	127.80	6.90	127.20
O-C-C	7.29	116.40	12.04	116.80
	torsional parameters			
	k (eV deg.$^{-2}$)	ϕ_0 (deg.)	k (eV deg.$^{-2}$)	ϕ_0 (deg.)
O-C-C-O	0.01968	0	0.02014	0
C-C-O-O	0.2689	0	0.2961	0

Table 3.9: Oxalate intramolecular potentials.

Atom types	Interaction parameters	
	harmonic	
	k (eV Å^{-2})	r_0 (Å)
H-O	23.717	0.96
	three body	
	k (eV deg.$^{-2}$)	θ_0 (deg.)
H-O-H	2.4505	104.5

Table 3.10: CVFF water intramolecular potentials.

The sodium oxalate unit cell parameters and atomic coordinates were obtained from the X-ray diffraction study of Reed and Olmstead [213]. Potassium oxalate monohydrate data was obtained from the neutron diffraction study of Sequeira, Srikanta and Chidambaram [214]. The calcium oxalate monohydrate data was obtained from the from the X-ray diffraction study of Tazzoli and Domeneghetti [215]. In fitting calcium oxalate monohydrate, the partial occupancy of the water oxygens was neglected due to limitations of the GULP code when dealing with fractional occupancies. The dominant positions were assigned an occupancy of 1.

Initial values of the oxygen potentials were taken from the sulfate Buckingham model of Allan *et al.* [129]. This assumes that oxalate oxygen terms will be comparable to the sulfate oxygen terms. This approximation was made as the partial charge on oxygen in the sulfate model (-0.84) is very similar to the Mulliken charge from the Hartree-Fock calculations (-0.8492). Intramolecular water parameters necessary for the potassium and calcium oxalate structures were taken from the CVFF forcefield from MSI (Table 3.10). In similar fashion, the oxygen interactions for water were taken as being the same as the equivalent

Atom types	Hartree Fock	density functional
C	0.60	0.60
O	-0.80	-0.80
Na	1.0	1.0
K	1.0	1.0
Ca	2.0	2.0
O	-0.76	-0.76
H	0.38	0.38

Table 3.11: Atomic point charges.

Atoms	Buckingham parameters		
	A (eV)	ρ (Å)	C (eV Å^6)
C-C	993.630	0.2405	10.05
C-O_{oxal}	604.568	0.2782	13.56
O_{oxal}-O_{oxal}	5161.96	0.2224	19.97
C-O_{water}	1130.75	0.2464	18.64
O_{oxal}-O_{water}	482.146	0.3700	23.00
O_{water}-O_{water}	10520.5	0.2614	50.29
Na-C	730.60	0.3060	0.0
Na-O_{oxal}	2501.76	0.2488	0.0
K-C	6300.00	0.2628	0.0
K-O_{oxal}	12287.96	0.2441	25.0
K-O_{water}	1959.29	0.2866	14.0
Ca-O_{oxal}	1976.91	0.2830	0.0
Ca-O_{water}	2652.63	0.2753	0.0

Table 3.12: Oxalate intermolecular potentials.

sulfate interactions as the charge on oxygen in water is -0.82. The final charges assigned to each atom are given in Table 3.11. Formal charges were given to cations, whilst the partial charges on the oxalate atoms were allowed to vary as part of the fitting process.

Table 3.12 lists the intermolecular potentials for sodium oxalate, potassium oxalate monohydrate and calcium oxalate monohydrate. The first six potentials in Table 3.12, along with the charges and intramolecular potentials constitute the transferable oxalate-hydrate potential model. The models were able to reproduce the experimentally observed unit cell parameters with a reasonable degree of accuracy. Table 3.13 indicates the differences in unit cell parameters between the calculated model and the experimentally observed values. For each oxalate modelled with each method the first three calculated phonon frequencies were zero, as required. The diagonal elastic constants and the dielectric constants were positive, also as expected for a stable model. The bond lengths were recalculated

	Percentage deviation for computed versus experimental parameters					
parameter	Hartree-Fock			density functional		
	sodium	potassium	calcium	sodium	potassium	calcium
a	-0.43	1.44	-1.74	-0.68	2.17	-1.16
b	0.95	-0.15	-1.73	1.90	0.14	-1.39
c	-1.60	-1.87	1.41	-1.43	-1.88	1.80
β	-0.10	-0.58	0.09	-0.54	-0.28	0.01

Table 3.13: The accuracy of potentials derived from Hartree-Fock and density functional methods in reproducing the experimental structure of sodium oxalate, potassium oxalate monohydrate and calcium oxalate monohydrate.

and still agreed with the distances expected for oxalate and water. The potassium oxalate fitting was less accurate, probably due to its longer carbon-carbon bond length, which was less accurately modelled by the general oxalate model than was the case for either the sodium or calcium structures.

3.2.4 Discussion

Although both potential models perform well, the Hartree-Fock derived potentials are slightly superior. This may seem surprising as these calculations did not utilize any post Hartree Fock corrections, so the density functional calculations are expected to be more accurate. However, given that the Hartree-Fock calculations best reproduced the geometry of the oxalate anion, it can be seen that the best potential set is the one which includes the best intra-molecular potentials. This supports the use of accurate quantum mechanical calculations in the derivation of interatomic potentials of molecular anions.

It is notable that standard Hartree-Fock, density functional and semi-empirical calculations on the molecular oxalate dianion failed to produce a stable geometry. Embedding the dianion in a point charge array stabilized the molecule allowing derivation of a potential energy hypersurface. Intramolecular potentials were successfully fitted to the hypersurface. Subsequent fitting of intermolecular potentials resulted in models that matched the unit cell parameters for sodium oxalate, potassium oxalate monohydrate and calcium oxalate monohydrate. The potentials derived with the Hartree-Fock intramolecular potentials represent a good transferable potential model.

Chapter 4

Predicting Gibbsite Morphology

4.1 Background

Industrially, the crystalline habit is often highly important, as certain forms are more desirable than others. This may be due to intermediate processing concerns, such as filtering, or for convenience in transportation and handling. In the case of a final product, there may be a desired habit specification. In all cases, a knowledge of the morphology and how to properly regulate it is of considerable importance. To this end, the application of molecular modelling to gibbsite morphology prediction was examined.

As was noted in the discussion in section 1.2.2, gibbsite growth is integration controlled. Thus, morphology techniques involving surface and structural energetics should provide a convenient means for predicting the crystalline habit. A predicted habit can be readily obtained using molecular modelling, providing the interatomic forces within the crystal can be ascertained with reasonable accuracy. The interatomic potential model, developed in the previous section, was constructed independently of any morphology estimate. Comparison of the morphology prediction and the experimental data should therefore yield an indication of the quality and transferability of the underlying potential model.

There are a few different forms of gibbsite morphology that have been documented in the literature. The most commonly occurring are the diamond and hexagonal prisms [91][152][217]. In both cases, the basal {002}, and the prismatic {110} planes are morphologically dominant. In addition, the faces that complete the hexagonal prism in the morphology are the {200} planes. Mineralogy references give a typical morphology (Figure 4.1) that is similar to the illustration in Deer, Howie and Zussman [57]. Industrially manufactured crystals (grown in caustic conditions), while consistent with this habit, tend to exhibit certain ad-

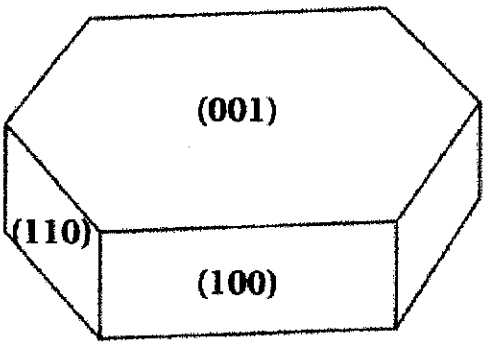

Figure 4.1: Commonly occurring faces in gibbsite morphologies.

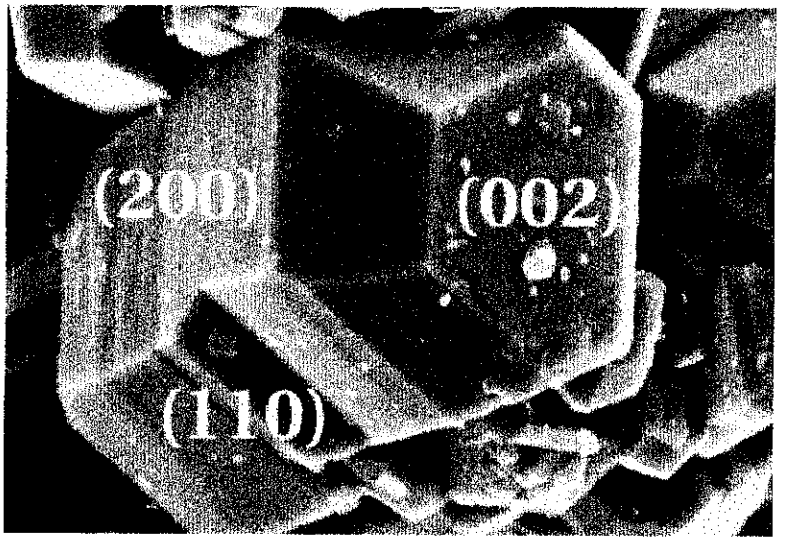

Figure 4.2: Experimental gibbsite morphology, grown slowly from caustic (from Lee [218]).

ditional faces. Figure 4.2 is an SEM micrograph of laboratory produced gibbsite, taken by Lee [218]. These crystals were grown by ageing sodium aluminate solutions (2.7 moles L^{-1} aluminium, 3.8 moles L^{-1} sodium hydroxide), prepared by dissolving gibbsite (C31, Alcoa Chemical Division, Arkansas) in sodium hydroxide (AR Grade) solution, and subsequent holding at 80 ^{o}C for 95 hours. The morphology of the crystal is clearly pseudo-hexagonal. The principal crystals faces were identified as the (002) basal plane, with the (110) and (200) planes forming the edges of the crystal. The presence of additional surfaces was also noted in these morphologies. By measuring the angles between the prismatic planes and the unknown faces, these were identified as the {101} and {112} surfaces.

Morphology prediction techniques are not limited solely to the case of single crystals, and may be readily extended to examine the phenomenon of crystalline

twinning. As mentioned in section 1.1.1, twinning arises due to a mis-orientation in lattice stacking. In this chapter, the prediction of contact twins has also been attempted. Contact twins are defined as two or more connected crystallites, each possessing orientations that are related by the action of a symmetry operation. Restriction is also made to the cases where the symmetry relationship between twinned crystallites is either a mirror or glide plane.

The compounds gibbsite and sodium oxalate, both of which are relevant to the Bayer process, are known to form twins. Researchers have observed gibbsite crystals that are six-fold twinned about [001], resulting in a pseudo-hexagonal morphology [91][217]. These are distinct from the single crystal hexagons mentioned previously, as the twin boundaries (particularly at the vertices of the hexagon) display some degree of discontinuity. In an aqueous environment, Strom *et al.* [142] observed that sodium oxalate is almost always twinned on the (200) face. Thus, an investigation into the twinning of these two substances was conducted. The model was tested for generality by applying it to several other systems in which twinning is also observed. Selection of these trial systems was limited primarily by the availability of good potential models for crystals that are known to form twins. All potentials were obtained from the literature, with the exception of those for sodium oxalate and gibbsite which were derived during the course of this project. Potentials for zircon and corundum were extracted from the paper by Gay and Rohl [141], potentials for rutile were taken from the work of Sayle *et al.* [219], and aragonite parameters were obtained from Pavese *et al.* [128].

4.2 Gibbsite Single Crystal Habit

4.2.1 Methodology

A morphology prediction scheme similar to that outlined in the paper by Gay and Rohl [141] was used. In order to predict the morphology of gibbsite, a set of suitable faces for which the surface and attachment energies are computed must be obtained. Low index faces are generally more dominant in the morphology, as they tend to have thicker growth slices than the higher index faces. Thus, calculations could be limited to a set of candidate faces comprising the lower index faces. However, without any additional information, this list would still be quite large. A better approach is to use a BFDH analysis to generate a set of candidate faces that is only slightly larger than necessary. Thus, selection of candidate faces (hkl) is determined by the corresponding interplanar spacings

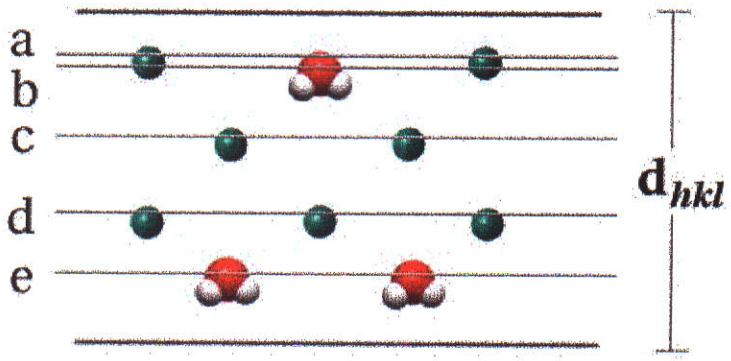

Figure 4.3: Shifts generated by z coordinates of atoms and molecule centroids in a given growth slice of thickness d_{hkl}. The five possibilities a to e represent all possible unique cuts, each of which generates a different planar surface.

(d_{hkl}). Although an over simplification, this method proved useful in reducing a large number of possibilities to a small set which includes the faces that appear in the morphology.

All surfaces are constructed by discarding atoms and molecules in the bulk structure that are above an appropriate planar cut. However, the variable factor is the offset or shift of the cleavage plane within the interplanar spacing d_{hkl}. That is, there may be more than one surface configuration for any given plane, according to where the bulk is cleaved within a layer. The surfaces in gibbsite are all type II faces, which means that not all cuts are valid, due to the formation of a perpendicular dipole moment and thus an infinite surface energy [188]. To automate the process of determining the allowed cuts for each face, a Perl script COMPSHIFT (see Appendix A.2.1) was written. This program scans all possible shift values within the interplanar spacing. These shifts are derived from the z coordinates of the atoms in a growth slice. Thus, each cut is generated in order to keep a layer of atoms, and all those below, while discarding those above. Molecules are considered whole entities, so that the centroid of the molecule corresponds to one shift value (Figure 4.3). For each of these shift values, the dipole moment perpendicular to the surface is computed, and discarded if it does not vanish. For this dipole calculation, the data from a single point MARVIN calculation is used.

For gibbsite surfaces, the resulting output from the COMPSHIFT program is summarized in Table 4.1. Here, the number of raw cuts is the total number of cuts with zero dipole, whilst the reduced cuts have some of these shift values ignored because the corresponding unrelaxed surface energy is too high. Eliminated shift

(hkl)	d_{hkl}	Raw cuts	Reduced cuts	Shift values
(002)	4.8675	2	1	0.25
(200)	4.3810	2	1	0.0
(110)	4.3384	4	1	0.0
(101)	6.2006	4	2	0.0 0.5
$(10\bar{1})$	6.8769	4	2	0.0 0.5
(112)	3.1591	2	1	0.0
$(11\bar{2})$	3.3246	2	1	0.0
(011)	4.4430	4	2	0.28 0.72
(012)	3.4855	4	2	0.24 0.76
(111)	3.8891	6	2	0.0 0.5

Table 4.1: Computed shifts for gibbsite.

values were those with associated surface energies differing by more than 1.5 J m^{-2} from the lowest unrelaxed surface energy for that face (Appendix A.2.1).

Surface configuration

As the only molecules in the model are the hydroxide ions, the hydrogen positions play an important role in determining the configuration of a cleaved surface. As a result, it was necessary to compare the surfaces generated from the experimental crystal structure (modified by the quantum mechanically derived proton positions) with the surfaces obtained using the developed empirical potentials. This is due to the fact that the proton positions are not perfectly reproduced by the latter. For most faces, the application of a planar cut to both the experimental and computed crystal structures produced equivalent surfaces. In Figure 4.4, the generated (002) face has been illustrated, showing the six coordinate surface aluminium atoms. In addition, the chamfering faces {101} and {112}, as well as the prismatic {200} face were all successfully reproduced using the planar cut mechanism. Note that the monoclinic symmetry of the gibbsite lattice means that separate calculations are required for the (101) and $(10\bar{1})$, as well as for the (112) and $(11\bar{2})$ faces. Each of the remaining chamfering faces are related by symmetry to those above.

When the planar cut method was applied to generate the (110) face from the computed structure, there were found to be alternate rows of four and six

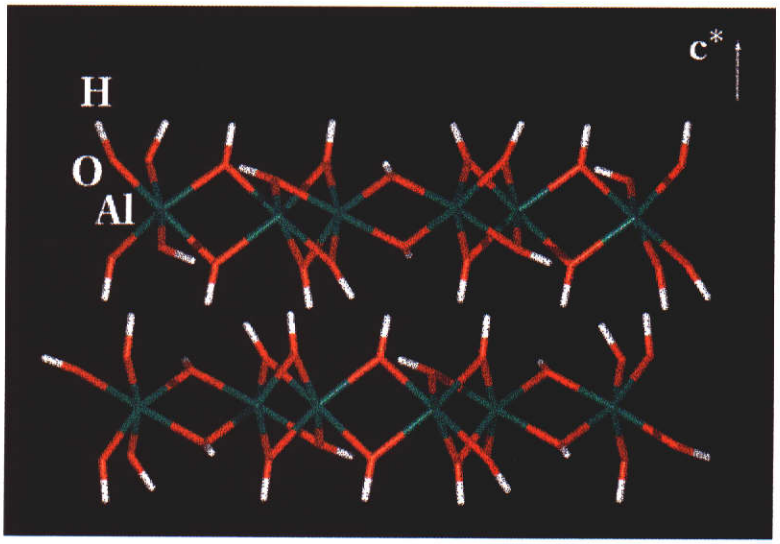

Figure 4.4: The (002) surface generated *via* planar cleavage, viewed side on.

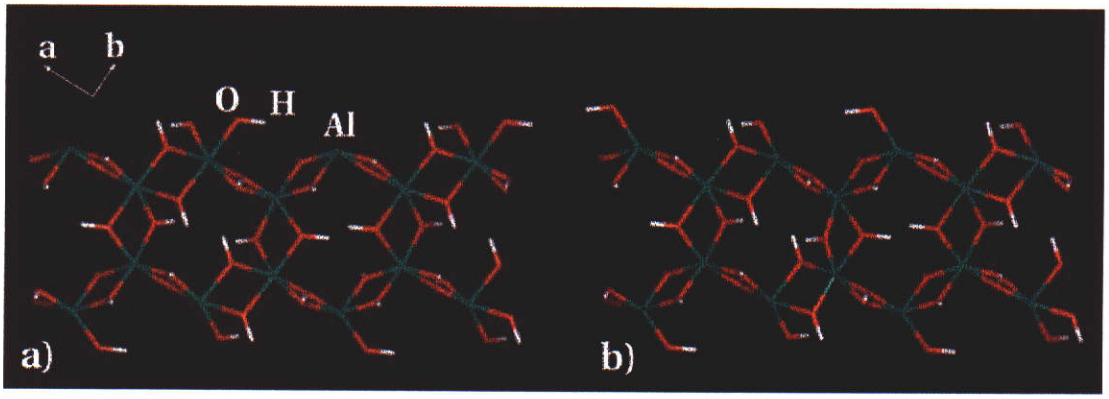

Figure 4.5: Cleavages of the (110) surface, a) planar cut, b) manual cut.

coordinate aluminium atoms (Figure 4.5a). On relaxation, the surface energy of this face is too large for it to appear in the morphology. As an alternative to the planar cut method, a surface may be manually constructed by swapping pairs of hydroxide ions on either side of the cleavage plane. This enables a different configuration to be examined, whilst preserving the system's overall charge neutrality. By altering the positions of two hydroxide ions, it was possible to create a surface where all the surface aluminium atoms are five coordinate (Figure 4.5b). This configuration matches the coordination of the experimental surface, created using a planar cut. Subsequent calculations have indicated that this is the preferred configuration, as the relaxed surface energy is significantly lower than that of the planar cut shown.

Apart from (110), the (111) surface was the only other for which a planar cut did not produce a configuration that matched that generated from the crystal

plane	E_{surf} (J m^{-2})	E_{att} (kJ mol^{-1})
(002)	0.22	-347
(200)	0.59	-1991
(110)	0.59	-3196
(101)	0.45	-996
$(10\bar{1})$	0.53	-949
(112)	0.46	-1922
$(11\bar{2})$	0.50	-2154
(011)	0.75	-3149
(012)	0.63	-5893
(111)	0.64	-2964

Table 4.2: Relaxed surface and attachment energies for gibbsite.

structure, and needed to be manually constructed.

4.2.2 Results

The relaxed surface and attachment energies were computed using the MARVIN [141] package (Table 4.2). Clearly, the basal plane is dominant in both predictive schemes. However, the attachment energy for the (110) face is more negative than most of the other attachment energies. Thus, although the other prismatic face (200) will not be eliminated by virtue of the magnitude of E_{att}, it will nonetheless have little morphological importance due to the apparent rapid growth of the (110) face. It is also evident that the difference in relative growth rates of the two prismatic faces will also generate an elongation in the growth morphology that is perpendicular to c. Such morphologies have not been experimentally observed.

Using the data in Table 4.2, Wulff plots may be generated to visualize the calculated gibbsite habits for the two schemes. The calculated equilibrium and growth morphologies are illustrated in Figures 4.6 and 4.7 respectively. As expected, the morphology predicted from the relaxed attachment energy differed from experimental observation, as the {200} and {110} faces are not reproduced. Also, the predicted growth morphology is significantly elongated in the [010] direction, which is not consistent with experimental observations. However, the relaxed equilibrium morphology, with a non-planar cut for the (110) face, successfully predicted the appearance of the prismatic faces, giving the best agreement with experiment. The only difference between the predicted tabular hexagonal morphology and experiment appears to be the lower than observed morphological importance of the {200} and {110} prismatic faces.

The actual structure of the relaxed gibbsite surfaces was examined. The (200)

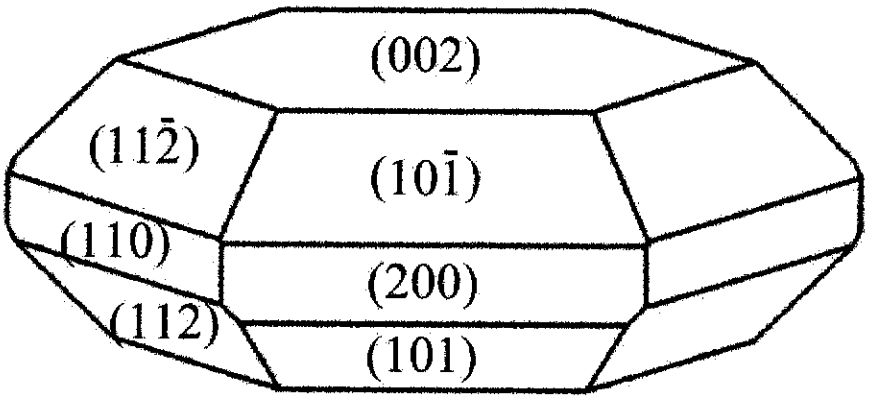

Figure 4.6: Relaxed equilibrium morphology of gibbsite.

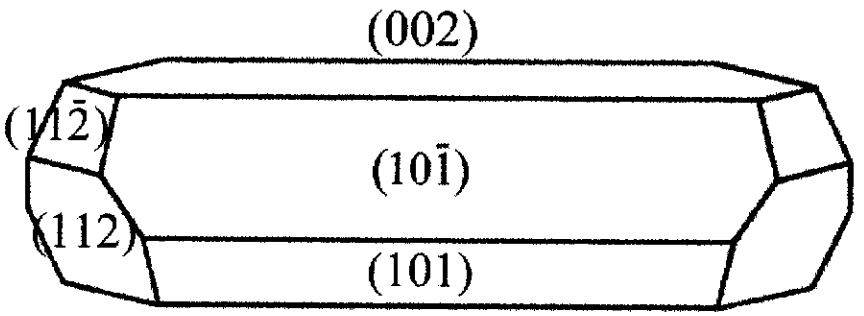

Figure 4.7: Relaxed growth morphology of gibbsite.

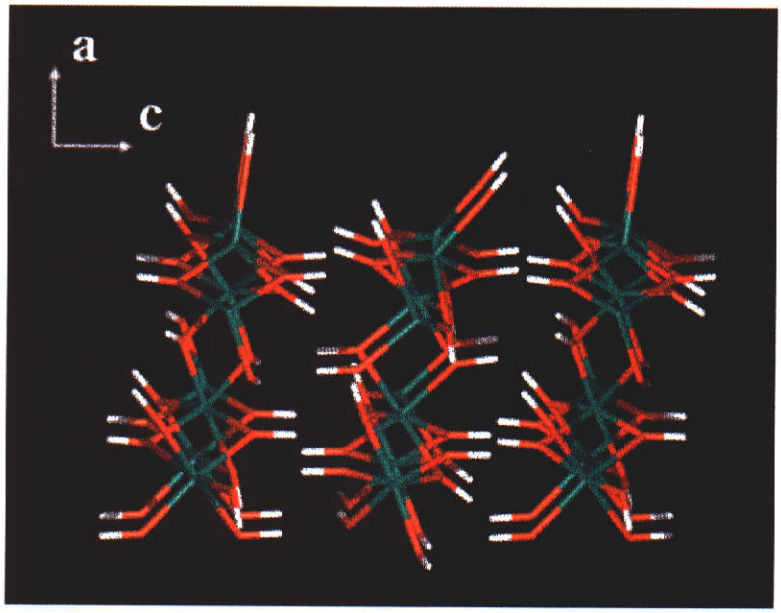

Figure 4.8: The relaxed (200) surface configuration.

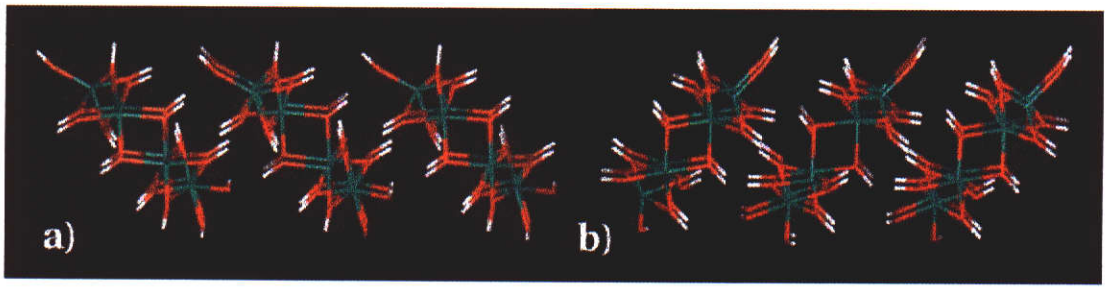

Figure 4.9: Side views of a) the (101) face and b) the $(10\bar{1})$ faces.

face, unlike the other prismatic (110) face, was cleaved to form a five coordinate aluminium surface without any manual modification (Figure 4.8). The relaxed (101) face and $(10\bar{1})$ faces, both obtained with a purely planar cut, are shown in Figure 4.9. The relaxed (112) and $(11\bar{2})$ surfaces, each of which was also generated using a planar cut, are shown in Figure 4.10. From the above figures, and those shown in the previous section, it is clear that the stable surfaces of gibbsite are all characterized by five coordinate aluminium atoms, with the six coordinate basal plane being the only exception. The other faces, selected as possible candidates in the morphology due to their d_{hkl} value, but discarded due to large surface energies, have at least one four coordinate aluminium at the surface. For most of these cases, it was not possible to modify the surface hydroxyl groups to increase the coordination of all the four coordinate aluminium atoms. This was due to charge neutrality constraints combined with the lack of suitable six coordinate surface aluminium atoms for exchange. In addition, some cases resulted in configurations

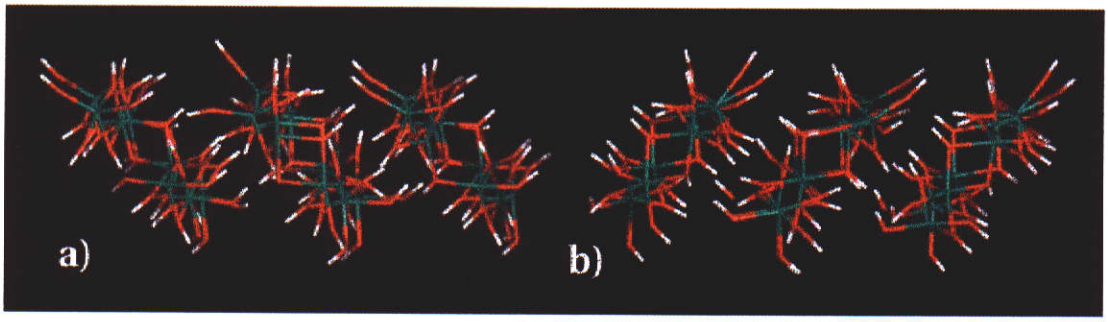

Figure 4.10: Side views of a) the (112) and b) the $(11\overline{2})$ surfaces.

that were unstable due to very uneven surface structure.

4.2.3 Discussion

With no consideration of solvent, the morphology predictions in this work have been carried out in what amounts to vacuum conditions. It follows that a large component in determining the validity of this work is the effect of the presence of a solution on the morphology. In principle, the calculations could be modified to account for the presence of solvent molecules. If steady-state conditions were assumed, this would yield little additional information, as all faces would be likely to be affected equally. For the kinetics of the growth to be properly modelled, the use of molecular dynamics to investigate the liquid-solid interface would be required. Unfortunately, this would be too computationally demanding. Hence, the methods discussed and developed here succeed where the growth mechanism is dominated by structural factors alone.

Despite problems when generating the correct cut for the (110) surface of gibbsite, the relaxed equilibrium morphology yielded a prediction that was in good agreement with experiment. It is likely that the configuration chosen, namely 5 coordinate aluminium atoms, is representative of the true surface. This is supported by the fact that the experimental structure with quantum mechanically optimized proton positions produces a similar surface. However, the configuration of the corresponding growth slice proved more difficult to ascertain. This uncertainty is compounded by the lack of available data concerning the growth unit(s) involved in gibbsite precipitation. The growth morphology, which is highly dependent upon a knowledge of growth units, failed to reproduce a result consistent with experiment. In particular, the attachment energies for the (110) surface is too low for the face to appear in the growth morphology.

It is notable that, for all the stable faces, the surface aluminium atoms are five coordinate. The basal plane, which is six coordinate, is the only exception

and will be morphologically dominant due to its significantly lower surface energy. This trend for gibbsite surfaces to be composed of entirely five coordinate aluminium atoms may have implications in growth unit studies of the prismatic and chamfering faces. In addition, as the basal plane is the only surface with 6-coordinate aluminium atoms, it seems likely that there is different growth mechanism and growth unit for the basal plane. These findings are supported in recent experimental AFM work by Freij *et al.* [96].

4.3 Gibbsite twinning

While not of significant concern to the Bayer industry, twinning can sometimes be a problem in areas of crystal growth where any form of defect is undesirable. For example, twinning is of interest in the production of thin-films [220]. In addition, twinning can sometimes play an important role in the growth kinetics [221]. Establishing a method for the general prediction of twinning in systems other than gibbsite would therefore be of interest. In addition, if the twinning model were successful, the potentials developed for gibbsite would receive some justification. As gibbsite morphology has been illustrated previously, representative habits for the other trial systems selected for twinning have been included. Illustrations of single crystal habits of aragonite (Figure 4.11), corundum (Figure 4.12), rutile (Figure 4.13) and zircon (Figure 4.14) were taken from the literature [222]. The habit of sodium oxalate was taken from Strom *et al.* [223], which was noted by the authors to be almost always twinned on the (200) plane as shown in Figure 4.15.

4.3.1 Methodology

It has been suggested that the main requirement for the formation of twins is a relatively small stacking fault energy [1][224]. Since this phenomena is directly related to the stacking of adjacent growth layers, it is apparent that a model based on attachment energy is the most suitable for analysis. To calculate the attachment energies of normal (E_{att}^{normal}) and reflected growth slices ($E_{att}^{reflected}$), the simulation model illustrated in Figure 4.16 was used. The simulation cell represents a vertical segment of crystal which is repeated infinitely in two dimensions. The normal growth slice is simply a continuation of the bulk structure, and thus has a known spatial location. The reflected slice was generated by constructing the mirror image of the growth slice, with the mirror plane parallel to the surface under consideration. Here, the situation is more complicated as

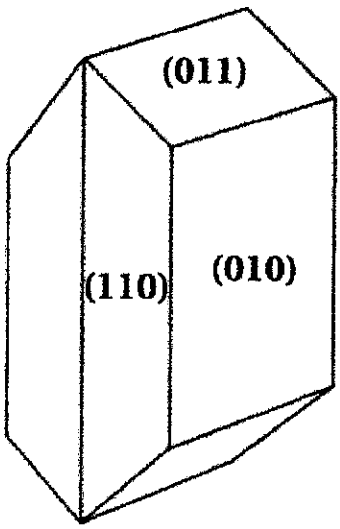

Figure 4.11: Sample habit for aragonite.

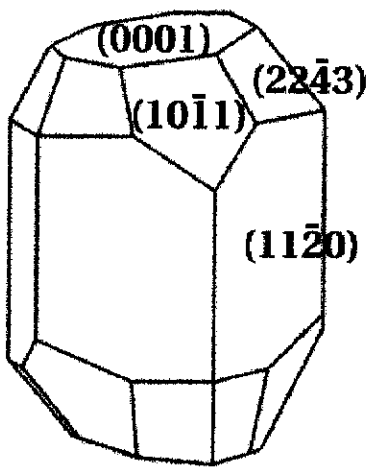

Figure 4.12: Single crystal morphology for corundum.

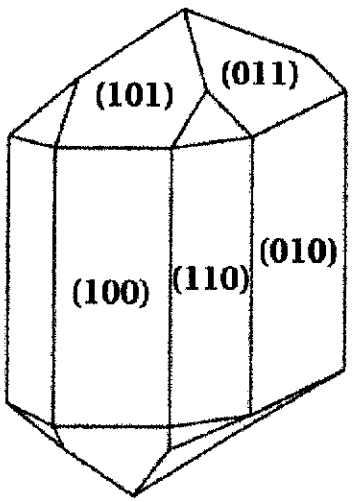

Figure 4.13: Single crystal morphology for rutile.

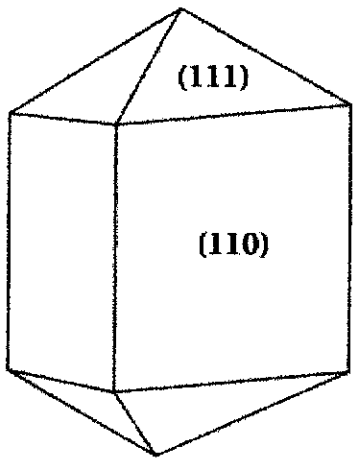

Figure 4.14: Single crystal morphology for zircon.

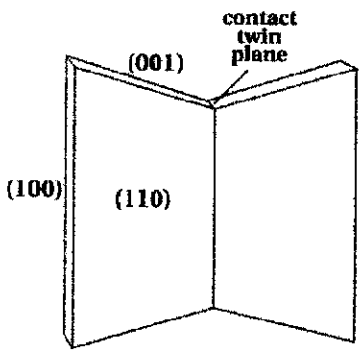

Figure 4.15: Sodium oxalate twinned crystal morphology.

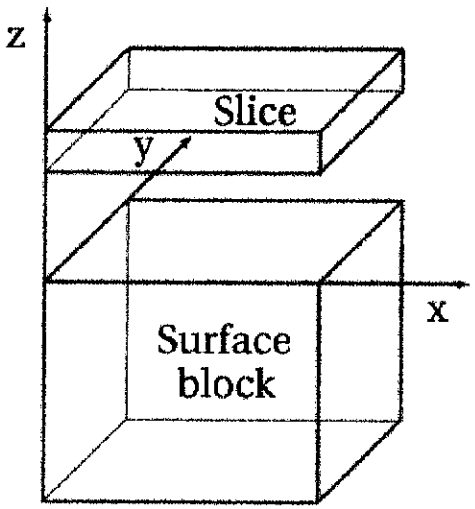

Figure 4.16: Simulation cell, for computing the attachment energy of a growth slice.

neither the vertical nor horizontal positions are known. For each value of (x, y, z), representing the position of the slice parallel to the growing surface, the total energy of the system can be computed using,

$$E_{total} = E_{surface} + E_{slice} + E_{inter} \tag{4.1}$$

Where $E_{surface}$ is the interaction of particles in the surface block, E_{slice} represents the interaction of particles in the slice, and E_{inter} is the interaction between particles in the slice with particles in the bulk. All quantities are computed using a periodic summation method.

Minimisation of E_{total} was achieved by translating the reflected slice in three dimensions. This yielded the minimum value of E_{inter} ($\equiv E_{att}^{reflected}$) since $E_{surface}$ and E_{slice} are both independent of location. Subtracting the (minimized) total energy of a reflected slice from the total (minimized) energy calculated for a normal growth slice, produces the difference in attachment energy between the two slices,

$$\begin{aligned} E_{total}^{normal} - E_{total}^{reflected} &= E_{att}^{normal} - E_{att}^{reflected} \\ &= E_{twin} \end{aligned} \tag{4.2}$$

Thus, E_{twin} is the extra energy required for a reflected layer stacking fault to form. This quantity must be sufficiently small if twinning is to occur. However, in order to establish a quantitative means for predicting twinning, the E_{twin} values must be converted into a percentage of the total intermolecular lattice energy. This

particular scaling is used as the magnitude of the attachment energy is dependent upon only the intermolecular contributions to the energy. Intramolecular terms will not contribute as no new bonds are formed when a slice is attached to the surface.

Energy minimisation

A Perl script (Appendix A.2.2) was employed to compute E_{twin} in the simulations. Input parameters for the script are a list of Miller indices plus a valid corresponding shift. For gibbsite, these were taken from the results given in Table 4.1, and values for the other salts were calculated in a similar fashion. The script utilizes the computer program MARVIN [141] for single point energy calculations. Thus, a suitable input file containing the unit cell coordinates and a set of potentials was also required.

For each candidate face, a normal growth slice is reflected to form the twinned growth slice. Faces that possessed a mirror or glide plane which did not pass through any molecules were discarded. This removes the candidate faces where the twinned state is identical to the untwinned state. The mirrored slice is then placed above the bulk and allowed to translate as a rigid entity in three dimensions toward the position of lowest energy. To solve this minimisation problem, a modified version of the SIMPLEX algorithm [225] was employed. The action of the SIMPLEX method can be described using a tetrahedron sitting on the potential energy hyper-surface. Each vertex of the tetrahedron represents the spatial location of the growth slice, with a corresponding energy. By reflecting the point of highest energy through the opposing face, a new tetrahedron is created with a lower maximum energy. Thus, successive iterations will approach a minimum point, where all four vertices converge to within a given tolerance. For each face, the minimized total energy of the reflected system is subtracted from the total energy of the normal system and scaled to yield E_{twin}.

The basic SIMPLEX method was modified slightly in an attempt to improve the likelihood that the global minimum is reached. This change consisted of executing several independent minimisation cycles, each of which was performed with the mirrored slice starting from a different location above the surface of the crystal. The starting locations were constrained to form a grid with lateral coordinates ranging from 0 to the length of the surface repeat unit. This approach resulted in a lower minimum for a few cases. However, the local minima found previously did not differ greatly in energy from the more accurate answer. Increasing the number of grid locations beyond 3 by 3 was found to be computa-

	(hkl)	E_{twin}	(hkl)	E_{twin}	(hkl)	E_{twin}	(hkl)	E_{twin}	(hkl)	E_{twin}
gibbsite	(002)	-0.33	(110)	0.035	(200)	1.5	(101)	1.5	(112)	3.3

Table 4.3: Gibbsite twinning, E_{twin} as a percentage of the intermolecular lattice energy.

	(hkl)	E^1_{twin}	(hkl)	E^2_{twin}	(hkl)	E^3_{twin}
sodium oxalate	(200)	0.034	(001)	6.1	(110)	6.5
aragonite	(110)	0.30	(101)	2.3	(111)	9.3
corundum	(100)	0.64	(10-1)	3.9	(20-1)	7.6
rutile	(011)	0.25	(211)	3.3	(111)	5.1
zircon	(112)	1.8	(101)	2.3	(301)	8.2

Table 4.4: E_{twin} values as a percentage of the intermolecular lattice energy, for the three faces in each structure with the lowest energy requirement for twinning.

tionally infeasible.

4.3.2 Results

For all faces in our computed gibbsite morphology, the values of E_{twin} were calculated using a three by three grid and listed in Table 4.3. A negative value indicates a twinned structure that is more stable than the normal structure. For the (002) face of gibbsite, the preference for the twinned state is representative of the higher thermodynamic stability of bayerite *in vacuo*, which has a comparable Al, O stacking (see section 1.2.1). As the interatomic potentials derived were fitted to both structures, the higher stability of the bayerite stacking is reflected in the results. However, gibbsite twinned in this manner is not completely identical to bayerite, as the hydroxyl orientations are different. With the exception of the (002) face, all the other gibbsite surfaces in Table 4.3 have positive twinning energies. Of these, the (110) face is clearly the most favourable for twinning to occur.

The lowest E_{twin} values, computed for the faces in the other trial systems, are presented in Table 4.4. In all cases, the energy required for twinning was found to be positive. The calculations were all performed using a 3 by 3 grid minima search, and all values have been converted to a percentage of the intermolecular lattice energy. It was confirmed from the literature (Table 4.5) that, for all the above systems, the face with the lowest E_{twin} value is experimentally observed to twin.

By scaling the E_{twin} values, it is anticipated that a maximal cut-off may be determined, below which twinning will occur. If the results in Tables 4.4 and 4.3

	Twinning	References
sodium oxalate	Usually twinned on (200)	[223]
gibbsite	Common on (001), and about [130] less common on (100) and (110)	[57][58][222][226]
aragonite	Very common on (110)	[58][222][226]
corundum	Common on (100)	[58][222][226]
rutile	Common on (011), also on (031) as interpenetrating twins.	[58][222][226]
zircon	Rare on (101) and (112)	[227][228][229][230]

Table 4.5: Observed twinning for the selected crystalline materials.

are compared to a cutoff of a little less than one percent of the lattice energy, then all systems are in agreement with experimentally observed twins that occur with a high degree of frequency. Note that the value of E_{twin} for sodium oxalate is quite small. This is consistent with the observations by Strom *et al.* [223] indicating that sodium oxalate, grown in an aqueous solution, is almost always twinned on the (200) plane. In the cases where twinning is rare or uncommon, it is anticipated that a cut-off of a little more than two percent is necessary, for example, in order to incorporate the (101) and (112) faces of zircon. In addition, this would also account for the rare (200) twinning in gibbsite. However, such a cutoff would also predict that the (101) face for both aragonite and gibbsite should also twin, which has not been observed to occur.

For each compound, the configuration of the face that produced the smallest value of E_{twin} was examined. These surfaces are evidently those that, according to the model, are the most likely to exhibit twinning. Two projections of the same twin on the (002) plane of gibbsite are shown in Figure 4.17. These views show that during the minimization process the reflected slice has undergone lateral translation in one direction only. The twinned symmetry relation is thus a glide plane, and not a pure reflection. Figure 4.18 shows two projections of the molecular arrangement in a minimized sodium oxalate twin, both of which are parallel to the (200) plane. In this case, the reflected slice has undergone lateral translation along both surface axial directions. Images of the twin boundaries for aragonite (Figure 4.19), corundum (Figure 4.20), rutile (Figure 4.21) and zircon (Figure 4.22) are also presented.

4.3.3 Discussion

From most of the mineralogy references, gibbsite twinning on the (110) face is relatively infrequent. The results from the calculations suggest otherwise. The

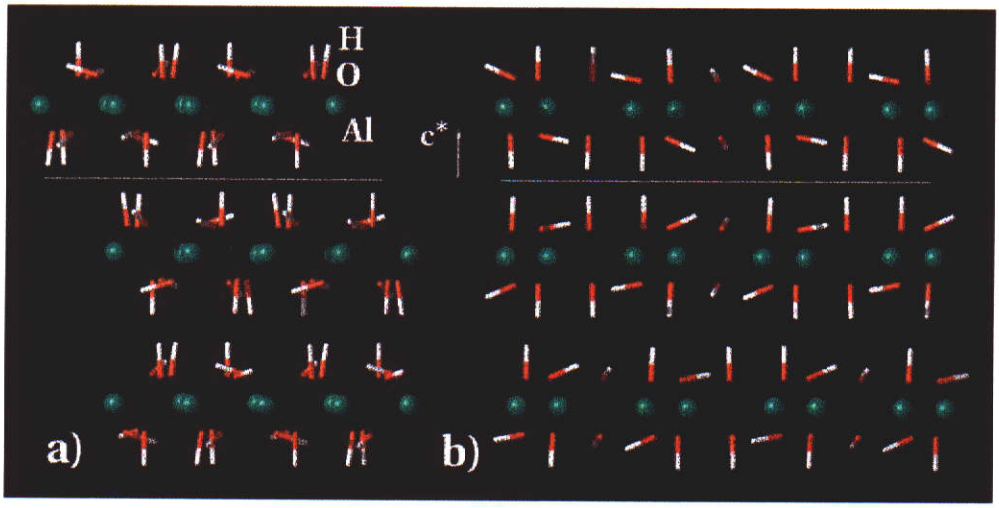

Figure 4.17: Minimized basal plane twin of gibbsite looking along the a) a axis and along the b) b axis.

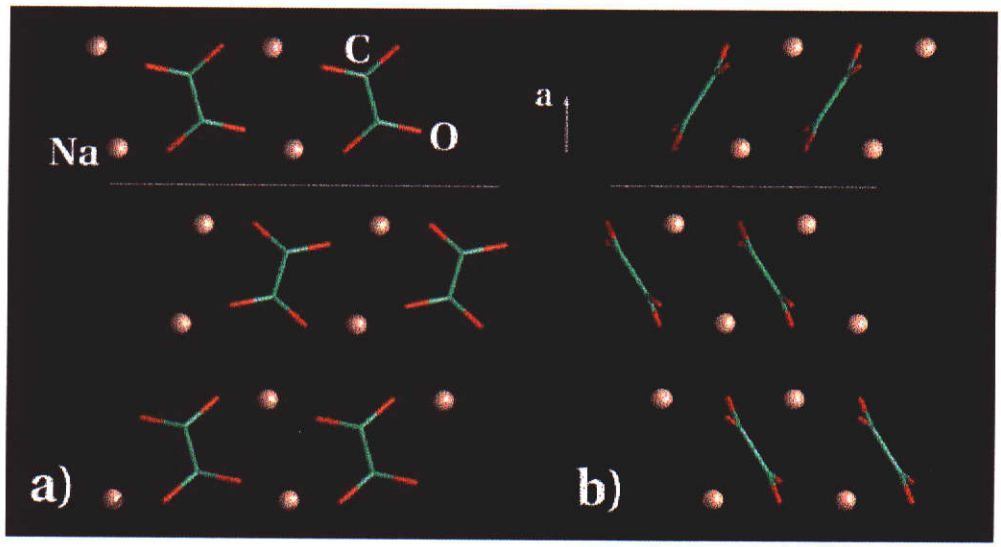

Figure 4.18: Sodium oxalate reflection twin on the (200) plane, viewed along the a) b axis and along the b) c^* axis.

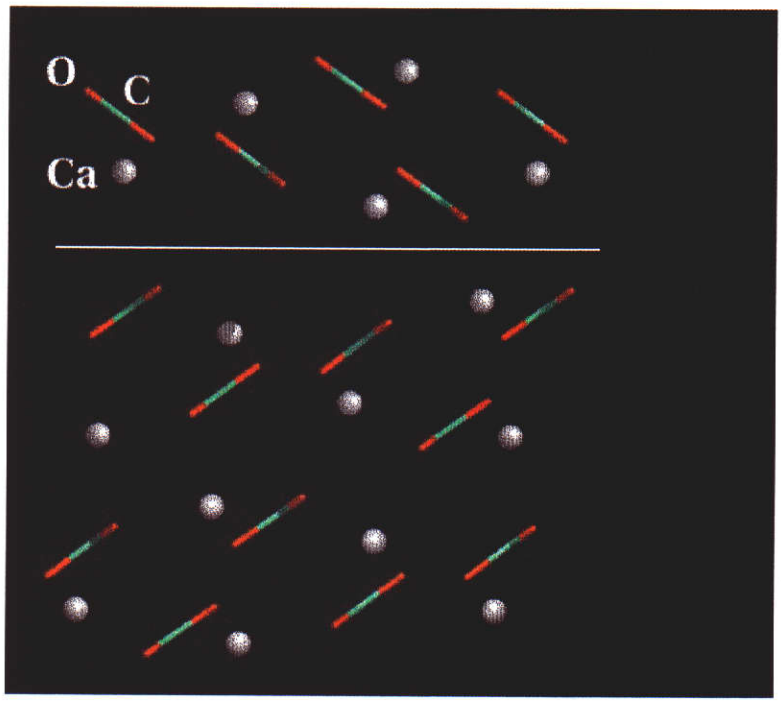

Figure 4.19: Aragonite, twinned on the (110) plane.

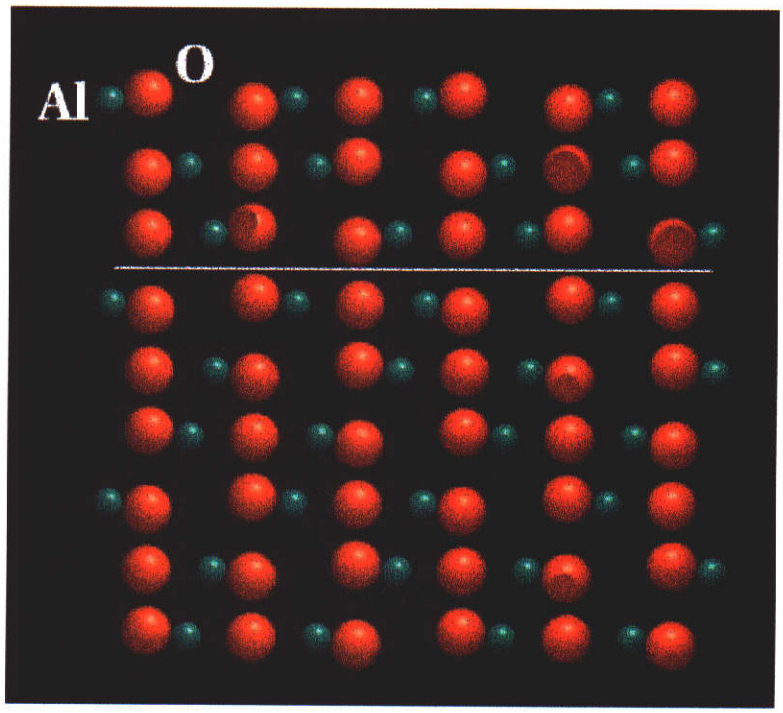

Figure 4.20: Corundum, twinned on the (200) plane.

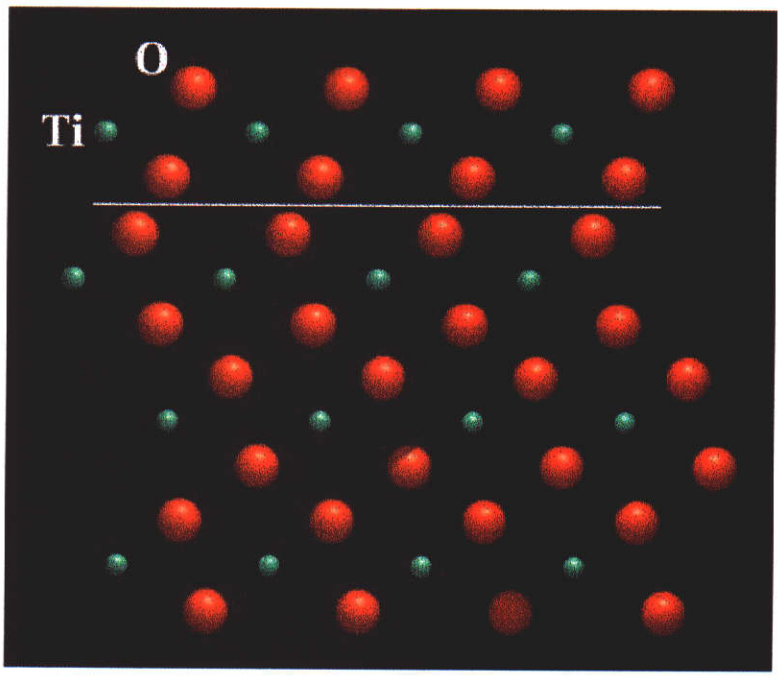

Figure 4.21: Rutile, twinned on the (011) plane.

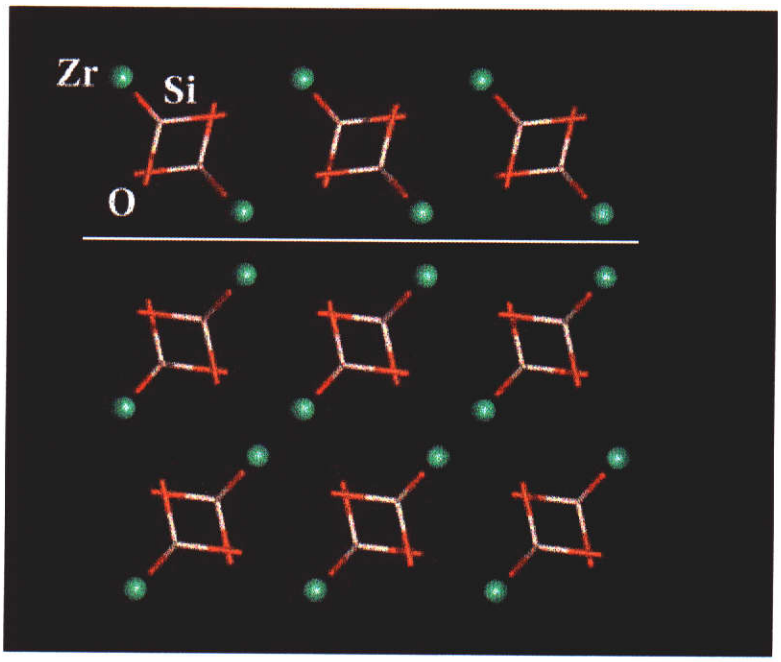

Figure 4.22: Zircon, twinned on the (101) plane.

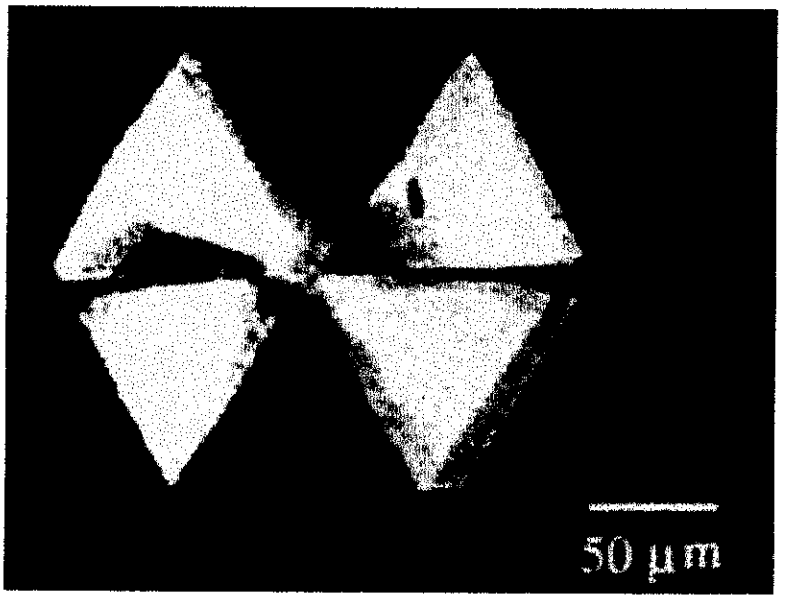

Figure 4.23: Polarization micrograph of a six-fold gibbsite twin. The common boundaries between segments are the twin planes, and are parallel to the (110) surface (from Sweegers *et al.* [217]).

highly favourable twinning energy for the (110) face of gibbsite has some support from Sweegers *et al.* [217]. Here, gibbsite crystals were grown from synthetic Bayer liquors (initial relative γ-$Al(OH)_3$ supersaturations of 0.21 - 0.45 at 80°C) with the authors noting a wide variety of gibbsite morphologies, plus a significant degree of twinning. The occurrence of relatively small elongated hexagonal gibbsite prisms (2-5 μm in width and 10-15 μm in length) as well as much larger (100-200 μm) hexagonal plates was noted. The authors state that the latter case was always the result of six-fold repeated twinning about the (110) plane (Figure 4.23). In addition, frequent polysynthetic twinning on the (002) plane was also observed (Figure 4.24), which is in good agreement with the low twinning energy requirement calculated in this chapter.

The method described in section 4.3.1 was successful in predicting forms of frequently occurring twinning. These involved only planes of reflection and subsequent translation. However, there exist other forms of twinning that the method cannot model in its current form. These include inversion and rotation twins, such as the one about the [130] direction in gibbsite (see Table 4.5). It is also evident that deformation twinning, resulting from the motion of dislocations in an applied stress field is unsuited to this method. Although only a mirror operation is coded, a rotation operation may also be incorporated provided care is taken to ensure that the lateral repeat vectors of the rotated slice are coincident with the surface lateral translation vectors. It is also feasible that the method-

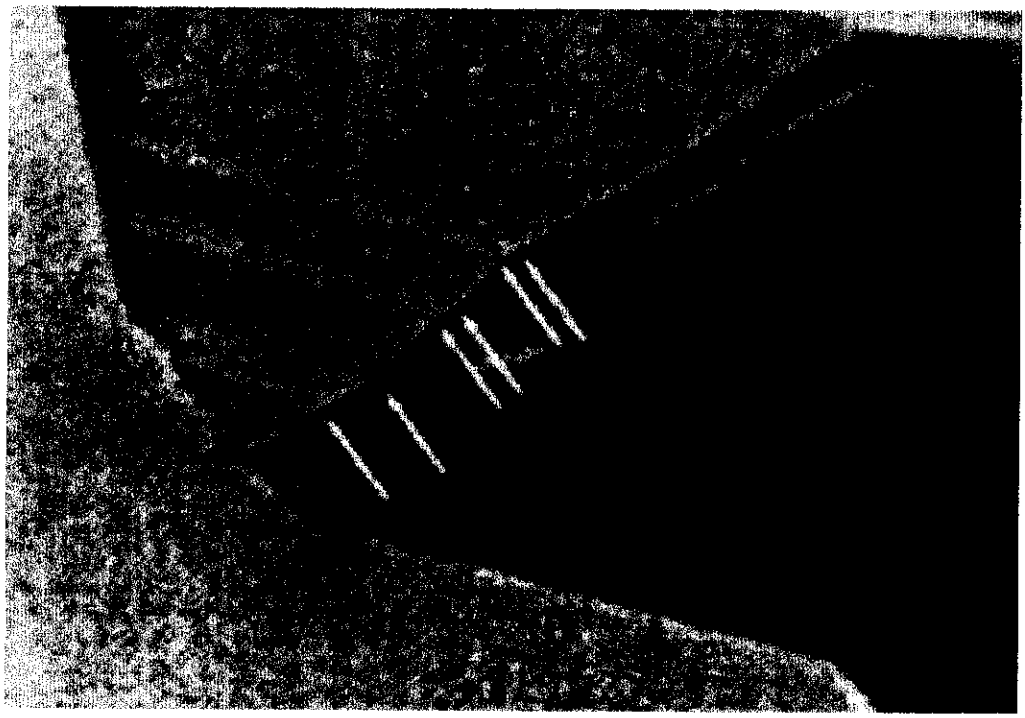

Figure 4.24: Polysynthetic twinning in gibbsite, with arrows indicating the multiple twin boundaries parallel to the (002) plane (from Sweegers *et al.* [217]).

ology employed to predict twinning may be extended to examine other types of stacking faults. It would be possible to determine the energy released when a suitably adjusted layer is attached to the crystal, and compare this value to the attachment energy of a regular layer. This would yield information about both the configuration, and the likelihood of formation for such a defect.

Chapter 5

Habit Modification

5.1 Background

The incorporation of foreign species into crystals has received much attention in the literature. Of particular relevance is the effect that these impurities have on the surface energy, and therefore crystal morphology. Many studies of defects have involved computing the surface segregation energy, which is defined as the tendency for a defect in the bulk to migrate to the surface [149]. For example, researchers have modelled the effects of cationic defects on the surface energy, and thus predicted the resulting effect on the morphology of calcite [148]. Further studies on this system have also yielded information concerning nucleation and crystal growth [150]. Corundum studies by Davies *et al.* [231] indicate that segregation tends to be highly surface and coverage dependent. However, most of the defect studies described above have been concerned with impurity migration to the crystal surface, important in processes such as sintering. In this work, the focus is on the energy required to replace a proton at the surface of gibbsite with a cation from solution. Thus, although a similar approach is adopted, the cationic replacement energy is used instead of the segregation energy.

The potential model presented in Chapter 4 successfully predicted all observed faces in the experimental gibbsite morphology. However, the calculations underestimate the importance of the prismatic faces, possibly due to the assumption of vacuum growth conditions. It is therefore of interest to include some effects of solution upon the morphological prediction. Solution studies have shown that cationic species tend to be closely linked with clusters of aluminate monomers [61]. This solution structuring is examined in more detail in Chapter 6. In addition, evidence of cation incorporation was noted in the work by Lee *et al.* [152]. Thus, it is possible that incorporation occurs when growth units (accompanied by

Cation	Incorporation (ppm)
Sodium	67000
Potassium	5050
Caesium	1750

Table 5.1: Cation incorporation in gibbsite crystals grown in three different alkali metal aluminate solutions, taken from Lee *et al.* [152].

associated cations) dock into the surface of gibbsite. This work has been based on the suggestion of Wefers [55], that the sodium cation acts as an impurity substitution by replacing a proton.

Gibbsite crystals grown experimentally in sodium, potassium, and caesium aluminate solutions are generally characterized by pseudo hexagons or diamond shaped morphologies. Typically, the importance of the prismatic faces in the pseudo-hexagonal morphology increases with the ionic radius of the cation [91]. Impurity incorporation levels in alkali aluminate solutions have also been measured by Lee *et al.* [152], as shown in Table 5.1. The key feature of these results is the relatively high degree of sodium bulk incorporation, compared to the larger alkali metal cations.

5.2 Method

5.2.1 Candidate replacement sites

Modelling of surface defects is usually assumed to follow a Langmuir isotherm, so that the enthalpy of adsorption of a single impurity is independent of the fractional coverage [175]. However, this is only true for values of surface coverage that are sufficiently low enough to prevent defects from repelling one another. In addition, the local surface structure at individual sites may also vary considerably, resulting in different degrees of defect interaction on each face of the same crystal. Surface hydrogen separations in gibbsite are relatively small, and high values of defect coverage would undoubtedly result in a considerable amount of repulsion. Thus, a minimum surface area requirement per defect will be imposed. This requirement is made to ensure that the calculations made are for isolated defects only. Clearly, extrapolation of this data to high values of defect concentration will not yield very reliable results. As a consequence, only relatively low surface coverage values will be considered for analysis.

The surface structure of gibbsite is quite complex, and for some faces it is not clear which protons should be considered as surface sites. The figures in Section

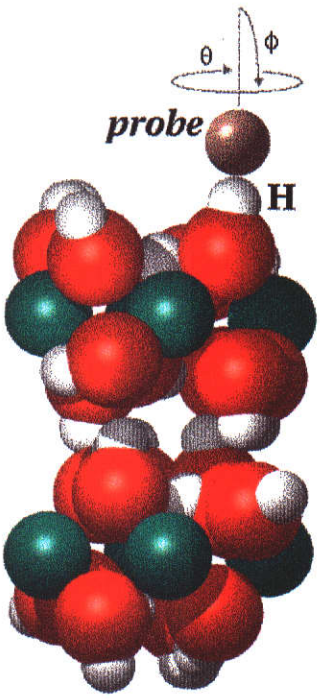

Figure 5.1: Hydrogen accessibility determination, achieved by sweeping a probe atom over the surface (spherical polar coordinates used).

4.2 show how there are a range of hydrogens at different elevations, with partial overlap making it difficult to distinguish between surface and bulk hydrogens. However, it is imperative to determine the total number of surface sites, firstly in order to determine candidates for substitution calculations, and secondly so that surface coverage values may be assessed. In this work, a modified version of the method of Connolly [232] has been adopted in order to analyze the surfaces in the calculated morphology of gibbsite. The essential idea is to determine how accessible the hydrogen atoms are to a given probe atom. A program was written in Perl to accomplish this (see Appendix A.2.3). The code determines how much of the surface of each proton can be brought into contact with a probe atom, without the probe overlapping the surface of any other atom. The size of each atom in the gibbsite surface was taken to be the appropriate Van der Waals radius, whilst the ionic radius was used for the probe. Two separate analyses, one with a sodium probe and the other with a potassium probe atom, were performed. The percentage accessible area of the entire sphere of the hydrogen atom is scanned by looping through values of ϕ and θ, as indicated in Figure 5.1.

A general scheme was applied to differentiate between a surface and a bulk hydrogen. A list of accessibilities of all unique protons was drawn up, and sorted from highest to lowest (Tables 5.2 and 5.3). Clearly, the largest accessibility values

	Hydrogen accessibilities (%)								
Candidate	(002)	(200)	(110)	(101)		(10$\bar{1}$)		(112)	(11$\bar{2}$)
1	32.46	49.27	49.12	37.87	38.60	41.67	41.81	44.44	43.86
2	30.85	39.62	47.66	37.87	38.45	41.67	41.81	41.96	40.94
3	30.41	30.85	46.78	36.70	37.43	32.60	33.19	37.57	38.60
4	28.80	15.35	39.91	36.70	37.43	32.02	33.19	37.13	38.01
5	8.48	3.36	29.09	22.37	23.39	15.94	13.30	36.40	34.06
6	2.05	0.00	27.34	22.37	23.39	15.94	13.30	34.94	32.46
7			21.49	0.15	0.00	0.00	3.95	34.80	30.99
8			17.84	0.15	0.00	0.00	3.95	26.90	29.82
9			12.87					23.98	20.18
10			10.23					22.37	16.96
11			2.34					14.33	16.08
12			0.15					12.72	9.06
13			0.00					3.65	2.78
14								3.07	0.58
15								0.00	0.15
Surface sites	4	3	5	6	6	4	4	9	8

Table 5.2: Accessibility (sodium probe) of surface protons in gibbsite. The two columns for the (101) and (10$\bar{1}$) faces list the results for the two possible shifts.

are a little under 50%, with most being in the 30%-40% range. Note that for the (110) face in the gibbsite morphology, the corrected cut described in the previous chapter was used. It is apparent that one distinguishing character between surface and bulk sites should be a large drop in the percentage accessibility. Given that the highest accessibility for most of these surfaces is ~40%, it is questionable whether a proton on the same surface with less than half this accessibility (for example) should also be treated as a true 'surface' site. Hence, for each face, the cutoff was calculated to be the accessibility of the most exposed proton, minus 20%.

5.2.2 Defect surface energy

Surface energy is computed by subtracting the energy of a slice at the surface from an equivalent slice in the bulk. This calculation is automated by the MARVIN [141] program. In the current study, the energy of the slice in the bulk will not be comparable when performing the surface defect calculation, as the bulk slices will contain no defects. Manual calculation was thus required. Using a Langmuir relation, the defect surface energy E_{surf}^{defe} for each face in the morphology can be written as [150],

	Hydrogen accessibility (%)								
Candidate	(002)	(200)	(110)	(101)		(10$\bar{1}$)		(112)	(11$\bar{2}$)
1	29.24	47.22	43.71	33.33	33.04	38.6	38.45	41.08	39.18
2	25.58	34.94	41.37	32.60	33.04	38.6	38.16	36.55	34.94
3	26.02	23.39	40.50	31.58	33.04	27.49	27.78	32.89	33.19
4	24.12	5.26	33.92	31.14	32.31	26.75	26.61	32.31	31.87
5	1.32	3.36	22.95	17.11	17.11	9.94	3.80	32.16	30.85
6	0.73	0.44	20.76	17.11	17.11	9.94	3.65	31.29	27.63
7			13.45	0.00	0.00	0.00	0.00	30.41	26.90
8			11.11					21.93	25.88
9			7.02					17.84	14.47
10			1.17					16.96	11.84
11			0.44					8.63	13.30
12			0.00					2.49	1.32
13								1.02	0.15
14								0.44	0.00
15								0.00	
Surface sites	4	2	4	6	6	4	4	8	8

Table 5.3: Accessibility (potassium probe) of surface protons in gibbsite. The two columns for the (101) and (10$\bar{1}$) faces list the results for the two possible shifts.

$$E_{surf}^{defe} = E_{surf}^{pure} + \chi . E_{rep} \tag{5.1}$$

The term E_{surf}^{pure} represents the surface energy of the pure surface, and was taken from the calculations described in Section 4.2. The fractional quantity χ is the defect surface coverage, and the replacement energy E_{rep} is the extra energy (per unit surface area) required to form a maximally covered defective surface from a pure surface. Thus, it might be expected that the replacement energy is the sum of the energies computed for each isolated defect. In such a scheme, if a value for χ were selected such that only one proton per repeat unit was replaced, the change in surface energy would be the average expected if all candidate sites were equally likely to be the defect.

This approach is flawed, however, as surfaces with low defect densities might be expected to favour sites with the smallest associated defect replacement energies. Thus, a new quantity E_{sub} is defined as the energy required to substitute a particular proton on the surface of gibbsite with a cation from solution. However, a given surface coverage value may correspond to a non-integer number of defect sites. In this work, each surface coverage value will be assumed to correspond to a number of whole sites and one partial site. For example, 30% coverage on

the $(11\bar{2})$ plane, which has 8 defect sites, implies 2.4 'effective' sites. This is broken into a number of 'whole' sites (2) plus a 'fractional' site (0.4). An effective E_{rep} can be calculated in such a way that $\chi.E_{rep}$ is the energy per unit surface area required to replace the two protons with the lowest associated E_{sub}, plus a weighted contribution from the third lowest. In the example selected this weight would be 0.4.

Calculation of the substitution energy was accomplished with the aid of the following expression,

$$n.Al(OH)_{3\,(s)} \;+\; MOH_{(aq)} \;\xrightarrow{\Delta E}\; (n-1)[Al(OH)_3].Al(OH)_2MO_{(s)} \;+\; H_2O_{(l)}$$

This describes the substitution energy required ($E_{sub} \equiv \Delta E$) to replace a proton on a gibbsite surface with a cation M^+ from solution. Thus, separate runs for each face in the morphology and each candidate site were required. The MARVIN code was employed to relax the gibbsite surfaces (with the substituted defect) and compute E_{R1}^{defe}, the total energy of region 1. The total energy of region 1 for the pure material E_{R1}^{pure} was taken from the calculations in Chapter 4.

Unfortunately, the difference between the two region 1 energies is not equal to E_{sub}. In order to compute the substitution energy, a suitable Born-Haber cycle was developed (Figure 5.2). The values of each reaction shown in this figure are listed below.

$$Na^+_{(g)} + OH^-_{(g)} \xrightarrow{\Delta E_1 = -936.7\ kJ\,mol^{-1}} Na^+_{(aq)} + OH^-_{(aq)}$$

$$K^+_{(g)} + OH^-_{(g)} \xrightarrow{\Delta E_2 = -852.9\ kJ\,mol^{-1}} K^+_{(aq)} + OH^-_{(aq)}$$

$$O^{-1.4}_{(g)} + H^{0.4}_{(g)} \xrightarrow{\Delta E_3 = -523.0\ kJ\,mol^{-1}} OH^-_{(g)}$$

$$2OH^-_{(g)} \xrightarrow{\Delta E_4 = 1133\ kJ\,mol^{-1}} O^{2-}_{(g)} + H_2O_{(g)}$$

$$H_2O_{(g)} \xrightarrow{\Delta E_5 = -44.00\ kJ\,mol^{-1}} H_2O_{(l)}$$

These enthalpies were determined with the aid of the values shown in Table 5.4. In the table, ΔH_f is the enthalpy of formation, E_{latt} represents the lattice energy, ΔH^j_{ip} is the j^{th} ionization potential, ΔH_{eg} is the electron gain enthalpy and ΔH_{sub} is the sublimation enthalpy. The enthalpies of hydration at infinite dilution were calculated by employing the cycle shown in Figure 5.3. Hence, ΔE_1 and ΔE_2 are

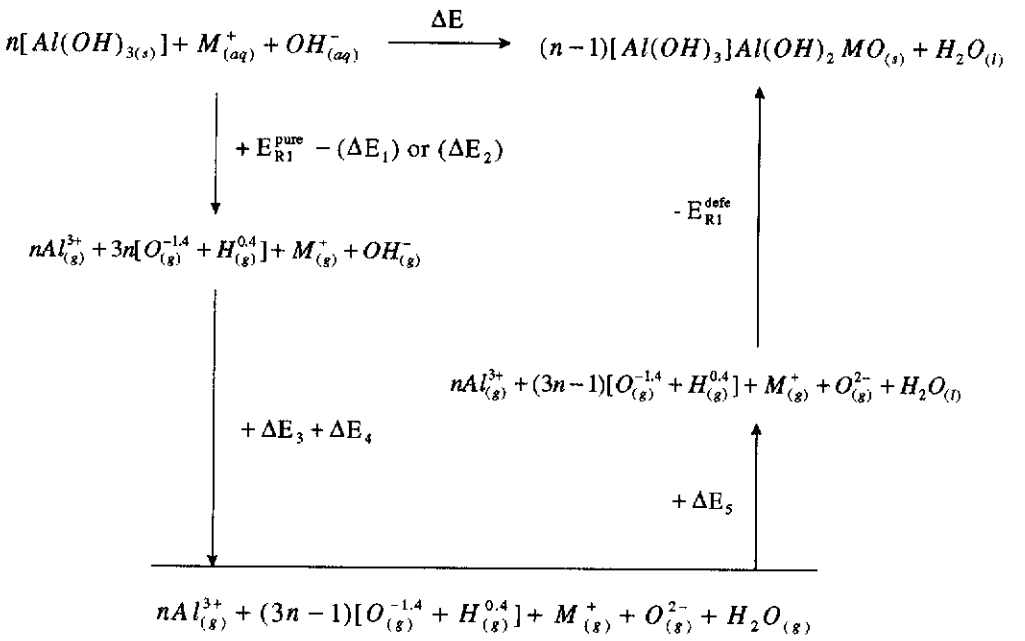

Figure 5.2: A Born-Haber cycle for the evaluation of the substitution energy ($E_{sub} \equiv \Delta E$). The diagram is written so that ΔE is the sum of the quantities on the lower reaction pathway.

Quantity	Intermediate steps	Value ($kJ\ mol^{-1}$)
$\Delta H_f[Na^+_{(aq)}]$	-	-240.34
$\Delta H_f[K^+_{(aq)}]$	-	-252.14
$\Delta H_f[OH^-_{(aq)}]$	-	-230.015
$\Delta H^{(1)}_{ip}[Na]$	-	495.84
$\Delta H^{(1)}_{ip}[K]$	-	418.81
$\Delta H_{sub}[Na]$	$H_f[Na_{(g)}]$	107.5
$\Delta H_{sub}[K]$	$H_f[K_{(g)}]$	89.0
$\Delta H_f[OH_{(g)}]$	-	39.3
$\Delta H_{eg}[OH]$	-	-176.34
$\Delta H_f[OH^-_{(g)}]$	$\Delta H_f[OH_{(g)}] + \Delta H_{eg}[OH]$	-137.03
$\Delta H_f[Al_{(g)}]$	-	330.4
$\sum_{i=1}^{3} \Delta H^{(i)}_{ip}[Al]$	-	5139.0
$\Delta H_f[Al^{3+}_{(g)}]$	$\Delta H_f[Al_{(g)}] + \sum_{i=1}^{3} \Delta H^{(i)}_{ip}[Al]$	5469.4
$\Delta H_f[H_2O_{(g)}]$	-	-241.826
$\Delta H_f[H_2O_{(l)}]$	-	-285.830
$\Delta H_f[Al_2O_3]$	-	-1675.7
$E_{latt}[Al_2O_3]$	-	15916

Table 5.4: Enthalpy values used in the Born-Haber cycles, together with any intermediate steps used to derive them. All quantities in this table were obtained from Lide [233].

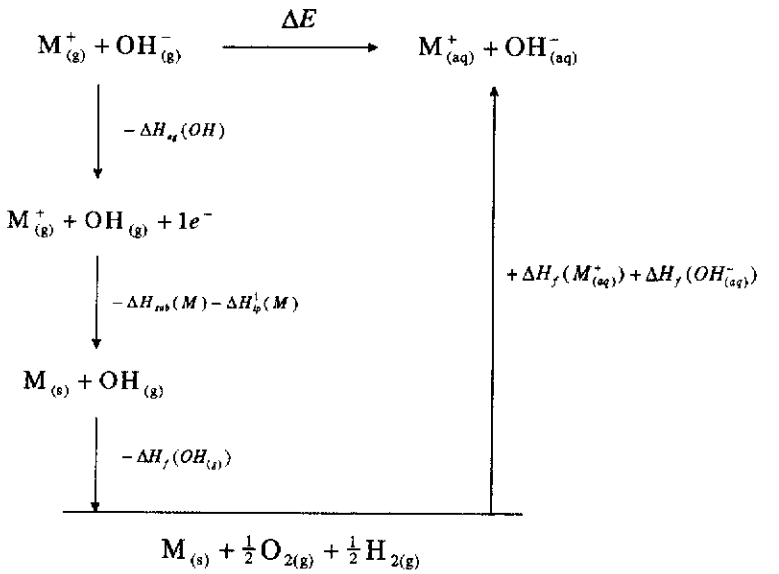

Figure 5.3: A Born-Haber hydration enthalpy cycle, written so that ΔE is equal to the sum of the enthalpies along the lower reaction pathway.

evaluated by adding all the quantities along the lower reaction pathway, with Na and K substituted respectively for M. For the $OH^-_{(g)}$ bond energy (ΔE_3), the minimum value of the Morse potential was used. The value of ΔE_4 was calculated using,

$$Al_2O_{3\,(s)} \xrightarrow{E_{latt}} 2Al^{3+}_{(g)} + 3O^{2-}_{(g)}$$

which yields,

$$\begin{aligned} \Delta H_f[O^{2-}_{(g)}] &= \tfrac{1}{3}(E_{latt}[Al_2O_3] + \Delta H_f[Al_2O_{3(s)}] - 2\Delta H_f[Al^{3+}_{(g)}]) \\ &= 1100\,kJ\,mol^{-1} \end{aligned}$$

thus,

$$\begin{aligned} \Delta E_4 &= \Delta H_f[O^{2-}_{(g)}] + \Delta H_f[H_2O_{(g)}] - 2\Delta H_f[OH^-_{(g)}] \\ &= 1133\,kJ\,mol^{-1} \end{aligned}$$

Finally, ΔE_5 is simply the difference between the enthalpies of formation of gaseous and liquid water,

$$\begin{aligned} \Delta E_5 &= \Delta H_f[H_2O_{(g)}] - \Delta H_f[H_2O_{(l)}] \\ &= 44.00\,kJ\,mol^{-1} \end{aligned}$$

(hkl)	Surface area ($Å^2$)
(002)	44.0
(200)	48.9
(110)	98.7
(101)	69.1
$(10\bar{1})$	62.3
(112)	135.6
$(11\bar{2})$	128.8

Table 5.5: A listing of the surface area of a single repeat unit, for each face in the gibbsite morphology.

In addition to gibbsite potentials, the defect minimization and subsequent energy calculations require functions describing cation interactions. In Section 3.4, suitable potentials for sodium and potassium interactions with hydroxides were developed. The metallic potentials have been fitted to more than one structure, and all non metallic terms are transferable between the 7 compounds with good agreement. Thus, the relevant cation potentials from Table 3.2 were employed in the calculations described in this chapter.

5.3 Results

5.3.1 Defect energy calculations

The model employed by MARVIN creates a surface by infinitely replicating the corresponding repeat units in two dimensions. The surface configuration of the repeat unit was examined for proton accessibility. Presented in Table 5.5 are the surface areas corresponding to each repeat unit of all faces in the gibbsite morphology. As any impurity substitution will also be repeated, the question of defect interaction must be considered. A surface area of approximately $100Å^2$ per defect was judged sufficient for a non-interacting single isolated defect to be studied. Thus, for the (002), (200), (101) and $(10\bar{1})$ faces, only one defect per four surface repeat units (in a 2x2 configuration) was created; thus quadrupling the surface area and the number of candidate proton sites (note that these extra sites will be identical by symmetry to those on the original repeat unit). In addition, as the (101) and $(10\bar{1})$ faces possess two possible cuts with similar surface energies, both of these were investigated. Finally, the (110), (112) and $(11\bar{2})$ surfaces possessed surface areas that were judged sufficiently large that a single supercell was not required.

	E_{sub} (kJ mol^{-1})								
Surface site	(002)	(200)	(110)	(101)		(10$\bar{1}$)		(112)	(11$\bar{2}$)
1	260.0	278.4	322.5	207.1	241.2	145.2	488.7	238.9	389.7
2	292.8	204.7	323.7	207.1	241.2	141.7	502.0	198.3	443.1
3	293.0	134.0	256.9	189.8	205.2	218.8	392.2	207.7	328.0
4	279.7		138.7	189.8	205.2	218.8	392.1	213.5	408.2
5			108.7	199.4	210.2			221.9	374.5
6				199.4	210.2			169.4	442.5
7								247.9	213.4
8								463.5	194.6
9								215.2	

Table 5.6: Substitution energies for the sodium defect surface. The ordering of defect sites corresponds that in Table 5.2.

	E_{sub} (kJ mol^{-1})								
Surface site	(002)	(200)	(110)	(101)		(10$\bar{1}$)		(112)	(11$\bar{2}$)
1	762.4	549.4	497.6	316.0	579.8	420.9	521.7	411.6	401.3
2	381.9	300.3	467.1	318.8	597.0	420.9	520.2	310.7	309.0
3	527.8		401.8	510.9	315.7	359.3	418.1	301.8	375.1
4	414.6		326.6	510.8	315.7	359.9	365.3	244.9	396.3
5				654.5	665.8			333.7	331.4
6				654.5	665.8			429.3	411.9
7								359.9	519.0
8								452.6	427.2

Table 5.7: Substitution energies for the potassium defect surface.The ordering of defect sites corresponds to that in Table 5.3.

Defects were created by substituting each candidate proton and the oxygen it was bonded to with a cationic species (Na^{+}or K^{+}) and an O^{2-} ion respectively. Replacement of the oxygen was required due to charge neutrality requirements, as a partial charge model was used for the hydroxyl component of gibbsite. Impurity substitution was made on the relaxed surfaces from the equilibrium morphology prediction (Section 4.2). The defect surfaces constructed were then allowed to relax to a minimum, where the final region 1 energy was recorded. For each face, the region 1 energy was combined with the enthalpies of the reactions shown previously to determine the direct substitution energy E_{sub}. These values were recorded in Table 5.6 for sodium, and Table 5.7 for potassium. The scheme described in Section 5.2.2 was then applied to calculate an effective E_{rep} and thus the defect surface energy for a range of surface coverage values.

For the (101) face, neither of the two possible shift values possessed lower surface energies for all fractional coverage values. Hence, the mean defect surface

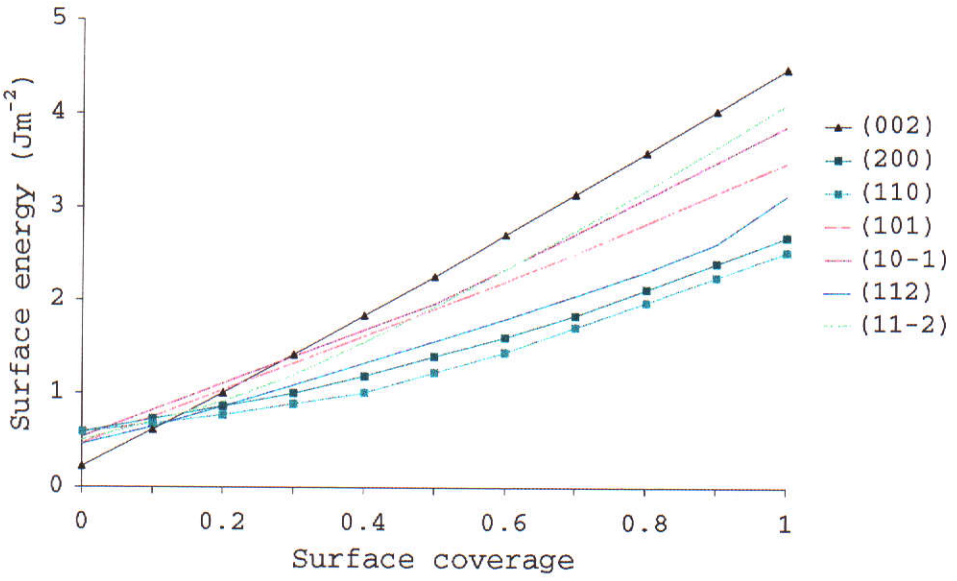

Figure 5.4: Sodium impurity surface energies. For clarity, the basal and prismatic faces have been drawn with line markers.

energy of the two shifts was plotted; yielding an estimation of the average morphological importance for the (101) face. This procedure was also required for the $(10\bar{1})$ face. A plot of the resulting defect surface energy for sodium is displayed in Figure 5.4, and for potassium in Figure 5.5. These figures show a degree of similarity, in particular for the low to medium surface coverage regime, as both the prismatic faces have lower slopes than all the other faces. Thus, assuming approximately uniform coverage, these plots suggest that cation incorporation will tend to have an elongating effect on the morphology, as the surface energy of the prismatic faces will rise the least quickly.

5.3.2 Predicted morphologies

Unfortunately, it is not known to what extent sodium and potassium are distributed over the respective faces in the morphology. Assuming an equal 10% coverage, the resulting equilibrium morphologies are shown in Figure 5.6. Clearly, cation incorporation assists in explaining the underestimation of the prismatic faces by the earlier equilibrium morphology prediction, shown in Figure 4.6. As a first step toward examining the consequence of different levels of defect coverage, the experimental differences in incorporation will be noted. Data from Lee *et al.*

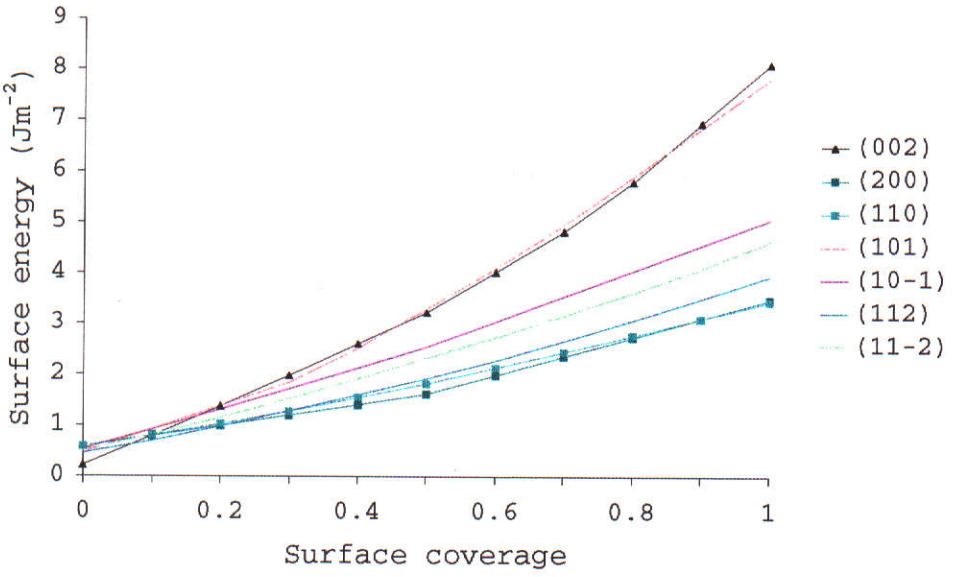

Figure 5.5: Potassium impurity surface energies. For clarity, the basal and prismatic faces have been drawn with line markers.

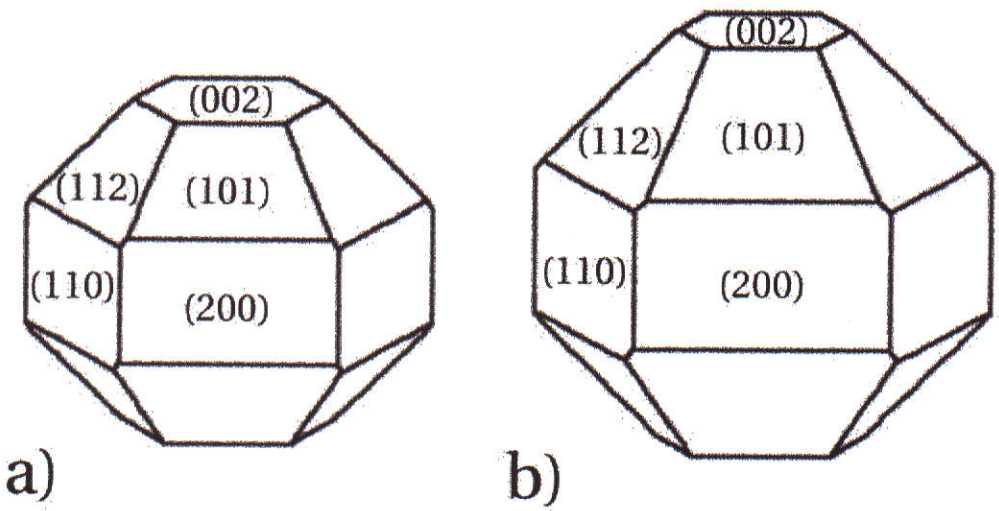

Figure 5.6: Defect morphologies for a) sodium incorporation and b) potassium incorporation. Both are determined from a uniform 10% surface coverage.

	E_{sub} (Jm^{-2})								
Site	(002)	(200)	(110)	(101)		$(10\bar{1})$		(112)	$(11\bar{2})$
1	0.245	0.114	0.183	0.114	0.123	0.0944	0.261	0.207	0.251
2	0.264	0.173	0.233	0.114	0.123	0.0968	0.261	0.243	0.275
3	0.276	0.236	0.432	0.120	0.126	0.146	0.326	0.254	0.423
4	0.276		0.543	0.120	0.126	0.146	0.335	0.261	0.483
5			0.545	0.124	0.145			0.264	0.502
6				0.124	0.145			0.272	0.526
7								0.293	0.570
8								0.304	0.571
9								0.568	

Table 5.8: Ranked sodium substitution energies. Note that for those faces with 2x2 surface constructions, only the defect energies from one surface repeat unit are listed as the other will be identical by symmetry.

	E_{sub} (Jm^{-2})								
Site	(002)	(200)	(110)	(101)		$(10\bar{1})$		(112)	$(11\bar{2})$
1	0.360	0.255	0.549	0.190	0.190	0.239	0.243	0.300	0.398
2	0.391	0.466	0.676	0.192	0.190	0.240	0.279	0.370	0.427
3	0.498		0.786	0.307	0.348	0.280	0.347	0.380	0.484
4	0.719		0.837	0.307	0.359	0.280	0.348	0.409	0.511
5				0.393	0.400			0.441	0.517
6				0.393	0.400			0.504	0.531
7								0.526	0.551
8								0.554	0.669

Table 5.9: Ranked potassium substitution energies. Note that for those faces with 2x2 surface constructions, only the defect energies from one surface repeat unit are listed as the others will be identical by symmetry.

[152] clearly indicates that cation incorporation is considerably higher for sodium than for potassium, as shown in Table 5.1. This suggests that an alternate approach may be required if the surface coverage is approximately proportional to the bulk incorporation.

As an estimation of the effects of higher concentrations of defects, the substitution energies shown in Table 5.6 and Table 5.7 were adjusted according to the appropriate surface area values. These were then ranked from lowest to highest and displayed in Table 5.8 (sodium) and Table 5.9 (potassium). Note that the (002), (200), (101) and $(10\bar{1})$ surface areas are quadruple the values shown in Table 5.5, due to the 2x2 construction used in the actual calculations. The resulting values calculated represent the average energy per unit surface area required to create an impurity substitution at a given proton site. Surface coverage may be

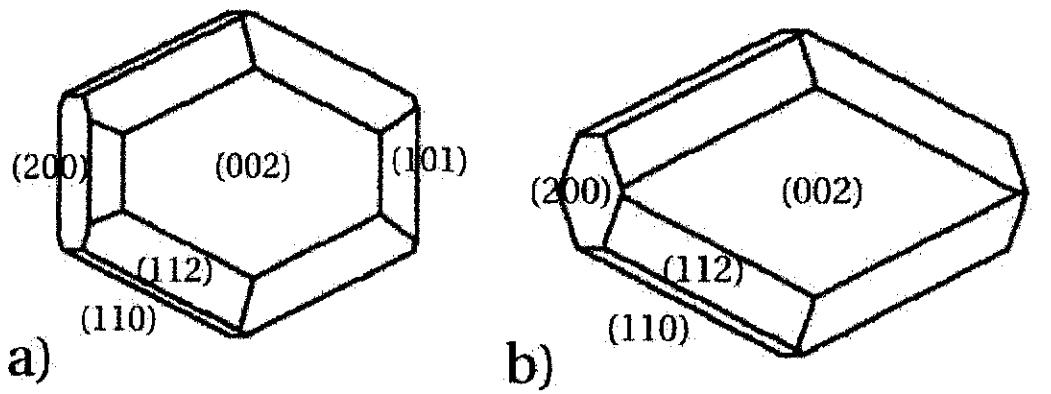

Figure 5.7: Sodium defect morphologies with increasing levels of surface coverage, 10% on a) and 20% on b) for those faces with the smallest defect energy per unit surface area; namely, {101}, {10$\bar{1}$} and {200}. All other faces have 5% defect coverage.

assumed to occur in such a way as to minimize the total defect energy. For sodium incorporation, the energy per unit surface area required to form (for example) the first four defects is significantly lower for the {200}, {101} and {10$\bar{1}$} faces. Consequently, these surfaces would be expected to experience more incorporation (at these sites) in a higher defect density regime. Two separate morphologies with 10% and 20% coverage for the {101}, {10$\bar{1}$} and {200} faces are illustrated in Figure 5.7, with 5% defect coverage applied to all the other faces. Clearly, the habit is approaching that of a truncated diamond; a morphology commonly found in sodium aluminate solutions [152][217]. An image of this experimental form has been reproduced in Figure 5.8.

Consideration of the average defect replacement energies per unit surface area for potassium should not be necessary. This is due to the low level of incorporation, which suggests that defects may be treated as completely isolated to good approximation. Thus, the relative rankings of the average energy to substitute a single impurity should provide an indication of relative coverage. It is notable that the direct potassium substitution energies (*i.e.* without surface area consideration) shown in Table 5.7 are generally higher for the basal plane. This suggests that incorporation on the (002) plane is considerably less, and perhaps even unfavourable, compared to the other faces. In addition, a similar argument will apply for sodium defects. Sodium and potassium defect morphologies with 5% coverage on the basal plane, and 10% on the other faces are shown in Figure 5.9. These illustrations show how the importance of the chamfering faces is reduced

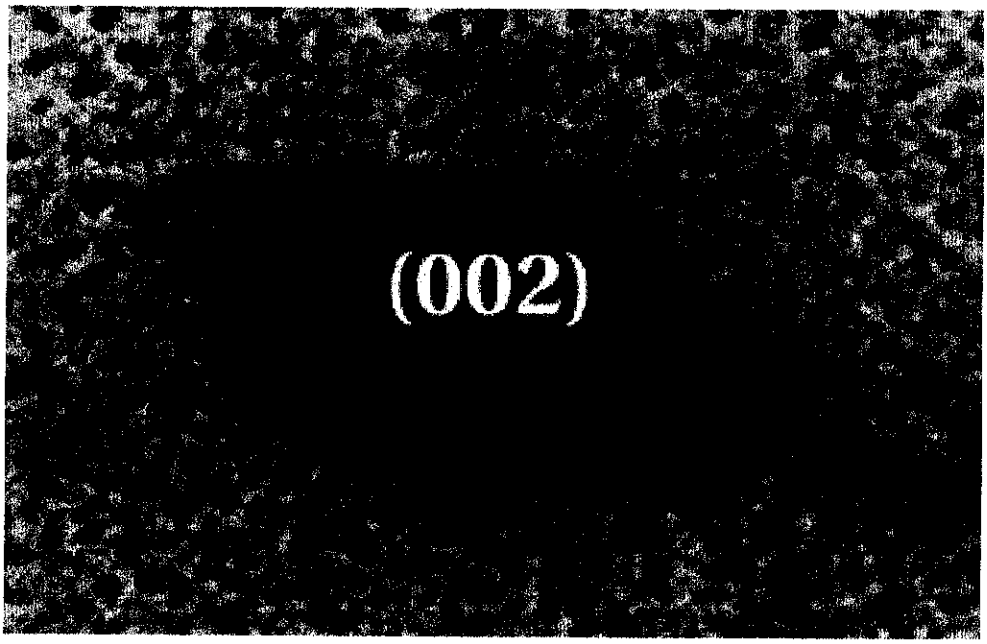

Figure 5.8: SEM image of gibbsite diamond morphology from Sweegers *et al.* [217]. The prismatic faces were identified as belonging to the {110} family.

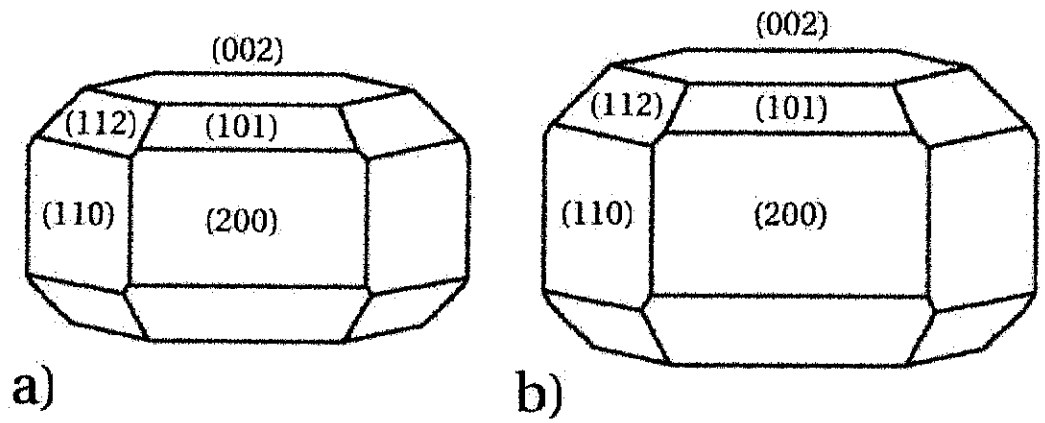

Figure 5.9: a) Sodium, and b) potassium defect morphologies, with 5% coverage on the basal plane. All other faces have 10% surface coverage.

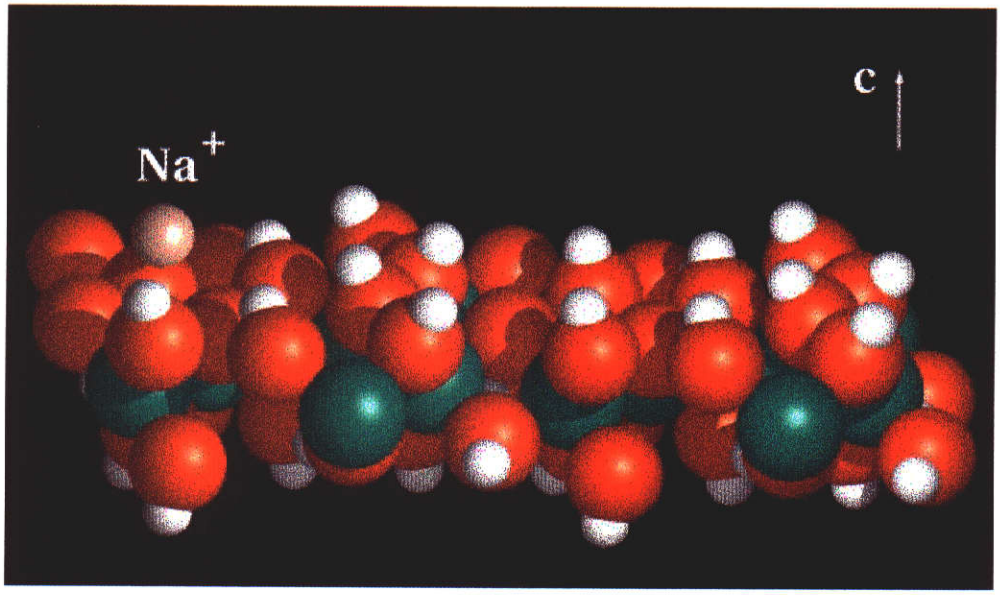

Figure 5.10: Relaxed (002) surface, containing a single sodium defect.

due to the promotion of the basal plane (*cf.* Figure 5.6).

5.3.3 Surface reorganization

An illustration of sodium and potassium defect configurations at one of the four proton sites on the basal plane is shown in Figure 5.10 and Figure 5.11, respectively. The cation positions of both optimized configurations are similar. However, there is noticeably more disturbance in the surface structure for the potassium defect. In particular, the hydroxyl orientations at the right end of the surface slab, which would be quite close to the periodic image of the potassium ion, have shifted noticeably. The magnitude of the disturbance in the crystal surface caused by the presence of a metallic cation is thus of some interest. The average displacements of atoms from their initial relaxed positions on the pure surface are shown in Table 5.10 for sodium defects, and in Table 5.11 for potassium. These were taken from the final relaxed (defect) structures of all candidate proton sites. The data show that the disturbance due to potassium is greater than sodium, which is expected from the respective ionic radii of the impurities. This provides some explanation as to why defect energy creation requirements are higher for potassium than sodium, and thus why the levels of incorporation of the respective cations are very different.

It is evident that the atoms on the (002) surface experience the least disturbance after cation substitution (Tables 5.10 and 5.11). The basal plane is also the only surface for which Al is six coordinate (all other faces have five coordinate

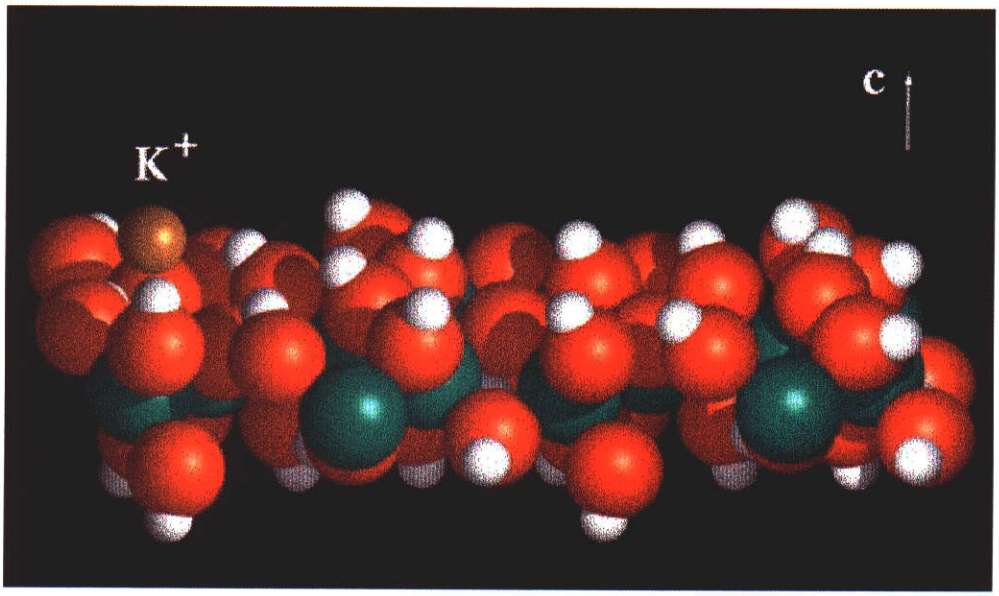

Figure 5.11: Relaxed (002) surface, containing a single potassium defect.

(hkl)	Mean atomic displacement (Å)		
	Al	O	H
(002)	0.018	0.020	0.035
(200)	0.044	0.048	0.065
(110)	0.059	0.068	0.10
(101)	0.061	0.065	0.087
$(10\bar{1})$	0.076	0.089	0.12
(112)	0.092	0.10	0.15
$(11\bar{2})$	0.12	0.15	0.22

Table 5.10: Surface atom distortion due to a sodium impurity.

(hkl)	Mean atomic displacement (Å)		
	Al	O	H
(002)	0.040	0.044	0.079
(200)	0.10	0.11	0.15
(110)	0.075	0.090	0.13
(101)	0.10	0.11	0.15
$(10\bar{1})$	0.11	0.13	0.18
(112)	0.11	0.13	0.20
$(11\bar{2})$	0.14	0.17	0.26

Table 5.11: Surface atom distortion due to a potassium impurity.

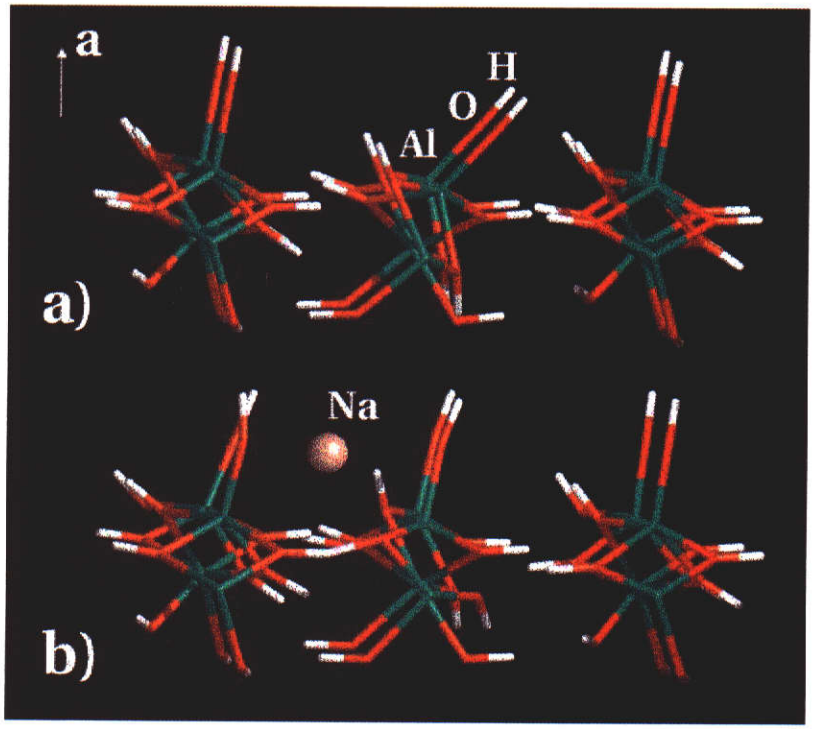

Figure 5.12: (200) surface of gibbsite, showing a) the pure surface and b) the minimized substitution defect position.

surface aluminium atoms). This suggests that the higher aluminium coordination on the basal face allows little hydroxyl reorientation to occur, due to high repulsion when disturbed from the 'normal' configuration. In contrast, the surfaces with only five coordinate aluminium atoms permit more reorientation of the surface hydroxyl groups. This greater freedom on the non-basal surfaces is also responsible for the lower replacement energies compared to the basal plane, as the cation to hydrogen repulsion and the oxygen to cation attraction contributions are more readily optimized. This can be seen in an image of the (200) surface in Figure 5.12, where the upper image shows the pure surface and the lower the minimized defect surface. Principally, the central row of hydroxyl groups have re-orientated to bring the oxygen atoms closer to the cation, which is embedded in the five coordinate surface (Figure 5.13).

5.4 Discussion

The predictions of this study are valid in the low to medium surface coverage regime, as the model has been defined such that defects do not interact. If approximately equal surface coverage is assumed, the effect of impurity substitution is to increase the importance of the prismatic faces. This consequently elongates

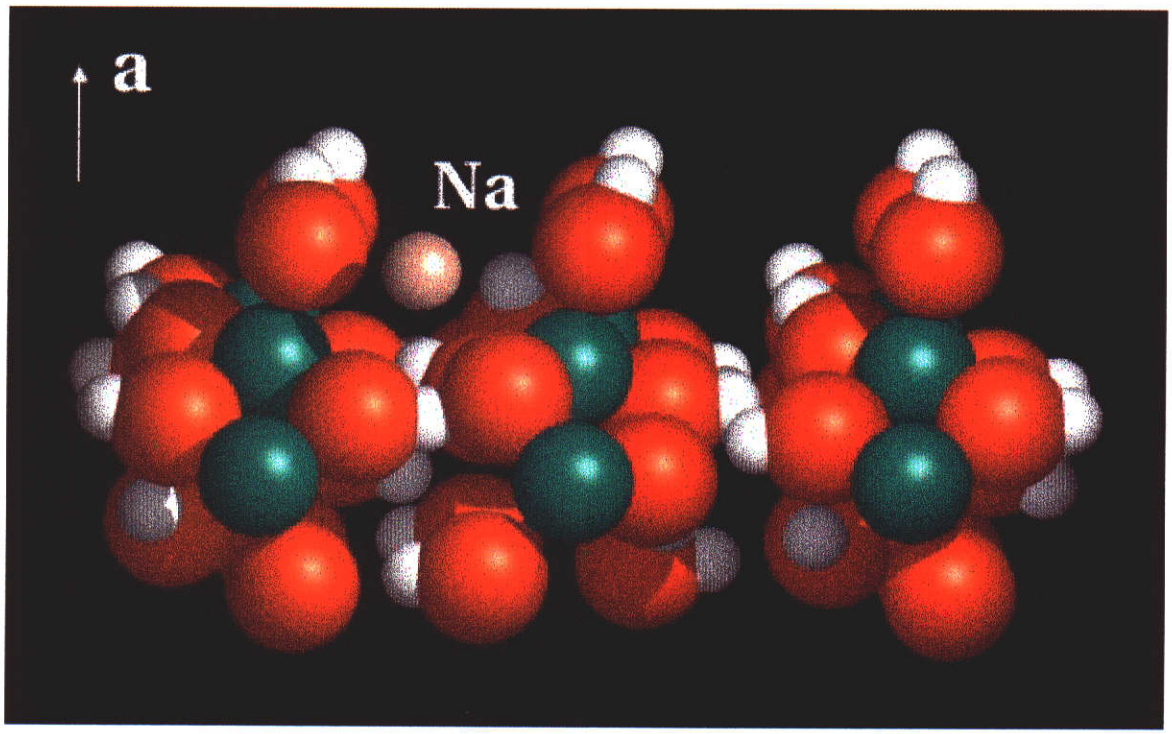

Figure 5.13: Embedded sodium defect on the (200) surface of gibbsite.

the equilibrium morphology, and helps to explain the previous underestimation in Section 4.2. This is in good agreement with experimental results, with observations by Mensah [91] and Lee *et al.* [152] noting the elongation of the prismatic faces in both potassium and sodium aluminate solutions. However, the disparity in the levels of incorporation for sodium compared to potassium necessitated a different means of estimating the relative levels of incorporation on each of the different faces in the morphology. This analysis has yielded a possible explanation for the diamond morphologies observed in sodium aluminate liquors. In addition, consideration of the magnitude of the defect replacement energy suggests that incorporation should reduce the importance of the chamfered faces, and promote the basal face. This accounts for the lower than expected morphological importance of the basal plane in the illustrations in Figure 5.6.

In both cation incorporation calculations, the defect surface energy tends to rise most quickly for the basal plane. Experimentally, the basal plane is more significant than this result might suggest. However, the basal plane does have significantly lower 'pure' surface energy. Hence, if coverage is low, the basal plane would still be dominant in the morphology. It may also be argued that as the basal plane generally has larger energy requirements for substitution to occur, it will experience the smallest defect coverage. In addition, the data from Tables 5.10 and 5.11 suggest that of all the faces, the basal plane is most suitable to

incorporating into the bulk any defects that happen to form on the surface. This might facilitate the overgrowth of any defects that do occur (so that the surface reverts back to its 'pure' state), resulting in a lower surface energy than expected from the data in Figures 5.4 and 5.5. As a consequence, the basal plane in the diamond morphologies in Figure 5.7 may be even more dominant. This would effectively reduce the importance of the {112} chamfering faces, and serve to explain why these faces are not noted in experimental observations.

Although this work predicts that gibbsite crystals from potassium aluminate solutions are longer than those grown from sodium aluminate, the elongation is not as distinct as experimental evidence suggests. If a growth model were utilized instead of the equilibrium model employed here, the change in attachment energy (and hence the growth rate) caused by the surface impurity could be considered. Clearly, the larger the disturbance in the surface caused by the defect the more subsequent growth unit incorporation will be retarded. If sufficient defects were formed, the face may even become completely blocked. Comparing Table 5.10 with Table 5.11, it is evident that potassium defects cause the greatest restructuring of the gibbsite surface. This suggests that face blocking is likely to be more dominant in potassium aluminate solutions over sodium aluminate solutions. A further argument for face blocking aiding the elongation of gibbsite morphology is the lower energy requirement for defect substitution on the prismatic faces. This should make defects preferentially form, and therefore block, the prismatic faces.

Chapter 6

Aluminate solution modelling

6.1 Background

Molecular modelling based on the structural energetics of the crystalline solid is useful in areas such as the prediction of final morphology. However, the pathway through which that result is achieved is also of importance. Primarily, it is the identification of the growth units that is lacking from the current understanding of the precipitation process of gibbsite, although several related pieces of information are known. The most abundant species in Bayer liquors is almost certainly the tetrahedrally coordinated $[Al(OH)_4]^-$ ion [61]. Since aluminium in gibbsite is octahedrally coordinated, there must be a coordination change from 4 fold to 6 fold coordination on crystallization. An answer to how this occurs has not been forthcoming. This is partly due to the extremely hostile nature of Bayer liquors, resulting in indirect and inconclusive experimental data. Computer modelling techniques present a complementary means to approach the problem.

Many recent studies on the nature of aqueous solutions have been accomplished using molecular dynamics. Work by Levitt *et al.* [234] documents a model (mainly for use with macromolecules) which reproduces selected physical properties of water. Laaksonen *et al.* [235] conducted an investigation of the methane-water system, employing molecular dynamics to generate spatial distribution functions describing the local solution structure. Investigations by Mancera *et al.* [236] focused on the effect of temperature on the aggregation of methane in aqueous solution. In addition, ionic systems have also been investigated. Ion selectivity was examined by plotting the free energy profile of cations to a crown ether in water [237]. In addition, a hybrid QM/MM technique has been employed to show that a classical approach is inadequate to describe the solvation of the doubly charged calcium ion in water [238].

Ideally, gibbsite precipitation may be studied by modelling the behaviour of gibbsite surfaces in contact with Bayer liquors. Unfortunately, running such a complex simulation for any reasonable length of time would require extensive computing power. An intermediate approach would be to model only the solution part using molecular dynamics. The aim of this investigation would be to note any structuring that may occur as a precursory stage to the formation of gibbsite growth units. The time scale of this experiment should be as long as possible, in order to accommodate the slow rate of gibbsite precipitation. Although there are hybrid quantum mechanics and molecular modelling techniques available, these would be impractical for the relatively lengthy simulation times required; at least until further data is obtained. Unfortunately, employing a molecular mechanics scheme to the dynamics will allow molecule flexing but not bond creation or annihilation. Thus, although new solution species will not be formed, any clustering that might precede such a transformation should still occur.

This chapter focuses on the structure of caustic aluminate solutions in relation to the precipitation of gibbsite. The composition of any given industrial Bayer liquor can be quite complex and varied, due to differences in both the raw ore and the precipitation procedure. Thus, fundamental investigations are often conducted using a synthetic Bayer liquor. These liquors are commonly prepared *via* the treatment of aluminium wire with a sodium hydroxide solution [61] [67], or the dissolution of gibbsite pellets in a hot caustic solution [152][239]. This suggests that the primary species desirable to model might be: $Al(OH)_4^-$, Na^+, OH^- and H_2O. The involvement of the cationic species in the gibbsite crystallization process was mentioned in section 1.2.2. As a result, studies have been conducted in order to probe the precise nature of the cationic influence on gibbsite morphology, in particular, the effects of replacing sodium with lithium, potassium and caesium [61][152].

An investigation of solution structuring in synthetic Bayer liquors was conducted with the aid of molecular dynamics simulations. Four separate simulations were performed, three of which were sodium aluminate solutions and the other a potassium aluminate solution. In addition, experimental data from vibrational spectroscopy experiments by Helen Watling [61] were compared to the results from the simulations. The experiment also involved separate spectral studies with Na^+, $K^+$and Cs^+ cations. As mentioned previously, there is some difficulty in unambiguously interpreting spectral data. Thus, combining computer simulation with experimental data should help improve current understanding of synthetic Bayer liquors.

6.2 Method

In this study, molecular dynamics simulations are employed to investigate possible ion associations in aluminate solutions (synthetic Bayer liquors), at concentrations of industrial relevance. All simulations were conducted on Silicon Graphics workstations, using the Discover package from Molecular Simulations Incorporated [192]. The ESFF potential set [240] was chosen to represent all force field interactions, since the system contains two metallic atom types in addition to oxygen and hydrogen. The molecular mechanics approach taken forbids the breaking or formation of bonds. However, a fully quantum mechanical simulation of this nature is not currently feasible, and the study was restricted to examine only possible solution structuring. The Verlet velocity method was employed to integrate the equations of motion. All simulations were conducted with the use of periodic boundary conditions; with the Ewald technique [241] employed to evaluate the van der Waals and Coulombic interactions.

Three systems were constructed, each possessing a composition similar to that of a synthetic Bayer liquor (7 $Al(OH)_4^-$, 5 OH^-, 12 Na^+ and 192 H_2O molecules at randomized locations). A target volume of about 6330Å^3 would thus yield a solution with $[Al(OH)_3]\approx$1.8M and $[NaOH]\approx$3.2M. An additional simulation, where K^+ was substituted as the cationic species, was also performed. The apparent experimental duplication was introduced as a result of the complexity of the systems under consideration. Dynamics simulations are sensitive to initial position and, if phase space is to be representatively sampled, multiple simulations with different starting configurations should eliminate problems if the system is not truly ergodic [115].

Due to the slow rate of gibbsite crystallization, it was decided that the simulations should be run for as long as possible; within the constraints of the available computational facilities. Large scale ($10^2 - 10^3$ atoms) molecular dynamics simulations are typically run for a net time of no more than about 300ps-400ps, which includes both data collection and equilibration phases [123][125][126]. In this work, the three sodium aluminate solutions were run for 500ps and the potassium aluminate solution for 300ps of simulation time. The step size for all stages was chosen to be 1fs, which is considered suitable for flexible molecular systems [114].

For each simulation, the molecules of the system were placed in a cubic lattice, the dimensions of which were selected to avoid overlap. A randomizing program, MDI (Appendix A.1.1) was written to select the location and orientation of each component, within the constraint of the overall required numbers of species. How-

ever, due to the use of a lattice for initial species placement, the simulation cell will be very ordered and unrepresentative of the liquid phase. Thus, it is convenient to quantify the ordering in a system of N particles (molecules or lone ions) in terms of a single parameter λ [114],

$$\lambda = \frac{1}{3N} \sum_{i=1}^{N} \sum_{j=1}^{3} cos\left(\frac{2\pi x_i^j}{a^j}\right) \tag{6.1}$$

where the i^{th} particle has a centroid located at (x_i^1, x_i^2, x_i^3) and the lattice vector is given by (a^1, a^2, a^3). For the case $|\lambda| = 1$, the system possesses maximal ordering, whereas the maximally disordered state is achieved when $|\lambda|$ is zero.

To obtain the correct cell density, a period of simulation at high pressure is required. Given the composition of the simulation cell, the enclosing cubic box is required to possess a dimension of L≈18.5Å, which corresponds to a density of approximately 1.2 g/cm^3. The concentration and density of this simulation therefore is similar to that of a synthetic Bayer liquor at $80°C$ [242]. A further period of simulation is also needed to properly equilibrate the liquid at the desired temperature. Equilibration was accomplished using the robust Andersen temperature control method [115], whilst the more realistic Berendsen method was used for the data collection [243]. Snapshots of the simulation, containing only positional data were stored every 100 steps. These were subsequently analyzed for possible solution structuring.

6.3 Sodium aluminate

6.3.1 Equilibration

The initial lattice configuration of the first simulation, generated by the MDI program, is displayed in Figure 6.1. In terms of purely structural arguments, it is evident that the simulation cell requires an equilibration period in order to approximate the liquid phase. In particular, the correct density and relaxation of the initial simulation lattice must be achieved. A short high pressure (constant NPT) dynamics run was conducted, with the configuration and density of each simulation step recorded. Subsequently, the configuration which best matched the required density was selected. However, the duration of this run is not long and the resultant simulation cell will still inherit some of the initial lattice ordering.

Further equilibration was attempted by conducting a 200ps NVT dynamics run. Sequential values of the order parameter during the first picosecond of

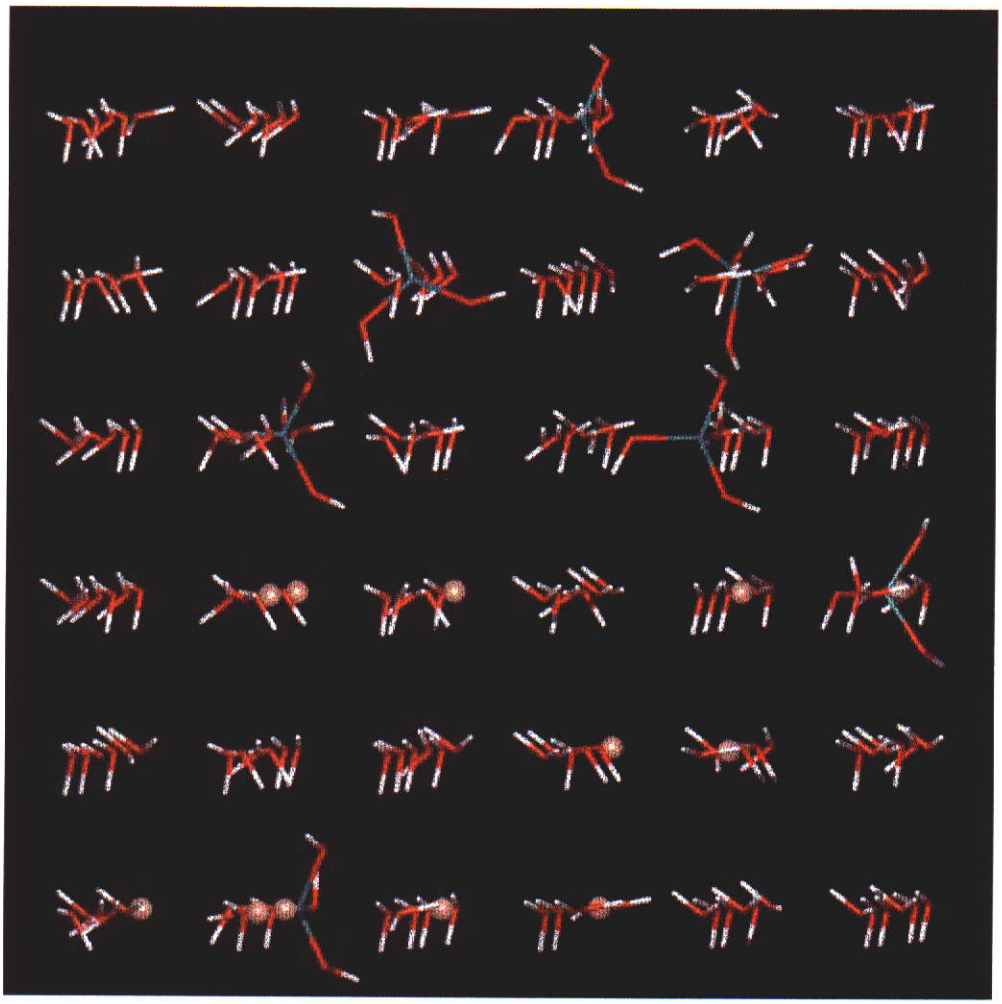

Figure 6.1: Cell configuration for simulation 1, illustrating the initial lattice placement.

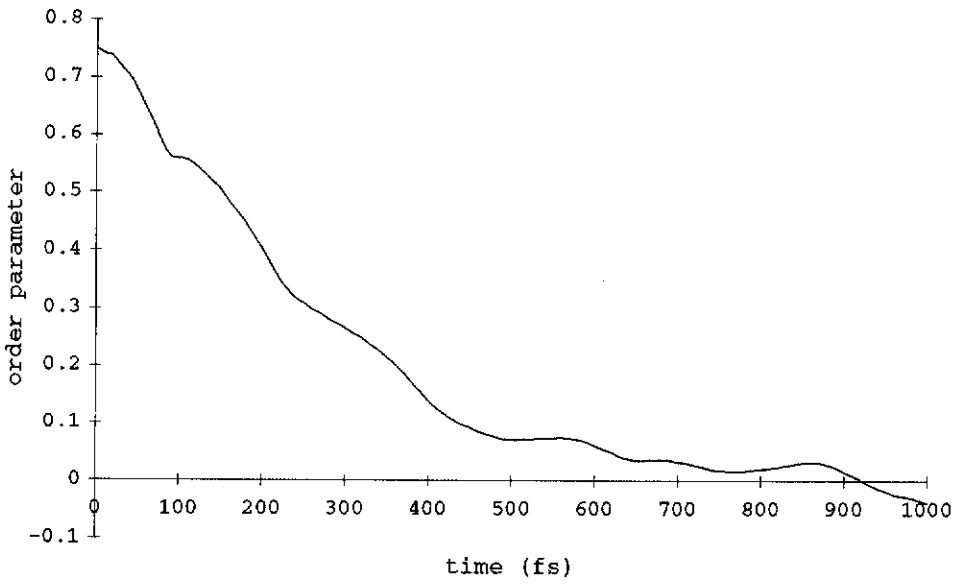

Figure 6.2: The first picosecond of equilibration, clearly demonstrating the lattice melting.

this equilibration period were plotted for the first simulation (Figure 6.2). The graph clearly indicates that lattice melting occurs quite quickly. To confirm that the structuring present in the initial lattice had vanished for all simulations, the order parameter was also calculated during the data collection period. A distribution (Figure 6.3) was constructed which has the expected centering around zero. Typically, for a system at equilibrium, λ experiences maximum fluctuations of the order of $\frac{1}{\sqrt{N}} \approx 0.07$ [114]. This confirms that the lattice structuring has sufficiently diminished by the time the simulation enters the data collection phase.

In an NVT ensemble, temperature is a calculated property of the atomic velocities and is regulated rather than fixed at some desired value. Thus, it is desirable to perform an equilibration of sufficient length that the temperature acquires a steady mean value. In the literature, synthetic Bayer liquors are typically held at a range of temperatures between $60°C$ and $100°C$ [67][91][244]. Hence, a target temperature of 350K with a tolerance of up to 20K was allowed. The duration of the equilibration period was 200ps. This was found to allow sufficient time for all lattice ordering to vanish, as well as the attainment of a steady mean temperature. The temperature distributions for all three simulations, taken throughout the equilibration period, are shown in Figure 6.4. From this plot, a very small percentage clearly fall outside the desired range of $\pm$20K. A plot of the tempera-

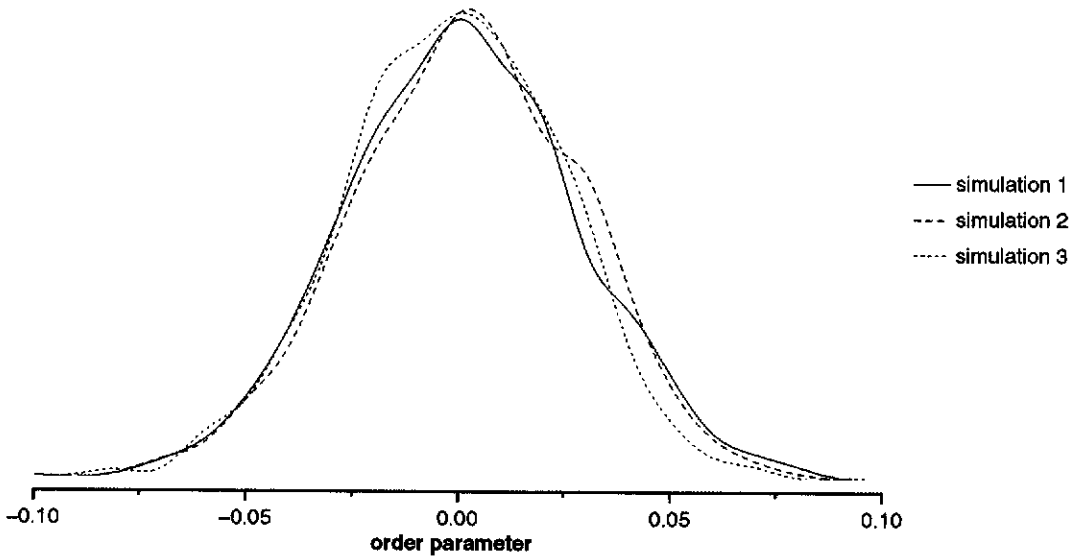

Figure 6.3: Sodium aluminate order parameter distributions during the data collection period.

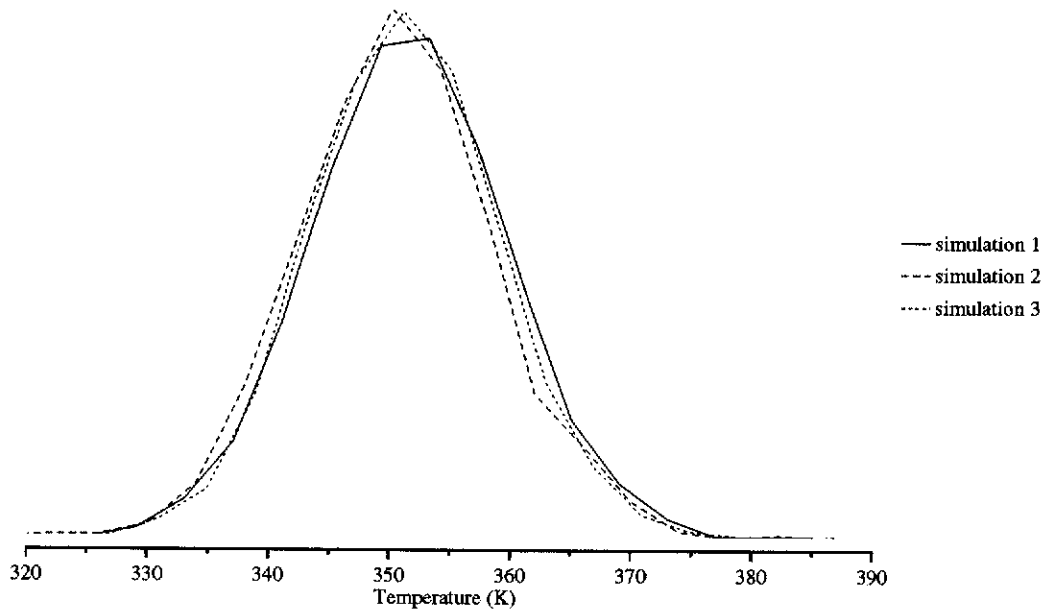

Figure 6.4: The temperature distributions of the sodium aluminate solutions, taken during the equilibration period.

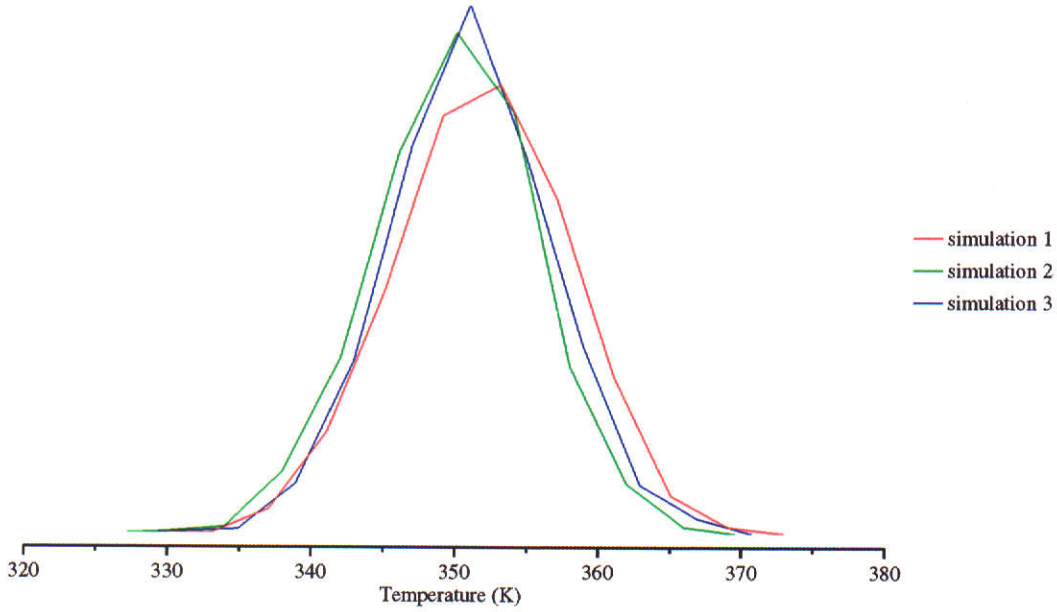

Figure 6.5: The temperature distributions of the sodium aluminate solutions, taken during the data collection period.

ture distributions for the data collection period is shown in Figure 6.5. Here the bulk of the temperatures fall within 15K of the target temperature range; this was considered acceptable.

To provide a contrast to Figure 6.1, a snapshot of simulation 1 after equilibration is shown in Figure 6.6. In addition to the considerations of temperature, density and lattice ordering discussed above; it is evident that components should be fairly uniformly dispersed. This has been achieved, enabling the data collection phase to begin from a reasonable initial solution state.

6.3.2 Data Collection

Of primary interest is the solution structuring, which is most directly probed with the use of radial distribution functions (RDFs). These functions were obtained by measuring the molecular and ionic separations and placing them in bins of 0.25Å resolution. The first measurements made were the distances from the center of each aluminate ion to all sodium cations. This data yielded the sodium aluminate RDFs shown in Figure 6.7. The large peaks in Figure 6.7, which are centered at an aluminate to cation separation of around 3.5Å, indicate that a significant degree of ion pairing is occurring. While the RDFs for simulation 1 and simulation 3 exhibit very similar behaviour, the second simulation experiences a greater degree of ion pairing. In addition, past the first large peak in the RDFs, the rest of the

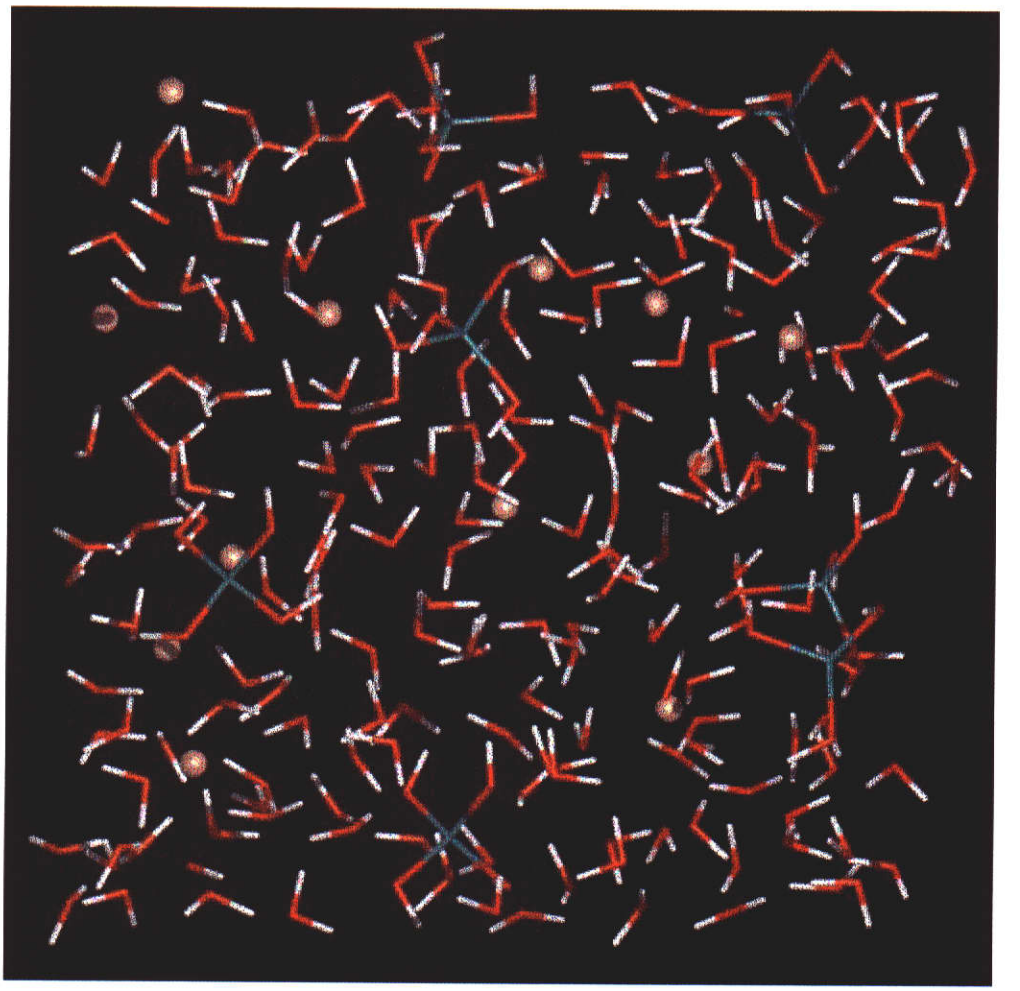

Figure 6.6: The cell configuration for simulation 1, immediately before the data collection phase.

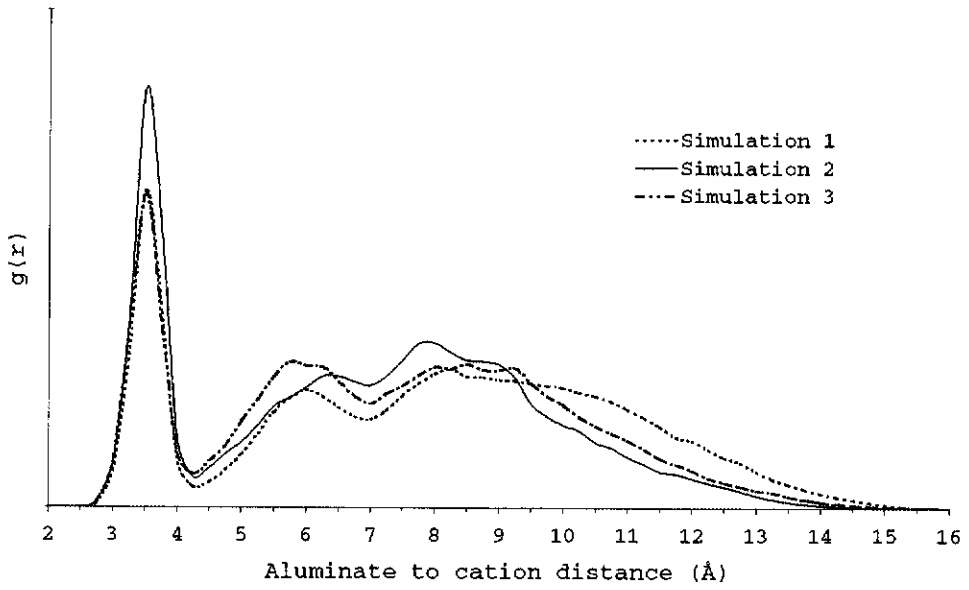

Figure 6.7: Radial distribution functions for the aluminium to sodium distances.

distributions appear to possess additional smaller peaks. While these are not as distinct, the deviation from a smooth curve suggests that some longer range solution structuring is also occurring.

In Figure 6.8, a graph of the number of cations paired with each aluminate, versus the proportion of occurrence is shown. This data was obtained by recording the number of cations paired to each aluminate, and summing over every frame in the data collection period. Thus, normalization of the graph is accomplished by dividing by the product of the number of aluminate ions (7) and the number of frames (3000). In addition, with the use of Figure 6.7, a paired cation was defined as being closer than 4.1Å to the Al atom of an aluminate ion. The data in Figure 6.8 indicate that the aluminate ions in simulations 1 and 3 are paired to an average of between two and three cations. For simulation 2, there is a large proportion of aluminates with three or more associated cations. As each cell has a composition of 7 aluminates and 12 sodium ions, the cations are clearly being shared by multiple aluminate ions which have come into proximity. This is particularly evident in simulation 2.

The sharing of cations between adjacent aluminates was investigated further by constructing radial distribution functions for aluminate Al-Al separations (Figure 6.9). Although all three of the plotted RDFs possess a peak at around 5Å, it is evident that simulation 2 experiences the greatest degree of aluminate 'clustering'.

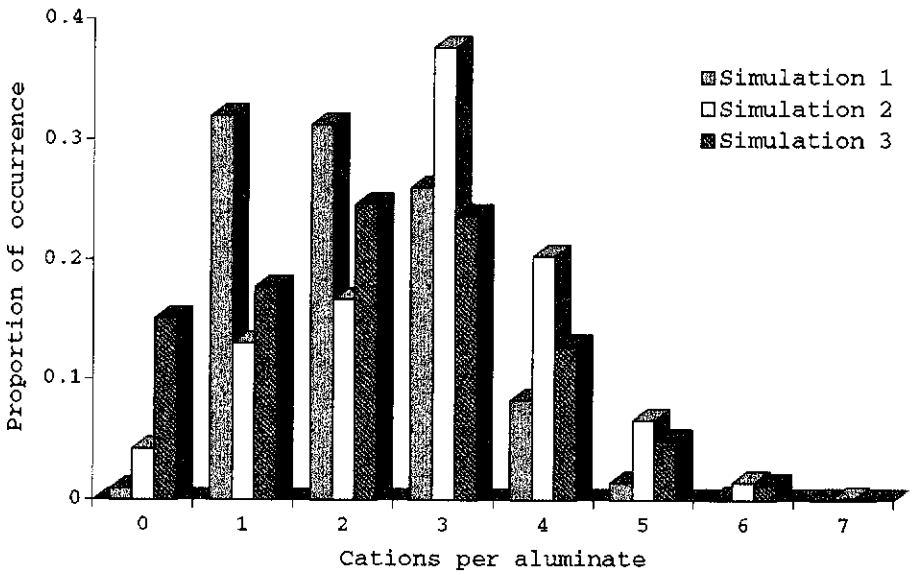

Figure 6.8: Proportion of clustering occurring between sodium and aluminate ions.

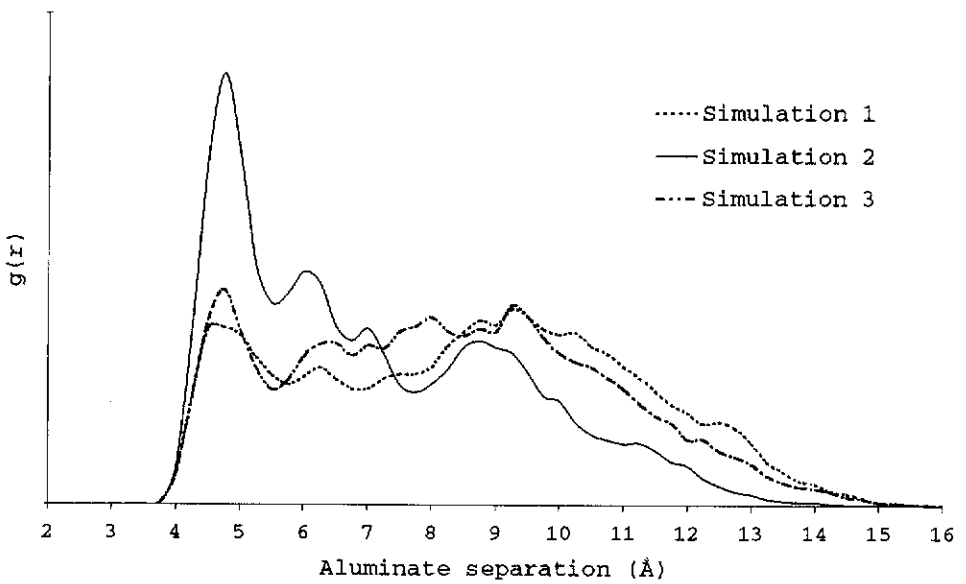

Figure 6.9: Radial distribution function for aluminium separations.

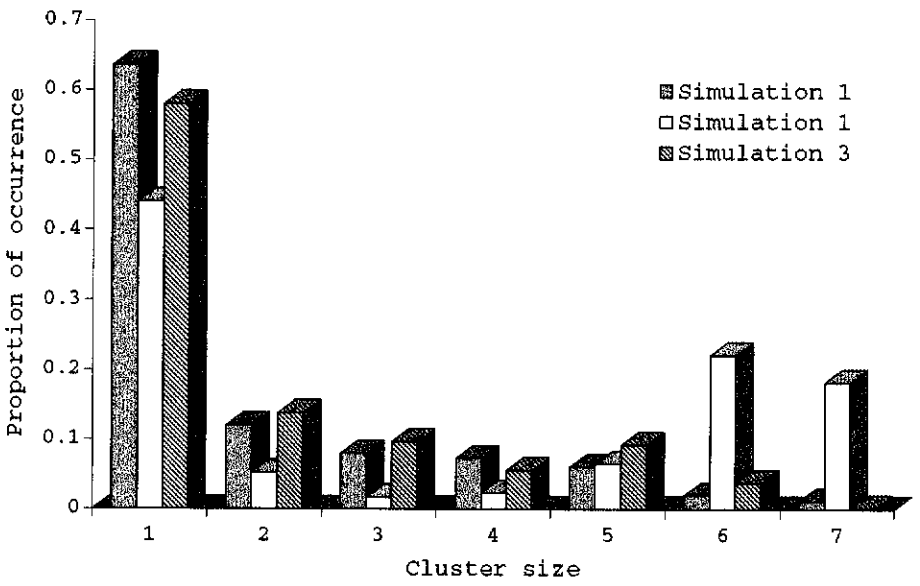

Figure 6.10: Proportion of clustering occurring between aluminate ions.

It is these clustered aluminates, each with their own shell of paired cations, that are responsible for the smaller peaks in Figure 6.7. Further information emerges from the aluminate cluster size distribution shown in Figure 6.10. Here, a cluster is defined as a group of linked aluminate monomers with Al-Al separations of not more than 5.5Å. This clustering data was collected by cumulatively recording the size of the group each aluminate belongs to in each simulation frame. Thus, the distribution was normalized with the same value employed to normalize the data in Figure 6.8. The aluminate clustering plot shows that simulations 1 and 3 experience relatively small amounts of clustering, with a large fraction of aluminates remaining 'isolated' during the simulation. However, the second simulation has formed a very large group that is stable enough to account for almost half of the simulation time. As the second simulation also possesses the highest number of shared cations, it is probable that these sodium to aluminate bridges are playing an important role in the formation of clusters.

In addition to the overall statistical information, the variance in cluster distribution as a function of time was also investigated. The cluster sizes present throughout each simulation were graphed. The cluster distributions in simulation 1 (Figure 6.11) confirm that most of the aluminates are 'isolated'. The clusters that do form however, occur in a randomly distributed manner. The larger degree of clustering in simulation 2 is evident in Figure 6.12. Here the large groupings

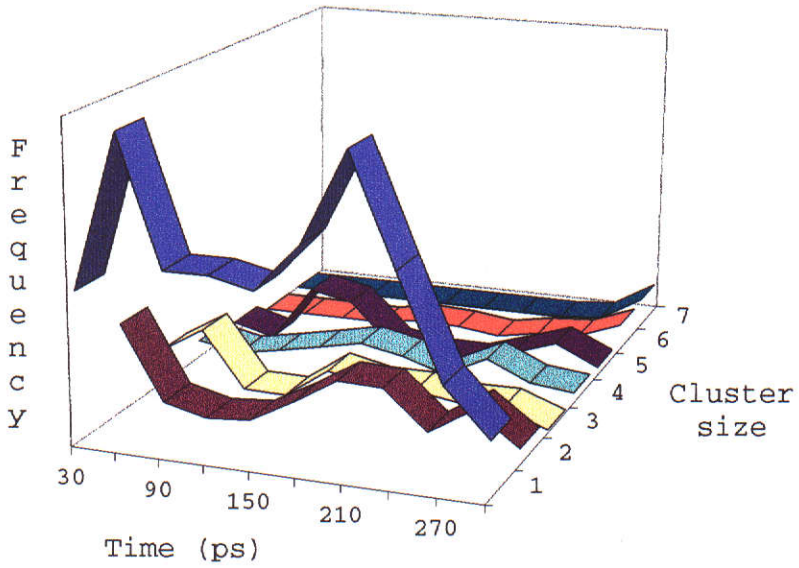

Figure 6.11: Frequency of occurrence of aluminate clusters of given size, throughout simulation 1.

of six and seven aluminates are stable enough to dominate the latter half of the simulation. The third simulation (Figure 6.13) is similar to the first; however, it is notable that larger clusters of six aluminates briefly form throughout the run. This can also be seen from Figure 6.10.

6.4 Ion pairing and cationic species

To examine the possible effects of replacing sodium with a different cationic species, a simulated potassium hydroxide solution was also investigated. Equilibration of the simulated liquor was accomplished in the same fashion as for the prior sodium aluminate simulations. However, after the 200ps of equilibration, only 100ps of data collection was conducted. This shorter run was considered sufficient to extract the desired information. The order parameter distribution for the data collection period is shown in Figure 6.14. This shows that the simulation has lost all lattice structure. The temperature distribution is shown in Figure 6.15. This shows that the simulation achieved a steady state temperature within the allowed tolerance.

The potassium to aluminate radial distribution function was constructed (Figure 6.16). This plot has a similar form to those shown in Figure 6.7, although the

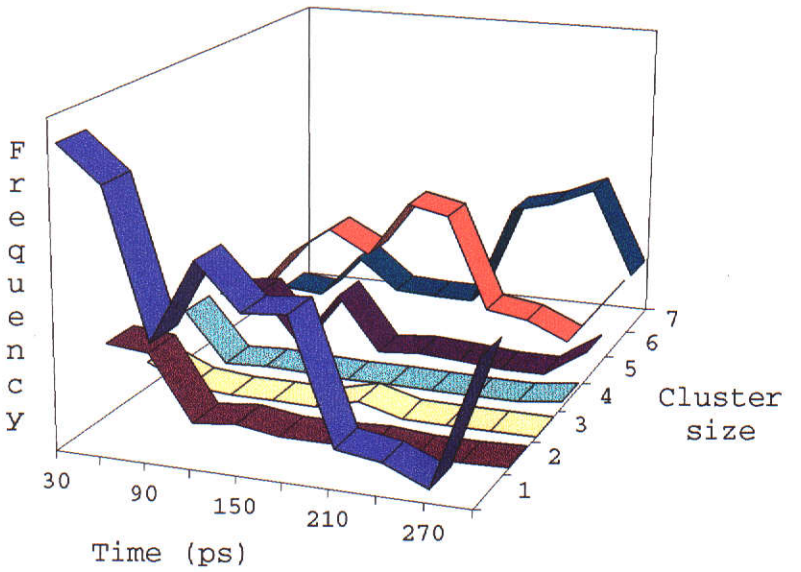

Figure 6.12: Frequency of occurrence of aluminate clusters of given size, throughout simulation 2.

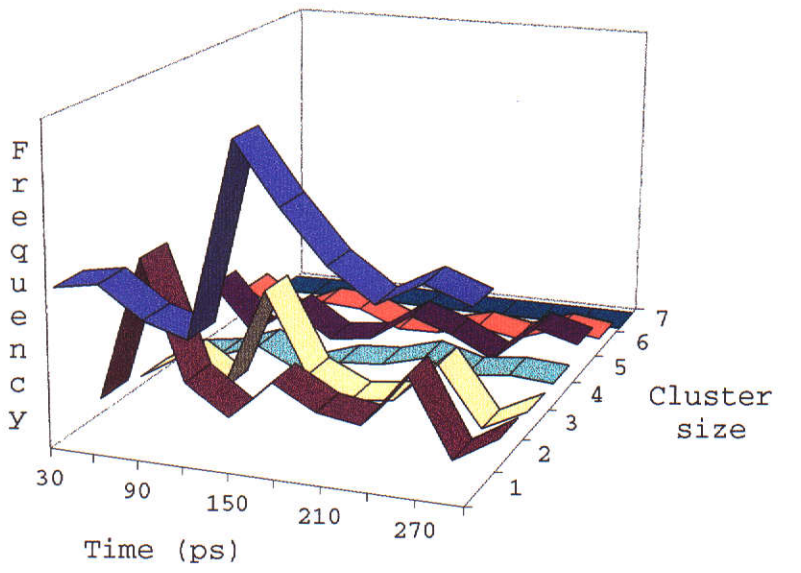

Figure 6.13: Frequency of occurrence for aluminate clusterings throughout simulation 3.

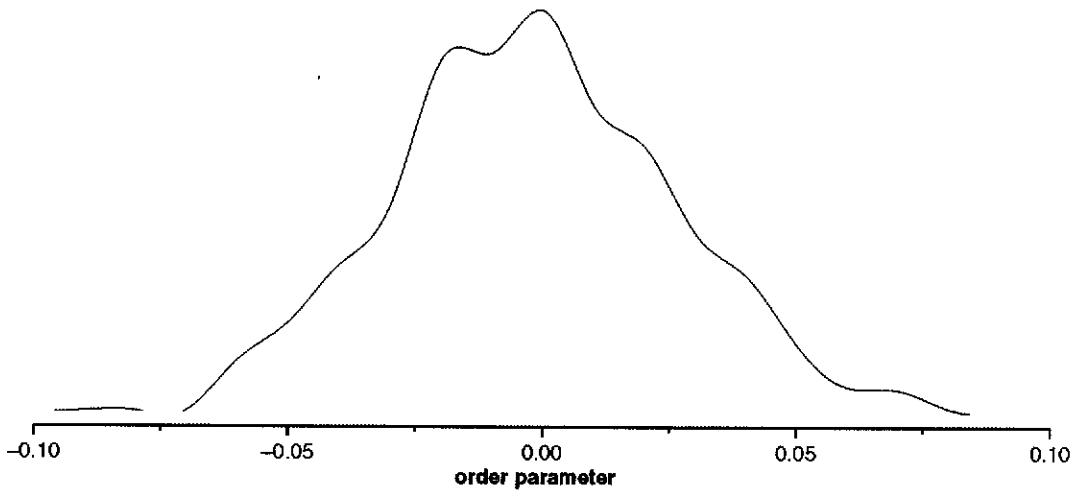

Figure 6.14: Order parameter distribution during the data collection period of the potassium aluminate solution.

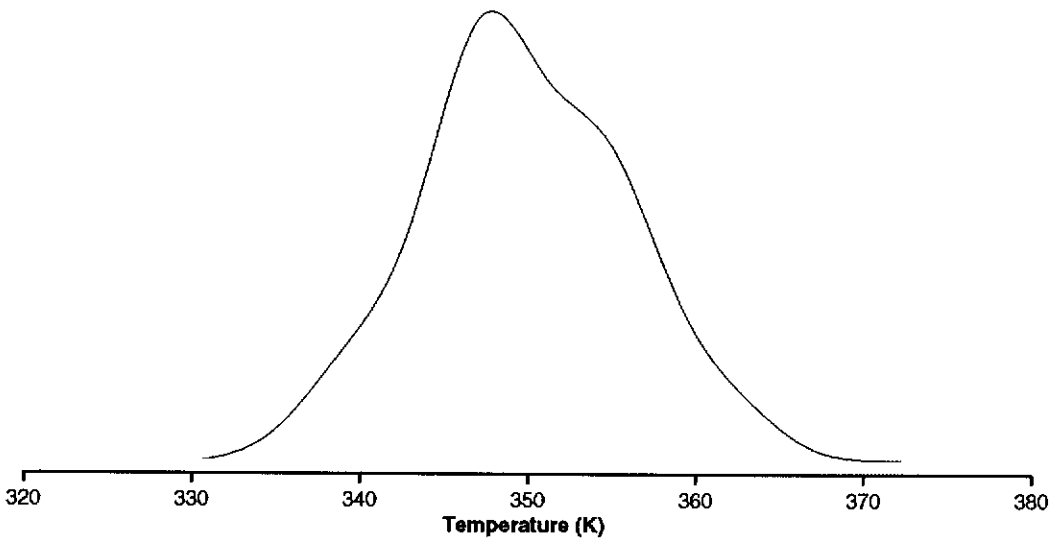

Figure 6.15: Potassium aluminate temperature distribution, over the data collection period.

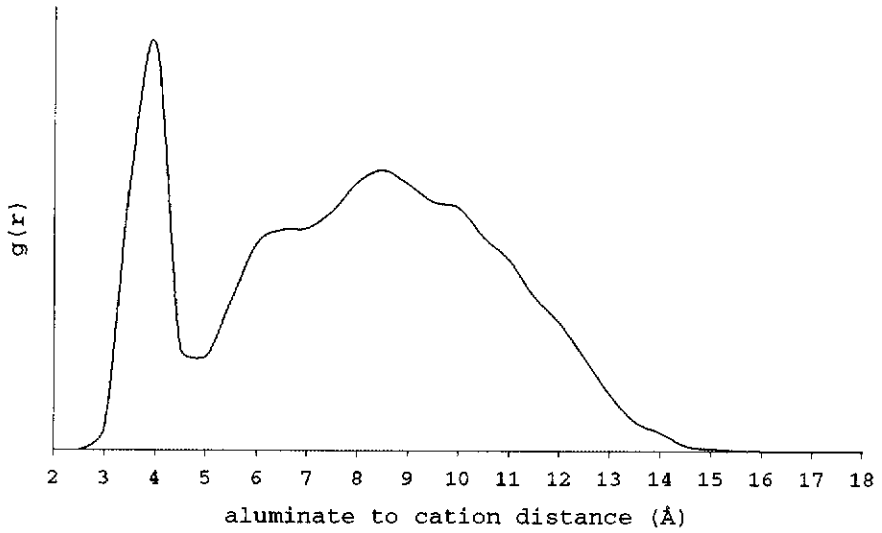

Figure 6.16: Potassium to aluminate radial distribution function.

mean ion pairing distance is greater at 3.9Å (compared to 3.5Å in sodium aluminate). This change must be related to the difference in ionic radius of sodium (0.97Å) compared to the larger potassium (1.33Å). In similar fashion, the aluminate to aluminate radial distribution function (Figure 6.17) exhibits the same general features as the equivalent RDF of the sodium aluminate solution. The mean separation of immediately adjacent clusters is larger, which is consistent with the larger ionic radius of potassium.

The occurrence of ion pairing is frequently invoked in order to explain anomalous reductions in the conductivity of an electrolyte. This is most evident for ion pairs that are charge neutral, and hence make no contribution to the conductivity. Thus, expressions for ion pair separation are centered around the electrolytic properties of the solution. In particular, Bjerrum in 1926 developed an expression for ion association based on the charges of the ions and the dielectric constant of the medium in which they interact [245]. However, the difference in the dielectric constant between sodium and potassium aluminate solutions is unlikely to be sufficient to completely account for the 0.4Å difference in ion pair separation. To develop a suitable explanation, clearly the dimensions of the cationic species involved should be considered. This approach is justified, as ion pairing is known to be dependent upon cation radii [175].

Thus, in order to predict an approximate ion pairing distance, a comparison

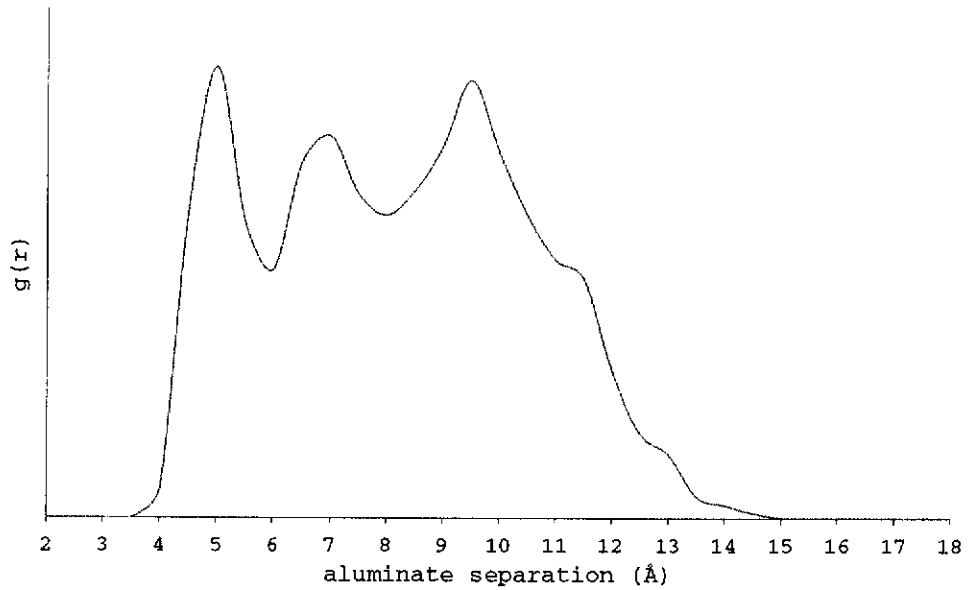

Figure 6.17: Aluminate to aluminate radial distribution function.

of the radii of the respective constituents was performed. This is trivial for the cationic species, but less so for the aluminate monomer. The molecular volume of an $Al(OH)_4$ molecule was calculated by rolling a probe ion over the van der Waals surface. This procedure was performed using the Cerius2 program from MSI [192]. For the probe atoms Na^+ and K^+, the volume was found to be 74.1Å^3 and 76.0Å^3 respectively. Although $Al(OH)_4^-$ is not a spherical molecule, only the average ion pairing distance is sought. Thus, the aluminate monomer was assumed to be a sphere for the purpose of obtaining an average molecular radius. The radii were found to be 2.61Å and 2.63Å for Na^+ and K^+ probe ions respectively. Adding to these the ionic radii of the probe gives distances of 3.58Å and 3.96Å. These compare very favourably with the values obtained from the simulations, and illustrates that the ion pairs are in close contact.

6.5 Discussion

The reaction for the dissolution of bauxite or the precipitation of gibbsite from hot alkaline aluminate solutions in the Bayer Process is often represented as,

$$Al(OH)_{4(aq)}^- \overset{NaOH}{\rightleftharpoons} Al(OH)_{3(s)} + OH_{(aq)}^-$$

but this expression fails to indicate the complexity of even the simplest case where gibbsite nuclei $(Al(OH)_3)_n$ form spontaneously in a supersaturated aluminate solution. Homogeneous gibbsite nucleation must involve processes such as:

- ordering of aluminate ions in solution,
- formation of hydroxo-bridges between Al atoms,
- a change from tetrahedral to octahedral Al coordination geometry,
- the release of sodium and hydroxyl ions from the nucleating crystal into the bulk solution,
- and the separation of solid phase from solution (dehydration).

Even in the Bayer process, where solutions are seeded and growth consists mainly of the incremental addition of gibbsite to seed crystal surfaces (during agglomeration, growth and secondary nucleation), aluminate ions must undergo changes similar to those listed above. Thus, any mechanism that is proposed for the precipitation of gibbsite must account for these processes, while being consistent with experimental observations. The data collected in the present study contribute mainly to an understanding of solution ordering.

A significant degree of ion pairing was noted in both sodium and potassium aluminate simulations. Experimental evidence lends some support to these findings. Raman spectra of alkali metal hydroxides have been found to exhibit a very broad low-intensity, low-frequency band at $\sim 300\ cm^{-1}$. This has been attributed to $MOH.H_2O$, in which an almost symmetric $O-M-O$ linkage exists [246]. The intensity of this band is reduced when aluminium is introduced into solution, a feature which has been interpreted as indicating that cation affinity is greater for the aluminate ion than for the hydroxide ion [247].

The data used to generate Figure 6.8 was examined in order to establish how the number of paired cations changes as a function of time. This analysis indicated that, in general, the cations associated with an aluminate are not rigidly bound. Thus, throughout the entire simulation, there is a random fluctuation in the number of cations paired with a particular aluminate. The mean value of these oscillations in associated cations was reported earlier as being between two and three. This mobility of the bridging cations is an important feature, since one of the intermediate steps in gibbsite precipitation involves the release of sodium ions to the bulk solution.

Interpretation of simulation data suggests that a similar trend in behaviour is occurring for both sodium and potassium solutions. The most notable difference is the weaker pairing between potassium and aluminate. This is supported by experimental findings. The results from analysis of spectra has showed the effects of the different cations on aluminate vibrational bands to be small [61]. Apart from a systematic decrease in the 620 cm^{-1} band intensity accompanied by a small increase in width at half height in the order Na>K, the spectra are remarkably similar across the full frequency range. Thus, comparisons between spectra of sodium and potassium aluminate solutions indicate that, if these spectral differences are due to cation-aluminate ion pairing, then pairing is stronger in the sodium system [61].

The data collected in the molecular dynamics simulations may be thought to indicate that the cationic and aluminate species evolve, from initial isolation, into subsequently larger and larger groups. These groups being characterized by ion pairing and aluminate clustering. This implies that, if given sufficient time, the simulations would exhibit the form seen in Figure 6.18, where all seven aluminates are bound into a maximal cluster. However, this will not necessarily occur as clusters may disperse again after formation. Simulations 1 and 3 both experience a certain degree of larger cluster sizes that do not persist. In particular, simulation 3, briefly forms size 6 clusters at various stages throughout the run (Figure 6.13). These clusters disperse relatively quickly, and do not form the larger maximal clustering seen in simulation 2. Thus, it is unlikely that a gradual acquisition of aluminates will in general continue until a maximal cluster is formed. The key factor in this regard is the number of sodium cations each aluminate ion is paired with. Figure 6.8 shows that simulations 1 and 3 experience an average of approximately two cation pairs per aluminate ion. Contrasting this with simulation 2, where the average number of paired cations is higher, it is evident that larger cluster sizes can be more readily sustained.

It has been assumed that monomeric $Al(OH)_4^-$ ions would be distributed randomly (homogeneously) in solution. However, the simulations predict that they aggregate into clusters which are stabilized by the cationic species. Unfortunately, the limitation of the simulation method used is that bonds cannot be formed or broken. Hence, although the aluminate monomers are predicted to form clusters which facilitate the formation of polyaluminate species, no information about either the bonding between aluminate ions or the aluminium geometry is obtained from these simulations. In addition, the stabilizing cations paired to the aluminates have been demonstrated to be quite mobile. This is certainly expected,

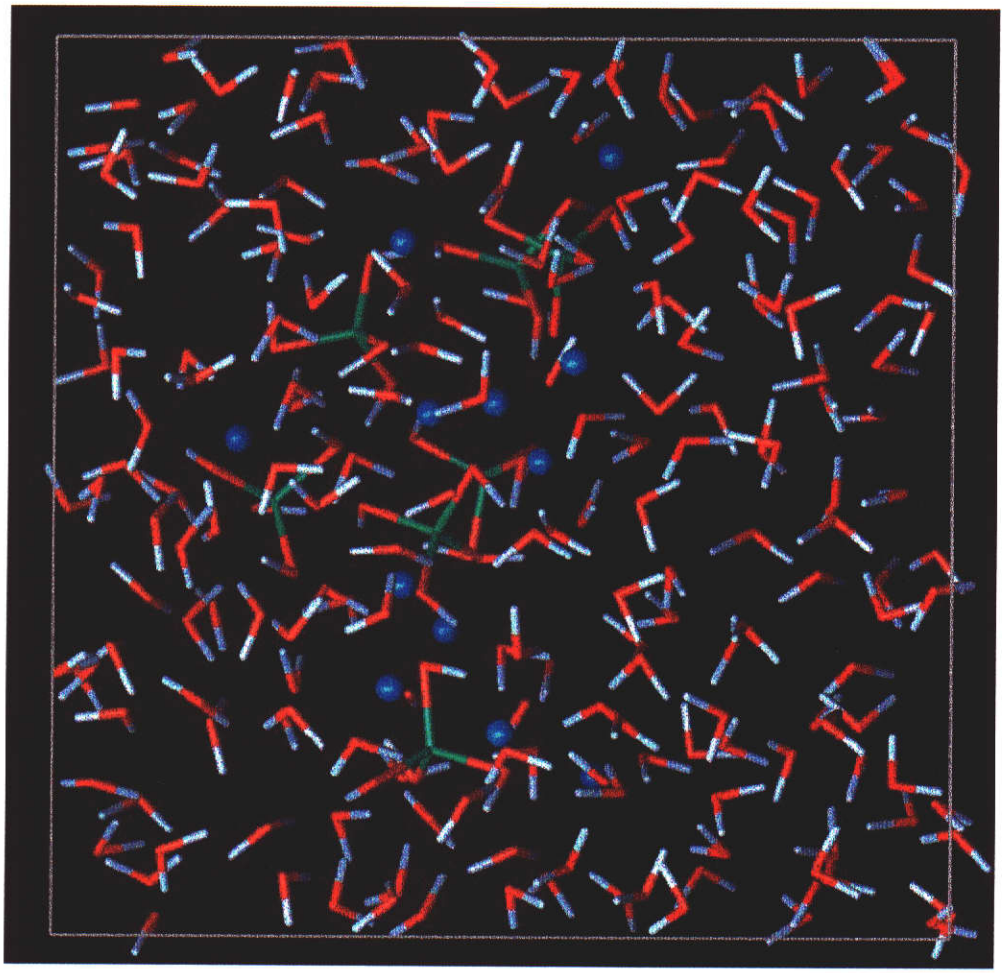

Figure 6.18: Snapshot of aluminates forming a maximal cluster size.

given that ejection of sodium is an important part of the gibbsite precipitation step.

Aluminate clustering has some corroboration in the literature. It has been noted that spectra collected during the induction period, up to the point where particles were visible as a faint suspension, are characterized by an increased intensity of Rayleigh scattering [61]. The estimated diameter of the particles was approximately one sixth of the wavelength of the laser used (1064 nm). Recent additional data from multiangle laser light scattering [248] also showed a progressive increase in light scattering prior to the appearance of suspended particles, consistent with a nucleation mechanism. In that case, particles were estimated to be $\approx 180nm$ in diameter, but according to Rossiter *et al.* [67] this might be an overestimation. The important point is that such particles could not form if the aluminate ion clustering, as predicted by the simulation, did not take place. In addition, certain bands observed in the Raman and infrared spectra of concentrated aluminate solutions have been interpreted as arising from the vibrations of aluminate clusters (and any polyaluminate anions which form from them) [61]. The absence of spectral features characteristic of a distinct interfacial aluminate layer at the growing crystal surface could be explained by the predicted clustering throughout the bulk solution, and the participation of such clusters in the formation of polyanionic growth units.

Conclusions and Future Work

This thesis has documented the development of a potential model for gibbsite that has successfully reproduced a range of experimental data. The model was obtained through fitting multiple hydroxide structures to ensure robustness and transferability, with an average error in lattice parameters of under two percent. Although a partial charge scheme was applied within the hydroxyl group, the overall charge was constrained to -1 in order to facilitate transferability. Hence, the model should be suitable for modelling other hydroxide materials of interest.

The importance of sodium oxalate to the Bayer process led to an attempt to construct a transferable oxalate potential model. Firstly, an intramolecular potential that described the oxalate dianion was sought. However, due to the instability of the dianion under vacuum conditions, a point charge array that approximately mimics the electrostatic field of a surrounding lattice was utilized. This had the desired effect of stabilizing the HOMO and allowing a suitable energy hypersurface to be constructed. The second stage of fitting involved developing intermolecular terms in order to reproduce the sodium, potassium and calcium oxalate monohydrate structures. This was accomplished by holding the previously determined intramolecular potentials fixed, while fitting all the other terms.

The gibbsite potential model was successfully employed in an equilibrium morphology prediction, with all experimentally observed faces reproduced. The predicted habit is a chamfered hexagonal prism, which matches well with experimentally grown crystals despite an underestimation of the prismatic faces. It was also noted that all the surface aluminium atoms for each gibbsite face were five coordinate. The one exception was the basal plane, with all the aluminium atoms being six coordinate. One of the key issues in gibbsite crystallization is the identification of the growth unit(s). The development of a good potential model, in addition to the studies on gibbsite surfaces, is an important contribution towards this end. Future effort is needed to examine the docking of aluminate species (that are known to be present in solution) onto gibbsite surfaces. Although the empirical model developed in this work could be used to describe interactions

within gibbsite, a hybrid QM/MM method would be needed to deal with possible aluminium coordination changes in the postulated growth unit(s).

Gibbsite is typically grown in alkali aluminate solutions, for which incorporation of the metallic cation is known to occur. This is thought to occur *via* proton replacement. Modelling the change in surface energy, caused by impurity substitution, allowed 'defect' morphologies to be predicted. These were found to be highly dependent upon the relative surface coverage of separate faces. However, with the assumption of near uniform coverage for all faces, defect incorporation was found to have an elongating effect on the morphology. Consideration of how coverage might vary in order to minimize the defect energy per unit surface area yielded a possible explanation for the formation of diamond gibbsite morphologies. Further information concerning defect morphologies may be ascertained by applying a growth model, as opposed to the equilibrium model described in this work. In particular, the calculation of the attachment energy of a growth slice onto a defect surface should be attempted. As a result, information concerning how such defects change the relative growth rates of the faces in gibbsite would be determined.

In the defect studies, the arrangements of atoms on the surface of gibbsite were complex and it was not trivial to ascertain which protons should be considered surface sites and therefore candidates for replacement. Hence, a scheme that computed each hydrogen's accessibility to a given probe atom was used. The SCURF code was written to accomplish this, and is suitable for application to any system where some means to distinguish surface sites from bulk sites is needed. In addition, this work employed a 2x2 (manual) surface construction to avoid artificially including edge sites as surface sites. A better method would be to automatically construct periodic images surrounding the slab of material under examination. This will be implemented in future revisions of the SCURF code.

Using attachment energy calculations of pure and reflected growth slices onto crystal surfaces, a scheme for predicting crystalline twinning was developed. The GTWIN code examines the difference in attachment energy for a normal growth slice and its mirror image (with the plane of reflection parallel to the crystal surface). Gibbsite was predicted to form reflection twins parallel to the (002) and (110) planes. This result is confirmed in the literature from mineralogy references and also from experimentally grown crystals in synthetic Bayer liquors. The use of this scheme to successfully predict twinning in several additional crystal systems suggests that it can be used generally as a means to predict reflection twins. However, an obvious improvement to the GTWIN code would be the inclusion of

additional symmetry operations so that other forms of twinning may be examined.

Complementing the solid state work, simulations of synthetic Bayer liquor were also conducted which yielded two important pieces of information. Firstly, a significant degree of ion pairing between the aluminate monomer and the cationic species was observed to occur. Secondly, the grouping of aluminates into cation mediated clusters was also noted. These findings helped propose an explanation for observations made using IR and Raman spectroscopy studies. The significant degree of binding between the cationic species and the aluminate monomer (which is the dominant solution species) also provides a basis for explaining cationic incorporation in gibbsite crystals. Further work, employing a computational methodology which allows for the formation and breaking of bonds is needed to probe this further. Additional studies, probing the effect of different concentrations should also provide valuable insight.

Bibliography

[1] H.M. Rosenberg. *The Solid State.* 1990.

[2] J.P. Glusker and K.N. Trueblood. *Crystal Structure Analysis.* Oxford University Press, Oxford, 1985.

[3] B.L. Silver. *The Ascent of Science.* Oxford University Press, Oxford, 1998.

[4] M.J. Buerger. *Introduction to crystal geometry.* McGraw-Hill, New York, 1971.

[5] D. Halliday and R. Resnick. *Fundamentals of Physics.* John Wiley and Sons, New York, 1988.

[6] G. Coddens and P. Launois, *Endeavour* **20** (1996) 16.

[7] L. Smart and E. Moore. *Solid State Chemistry: An introduction.* Chapman and Hall, London, 1995.

[8] M.J. Buerger. *Elementary Crystallography.* John Wiley and Sons, New York, 1963.

[9] P. Kramer and A. Neri, *Acta. Crystallogr.* **A 40** (1984) 590.

[10] D. Shechtman, I. Blech, D. Gratias and J. Cahn, *J. Phys. Rev. Lett.* **53** (1984) 1951.

[11] R. Penrose, *Bull. Inst. Math. Appl.* **10** (1974) 266.

[12] Pauling, *Nature (London)* **317** (1985) 512.

[13] Pauling, *Phys. Rev. Lett.* **58** (1987) 365.

[14] C. Janot. *Quasicrystals.* Clarendon Press, Oxford, 1994.

[15] T. Hahn, editor. *International tables for crystallography : brief teaching edition of volume A, space-group symmetry*, Dordrecht, Holland, 1985. International Union of Crystallography, D. Reidel.

[16] E. Prince. *Mathematical Techniques in Crystallography and Materials Science.* Springer-Verlag, New York, 1994.

[17] J. Feder. *Fractals.* Plenum Press, New York, 1988.

[18] N. Steno. *De solido intra solidum naturaliter contento dissertationis prodromus.* Florence, 1669.

[19] J.W. Mullin. *Crystallization.* Butterworth, London, 1997.

[20] F.G. Friedel. *Lecon de Cristallographie.* Hermann, Paris, 1911.

[21] J. Donnay and D. Harker, *Am. Mineral.* **22** (1937) 446.

[22] J. Donnay and G. Donnay, *C.R. Acad. Sci. Paris* **252** (1961) 908.

[23] M. Buerger, *Am. Mineral.* **32** (1947) 593.

[24] P. Hartman and W.G. Perdok **8** (1955) 49-52.

[25] P. Hartman and W. Perdok, *Acta Cryst.* **8** (1955) 521.

[26] P. Hartman and W. Perdock, *Acta Cryst.* **8** (1955) 525.

[27] P. Hartman and P. Bennema, *J. Crystal Growth* **49** (1980) 145.

[28] J.W. Gibbs. *Collected Works.* Longman, 1928.

[29] A.R. West. *Basic Solid State Chemistry.* John Wiley and Sons, Chichester, 1991.

[30] O. Sohnel and J. Garside. *Precipitation: Basic Principles and Industrial Applications.* Butterworth-Heinemann, Oxford, 1992.

[31] R.F. Strickland-Constable. *Kinetics and Mechanism of Crystallization.* Academic Press, London, 1968.

[32] G.M. Van Rosmalen and A.E. Van Der Heijden. Secondary nucleation. In J.P. van der Eerden and O.S.L. Bruinsma, editors, *Science and Technology of Crystal Growth* 259, Dordrecht, Netherlands, 1995. Kluwer Academic Publishers.

[33] W. Kossel, *Annalen. der Physik* **21** (1934) 457.

[34] E. Huitema. *Crystal growth and defect generation.* PrintPartners Ipskamp, Enschede, 1998.

[35] A.A. Chernov. In *Physico-Chemical Problems of Crystallization*, volume 1. Kasakh State University, 1969.

[36] R.A. Laudise, J.R. Carruthers and K.A. Jackson, *Ann. Rev. Mater. Sci.* **1** (1971) 253.

[37] K.A. Jackson. In *All-Union Conference on Crystal Growth*, Additional volume 7. A.S. Arm. SSR, Erevan, 1972.

[38] V.V. Solov'ev and V.T. Borisov, *Kristallografiya* **5** (1972) 918.

[39] W.K. Burton, N. Cabrera and F.C. Frank, *Trans. Roy. Soc.* **A 243** (1951) 299.

[40] J. Garside. Mass crystallization, number balances and size distributions. In J.P. van der Eerden and O.S.L. Bruinsma, editors, *Science and Technology of Crystal Growth* 209, Dordrecht, 1995. Kluwer Academic Publishers.

[41] A.D. Randolph and M.A. Larson. *Theory of Particulate Processes.* Academic Press, New York, 1988.

[42] T. Manth, D. Mignon and H. Offermann, *J. Cryst. Growth* **166** (1996) 998.

[43] B. Farkas, T. Blickle, Zs. Ulbert and M. Hasznos-Nezdei, *J. Cryst. Growth* **166** (1996) 1064.

[44] R.I. Ristic, J.N. Sheerwood and T. Shripathi. In *Advances in Industrial Crystallization* 77, Oxford, 1991. Butterworth-Heinemann.

[45] http://www.mineralswa.asn.au.

[46] Von R. Rothbauer, F. Zigan and H. O'Daniel, *Z. Kristallogr.* **125** (1967) 317.

[47] F. Zigan, W. Joswig and N. Burger, *Z. Kristallogr.* **145** (1979) 1977.

[48] H.J. Bosmans, *Acta Cryst. B* **24** (1982) 1968.

[49] G.R. Clark, K.A. Rodgers and G.S. Henderson, *Z. Kristallogr.* **213** (1998) 96.

[50] H. Saalfeld and M. Wedde, *Z. Kristallogr.* **139** (1974) 129.

[51] R.F. Giese Jr., *Acta Cryst.* **B32** (1976) 1719.

[52] J.D. Gale. Private communications.

[53] M.C. Payne, M.P. Teter, D.C. Allan, T.A. Arias and J.D. Joannopoulos, *Rev. Mod. Phys.* **64** (1992) 1045.

[54] J.P. Perdew. *Electronic Structure of Solids.* Akademie Verlag, Berlin, 1991.

[55] K. Wefers and C. Misra, *Alcoa Technical Paper No. 19, Revised* (1987) 8.

[56] G. Verdes, R. Gout and S. Castet, *Eur. J. Mineral.* **4** (1992) 767.

[57] W.A. Deer, R.A. Howie and J. Zussman. *An Introduction to The Rock Forming Minerals.* Longman, London, 1980.

[58] W.L. Roberts, T.J. Campbell and G.R. Rapp. *Encyclopedia of Minerals.* Van Nostrand Reinhold, New York, 1990.

[59] R.J. Moolenaar, J.C. Evans and L.D. McKeever, *J. Phys. Chem.* **74** (1970) 3629.

[60] N.-Y. Chen, M.-X. Lui, T.-L. Cao, B. Tang and M. Hong, *Science in China* **B36** (1993) 32.

[61] H.R. Watling, S.D. Fleming, W. van Bronswijk and A.L. Rohl, *J. Chem. Soc. Dalton Trans.* (1998) 3911.

[62] J.W. Akitt and W. Gessner, *J. Chem. Soc. Dalton Trans.* (1984) 147.

[63] J.W. Akitt, W. Gessner and M. Weinberger, *Magn. Res. Chem.* **26** (1988) 1047.

[64] E.A. Kopylova, J.P. No and M.V. Zakharova, *J. App. Chem. (USSR)* **47** (1974) 2396.

[65] L.A. Myund, V.M. Sizyakov, M.K. Khripun and A.A. Makarov, *Russian J. General Chem.* **65** (1995) 826.

[66] L.A. Myund, V.M. Sizyakov, K.A. Burkov, V.O. Zakharzhevskaya and O.A. Borzen, *Russian J. General Chem.* **68** (1995) 1721.

[67] D.S. Rossiter, P.D. Fawell, D. Ilievski and G.M. Parkinson, *J. Cryst. Growth* **191** (1998) 525.

[68] H.V. Olphen and J.J. Fripiat. *Data Handbook for Clay Materials and other Non-Metallic Materials.* Pergamon Press, Oxford, 1979.

[69] S.M. Bradley, R.A. Kydd and R.F. Howe, *J. Colloid Interface Sci.* **159** (1993) 405.

[70] H. Lui, S. Huang and N. Chen, *Trans. Nonferrous Met. Soc. China* **2** (1992) 43.

[71] N. Chen and H. Lui, *J. Mol. Struct (Theochem.)* **305** (1994) 283.

[72] P. Sipos, P.M. May, G.T. Hefter and I. Kron, *J. Chem. Soc. Chem. Comm.* (1993) 2355.

[73] L. Honglin, H. Shiping and C. Nianyi, *Trans. Non Ferrous Soc. (China)* **2** (1992) 43.

[74] A.R. Gerson, J.A. Counter and D.J. Cookson, *J. Cryst. Growth* **160** (1996) 346.

[75] J.D. Gale, A.L. Rohl, H.R. Watling and G.M. Parkinson, *J. Phys. Chem. B* **102** (1998) 10372.

[76] E.R. Lippincott, J.A. Psellos and M.C. Tobin, *J. Chem. Phys.* **20** (1952) 536.

[77] A.C. Hess, P.F. McMillan and M. O'Keefe, *J. Phys. Chem.* **92** (1998) 1785.

[78] Agnes Buvari-Bareza, Marta Rozsahegyi and Laos-Bareza, *J. Mater. Chem.* **8** (1998) 451-455.

[79] G.M. Parkinson. Crystallisation - from mountains to minerals. In *Fourth Alumina Quality Workshop* 271, Darwin, NT, 1996.

[80] C. Misra and E.T. White, *Chem. Eng. Prog. Symp. Ser.* **67** (1971) 53,110.

[81] C. Misra and E.T. White, *J. Cryst. Growth* **8** (1971) 172.

[82] W.R. King, *Light Metals* (1973) 551.

[83] S. Veesler and R. Boistelle, *J. Cryst. Growth* **142** (1994) 177.

[84] N. Brown, *J. Cryst. Growth* **12** (1972) 39.

[85] T.G. Pearson, in. *The chemical background of the aluminium industry.* Number 3 in Lectures, Monographs and Reports. The Royal Institute of Chemistry, 1955.

[86] P.R. Bloom and R.M. Weaver, *Clays and Clay Min.* **50** (1982) 281.

[87] H.A. Straten. *Formation of Solid phases from Supersaturated Aluminate Solutions.* PhD thesis, University of Utrecht, Holland, 1984.

[88] A. Violante and P.M. Huang, *Clays and Clay Minerals* **41** (1993) 590.

[89] A.G. Nesterov and V.G. Teslya, *Zhurnal Priklodnoi Khimii* **62** (1989) 1999.

[90] K. Wefers and G.M. Bell. *Oxides and hydroxides of aluminium: Technical paper No. 19.* Alcoa Research Laboratories, 1972.

[91] J. Addai-Mensah, *Minerals Engineering* **10** (1997) 81.

[92] J. Zambo, *Light Metals* (1986) 199.

[93] S.C. Grocott and S.P. Rosenberg. Soda in alumina: Possible mechanisms for soda incorporation, smelter grade alumina for the 1990s and beyond. In *First International Alumina Quality Workshop* 271, Gladstone, QLD, Australia, 1988.

[94] M.V. Chaubal, *Light Metals* (1990) 85.

[95] N. Brown, *J. Cryst. Growth* **16** (1972) 163.

[96] S. Freij, M.-Y. Lee, M.M. Reyhani and G.M. Parkinson. Investigation of the growth of gibbsite crystals by atomic force microscopy and optical microscopy. In *Proc. Fifth Int. Alumina Quality Workshop* 41, Bunbury, WA, 1999.

[97] D.H. Rouvray, *Chemistry and Industry* (1997) 587.

[98] K.S. Krane. *Introductory Nuclear Physics.* John Wiley and Sons, Singapore, 1988.

[99] I. Newton. *Principia Mathematica.* 1687.

[100] I.M. Torrens. *Interatomic Potentials.* Academic Press, New York, 1972.

[101] G.C. Maitland, M. Rigby, E.B. Smith and W.A. Wakeham. *Intermolecular Forces: Their Origin and Determination.* Oxford University Press, Oxford, 1981.

[102] I.G. Kaplan. *Theory of Molecular Interactions.* Elsevier, Amsterdam, 1986.

[103] C.T. Walker and G.A. Slack, *Am. J. Phys.* **38** (1970) 1380.

[104] H. Rechenberg. *Twentieth century physics,* volume 2 143. Institute of Physics and New York American Institute of Physics, Bristol, 1995.

[105] E. Merzbacher. *Quantum Mechanics.* John Wiley and Sons, Singapore, 1970.

[106] C.R.A. Catlow, R. G. Bell and J.D. Gale, *J. Mater. Chem.* **4** (1994) 781.

[107] M.J. Field, P.A. Bash and M. Karplus, *J. Comp Chem.* **11** (1990) 700.

[108] V. Thery, D. Rinaldi, J.-L. Rivail, B. Maigret and G. Ferenczy, *J. Comp. Chem.* **15** (1994) 269.

[109] R.V. Stanton, D.S. Hartsough and K.M. Merz, Jr., *J. Comp. Chem.* **16** (1995) 113.

[110] D. Bakowies and W. Thiel, *J. Comp. Chem* **17** (1996) 87.

[111] C.R.A. Catlow and G.D. Price, *Nature* **347** (1990) 243.

[112] C.M. Freeman and C.R.A. Catlow, *J. Chem. Soc., Chem. Commun.* (1992) 89.

[113] C.R.A. Catlow, L. Ackermann, R.G. Bell, F. Cora, D.H. Gay, M.A. Nygren, J.C. Pereira, G. Sastre, B. Slater and P.E. Sinclair, *Faraday Discuss.* **106** (1997) 1.

[114] A.R. Leach. *Molecular Modelling.* Addison Wesley Longman, Singapore, 1996.

[115] M.P. Allen and D.J. Tildesley. *Computer Simulation of Liquids.* Clarendon Press, Oxford, 1997.

[116] F. Figueirido, R.M. Levy, R. Zhou and B.J. Berne, *J. Chem. Phys.* **106** (1997) 9835.

[117] P.P. Ewald, *Ann. Phys. (Leipzig)* **64** (1921) 1.

[118] D.E. Parry, *Surf. Sci.* **49** (1975) 433.

[119] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller, *J. Chem. Phys.* **21** (1953) 1087.

[120] B.J. Alder and T.E. Wainwright, *J. Chem. Phys.* **27** (1957) 1208.

[121] B.J. Alder and T.E. Wainwright, *J. Chem. Phys.* **31** (1959) 459.

[122] B.J. Alder and T.E. Wainwright, *J. Chem. Phys.* **33** (1960) 1439.

[123] P.R. Wolde and M.J. Ruiz-Montero and D. Frenkel, *J. Chem. Phys.* **104** (1996) 9932.

[124] Y.C. Shen and D.W. Oxtoby, *J. Chem. Phys.* **104** (1995) 4233.

[125] I.M. Svishchev and P.G. Kusalik, *Phys. Rev. Letts* **73** (1994) 975.

[126] J. Anwar and P.K. Boateng, *J. Am. Chem. Soc.* **120** (1998) 9600.

[127] G.V. Lewis and C.R.A. Catlow, *J. Phys. C* **18** (1985) 1149.

[128] A. Pavese, M. Catti, G.D. Price and R.A. Jackson, *Phys. Chem. Minerals* **19** (1992) 80.

[129] N.L. Allan, A.L. Rohl, D.H. Gay, C.R.A Catlow, R.J. Davey and W.C. Mackrodt, *Faraday Discuss.* **95** (1993) 273.

[130] J.D. Gale and N.J. Henson, *J. Chem. Soc. Faraday Trans.* **90** (1994) 3175.

[131] A. Pavese, M. Catti, S.C. Parker and A. Wall, *Phys. Chem. Minerals* **23** (1996) 89-93.

[132] M. Born and K. Huang. *Dynamical theory of crystal lattices.* Clarendon Press, Oxford, 1954.

[133] S.C. Parker and G.D. Price, *Adv. Solid State Chem.* **1** (1989) 295.

[134] T.S. Bush, J.D. Gale, R.A. Catlow and P.D. Battle, *J. Mater. Chem.* **4** (1994) 831-837.

[135] A.M. Gorman, C.M. Freeman, C.M. Kolmel and J.M. Newsam, *Faraday Discuss.* **106** (1997) 489.

[136] Erik de Vos Burchart. Studies on zeolites: molecular mechanics, framework stability, and crystal growth. Technical report, Technical University of Delft, 1992.

[137] B. J. Teppen, K. Rasmussen, P.M. Bertsch, D.M. Miller and L. Schafer, *J. Phys. Chem. B* **101** (1997) 1579.

[138] K.S. Smirnov and B. van de Graaf, *J. Chem. Soc., Faraday Trans.* **92** (1996) 2475.

[139] General Discussion, *Faraday Discuss.* **95** (1993) 367.

[140] W.C. Mackrodt., R.J. Davey, S.N. Black and R. Doherty, *J. Cryst. Growth* **80** (1987) 441.

[141] D.H. Gay and A.L. Rohl, *J. Chem. Soc. Faraday Trans.* **91** (1995) 925.

[142] C.S. Strom, R.F.P. Grimbergen, I.D.K. Hiralal, B.G. Koenders and P. Pennema, *J. Cryst. Growth* **149** (1995) 107.

[143] Arunan Nadarajah and Marc L. Pusey, *Acta Cryst. D* **52** (1996) 983.

[144] H. Iwasaki and F. Iwasaki, *J. Cryst. Growth* **151** (1995) 348.

[145] D. Aquilano, M. Rubbo, M. Catti and A. Pavese, *J. Cryst. Growth* **182** (1997) 168.

[146] N.H. Leeuw and S.C. Parker, *J. Phys. Chem. B* **102** (1998) 2914.

[147] R.A. Jackson and G.D. Price, *Mol. Sim.* **9** (1992) 175-177.

[148] J.O. Titiloye, S.C. Parker, D.J. Osguthorpe and S. Mann, *J. Chem. Soc. Chem. Comm.* (1991) 1494.

[149] P.R. Kenway, P.M. Oliver, S.C. Parker, D.C. Sayle, T.X.T. Sayle and J.O. Titiloye, *Mol. Sim.* **9** (1992) 83.

[150] J.O. Titiloye, S.C. Parker and S. Mann, *J. Cryst. Growth* **131** (1993) 533.

[151] W.J. Benton, I.R. Collins, I.M. Grimsey, G.M. Parkinson and S.A. Rodger, *Faraday Discuss.* **95** (1993) 281.

[152] M-Y. Lee, A.L. Rohl, J.D. Gale, G.M. Parkinson and F.J. Lincoln, *Trans. IChemE* (1996) 739.

[153] F.J. Vesely. *Computational Physics, An Introduction.* Plenum Press, New York, 1994.

[154] G.K. Woodgate. *Elementary Atomic Structure.* Clarendon Press, Hong Kong, 1983.

[155] A. Hinchliffe. *Modelling Molecular Structures.* John Wiley and Sons, Chichester, 1996.

[156] D.V. George. *Principles of Quantum Chemistry.* Pergamon Press, New York, 1972.

[157] P. Hohenberg and W. Kohn, *Phys. Rev. B* **136** (1964) 864.

[158] R.G. Parr and W. Yang. *Density-Functional Theory of Atoms and Molecules.* Oxford University Press, New York, 1989.

[159] Molecular Simulations, San Diego. *DMOL User Guide*, 1996.

[160] M.J.S. Dewar and W. Thiel, *J. Am. Chem. Soc.* **99** (1977) 4899.

[161] M.J.S. Dewar, E.G. Zoebisch, F.H. Eamonn and J.J.P. Stewart, *J. Am. Chem. Soc.* **107** (1985) 3902.

[162] J.J.P. Stewart, *J. Comp. Chem.* **10** (1989) 221.

[163] J.C. Slater, *Phys. Rev.* **36** (1930) 57.

[164] E.R. Davidson and D. Feller, *Chemical Reviews* **86** (1986) 681.

[165] D.J. DeFrees, B.A. Levi, J.S. Binkley and J.A. Pople, *J. Am. Chem. Soc.* **101** (1979) 4085.

[166] R. Krishnan, J.S. Binkley, R. Seeger and J.A. Pople, *J. Chem. Phys.* **72** (1980) 650.

[167] R.J. Abraham and B.J. hudson, *J. Comp. Chem.* **6** (1984) 173.

[168] P. Comba and T.W. Hambley. *Molecular Modeling of Inorganic Compounds.* VCH, Weinheim, 1995.

[169] S.J. Weiner, P.A. Kollmann, D.A. Case, C. Singh, C. Ghio and S. Proteta, *J. Am. Chem. Soc.* **106** (1984) 765.

[170] M. Whitlow and M.M. Teeter, *J Am Chem Soc* **108** (1986) 7163.

[171] A. Warshel, *J. Phys. Chem.* **83** (1979) 1640.

[172] D. Hall and N. Pavitt, *J. Comp. Chem.* **5** (1984) 441.

[173] B.G. Dick and A.W. Overhauser, *Phys. Rev.* **75** (1992) 379.

[174] J.D. Gale, *Phil. Mag. B* **73** (1996) 3.

[175] W.J. Moore. *Physical Chemistry.* Longman, London, 1976.

[176] J.E. Lennard-Jones, *Proc. Roy. Soc.* **A106** (1924) 463.

[177] J.E. Lennard-Jones, *Physica* **4** (1937) 941.

[178] R.A. Buckingham, *Proc. Roy. Soc.* **A168** (1938) 264.

[179] J.C. Slater, *Phys. Rev.* **32** (1928) 349.

[180] W.E. Bleick and J.E. Mayer, *J. Chem. Phys.* **2** (1934) 252.

[181] A.L. Rohl. *The Size and Shape of Molecular Ions and their Relevance to Packing.* PhD thesis, University of Oxford, 1991.

[182] R.L. Burden and J.D. Faires. *Numerical Analysis.* PWS, Boston, 1993.

[183] M.J. Norgett and R. Fletcher, *J. Phys. C.* **3** (1970) L190.

[184] A. Jentys, R.W. Grimes, J.D. Gale and C.R.A. Catlow, *J. Phys. Chem.* **97** (1993) 13535.

[185] A.O.E. Animalu. *Intermediate Quantum Theory of Crystalline Solids.* Prentice-Hall, New Jersey, 1977.

[186] C. Kittel. *Introduction to Solid State Physics.* John Wiley and Sons, New York, 1986.

[187] G. Wulff, *Z. Kristallogr. Miner.* **34** (1901) 449.

[188] F. Bertaut, *Compt. Rend.* **246** (1958) 3447.

[189] E.S. Oran and J.P. Boris. *Numerical Simulation of Reactive Flow.* Elsevier, New York, 1987.

[190] H. Goldstein. *Classical Mechanics.* Addison-Wesley, Singapore, 1980.

[191] University of Cambridge, Cambridge. *http://ket.ch.cam.ac.uk/software/cadpac.html.*

[192] Molecular Simulations Incorporated, San Diego. *http://www.msi.com.*

[193] *http://SAL.KachinaTech.COM.*

[194] J.D. Gale, *J. Chem. Soc. Faraday Trans.* **93** (1997) 629.

[195] L. Wall, T. Christiansen and R.L. Schwartz. *Programming Perl.* O'Reilly and Associates, Sebastopol, 1996.

[196] *http://www.redhat.com.*

[197] *http://www.povray.org.*

[198] *http://www.gimp.org.*

[199] *http://www.nirim.go.jp/ weber/JAVA/jshape/jshape.html.*

[200] P.S. Baram and S.C. Parker, *Phil. Mag. B* **73** (1996) 49.

[201] J.A. Gadsden. *Infrared Spectra of Minerals and Related Inorganic Compounds.* Butterworth, London, 1975.

[202] L.K. Hudson. *Alumina Production.* Aluminium Company of America, U.S.A., 1982.

[203] W.G. Robertson. Urinary tract calculi. In B.E.C. Nordin, editor, *Metabolic Bone and Stone Disease.* Churchill Livingstone, Edinburgh, 1984.

[204] J.D. DeLong and D. Briedis, *J. Cryst. Growth* **71** (1985) 689.

[205] V.A. Ermoskin, K.S. Smirnov and D. Bougeard, *Surface Science* **368** (1996) 147.

[206] J.-R. Hill and J. Sauer, *J. Phys. Chem.* **99** (1995) 9536.

[207] J.D. Gale, C.R.A. Catlow and W.C. Mackrodt, *Modelling and Simulation in Material Science Engineering* **1** (1992) 73.

[208] A.T. Hagler, R. Sharon and M.-J. Hwang, *J. Am. Chem. Soc.* **118** (1996) 3759.

[209] H. Sun, S.J. Mumby, J.R. Maple and A.T. Hagler, *J. Phys. Chem.* **99** (1995) 5873.

[210] J.R. Maple, H.-J. Hwang, T.P. Stockfisch, U. Dinur, M. Waldman, C.S. Ewig and A.T. Hagler, *J. Comp. Chem.* **15** (1994) 162.

[211] L. Radom. Molecular anions. In *Methods of Electronic Structure Theory* 333. Plenum Press, New York, 1977.

[212] P.J. Bruna, R.M. Mawwhinney and F. Grein, *J. Phys. B* **29** (1996) 2413.

[213] D.A. Reed and M.M. Olmstead, *Acta Cryst. B* **B37** (1981) 938.

[214] A. Sequeira, S. Srikanta and R. Chidambaram, *Acta Cryst.* **B26** (1970) 77.

[215] V. Tazzoli and C. Domeneghetti, *American Mineralogist* **65** (1980) 327.

[216] D.R. Lide (ed.). *CRC Handbook of Chemistry and Physics.* CRC Press, New York, 1996.

[217] C. Sweegers, W.J.P. van Enckevort, H. Meekes, P. Bennema, I.D.K. Hiralal and A. Rijkeboer, *J. Cryst. Growth* **197** (1999) 244.

[218] M.-Y. Lee. *The Mechanism of Gibbsite Crystal Growth in Bayer Liquor.* PhD thesis, Curtin University of Technology, 1998.

[219] D.C. Sayle, C.R.A. Catlow, M.A. Perrin and P. Nortier, *J. Phys. Chem. Solids* **56** (1995) 799.

[220] J.G. Zheng, X. Pan, M. Schweizer, F. Zhou, U. Weinmar, W. Gopel and M. Ruhle, *J. Appl. Phys.* **79** (1996) 7688.

[221] B.W. van de Waal, *J. Cryst. Growth* **158** (1996) 153.

[222] C.J. Morrissey. *Mineral Specimens.* Iliffe Books, London, 1968.

[223] C.S. Strom, R.F.P. Grimbergen, I.D.K. Hiralal, B.G. Koenders and P. Pennema, *J. Cryst. Growth* **149** (1995) 96.

[224] A.C. Bishop. *An Outline of Crystal Morphology.* Hutchinson, London, 1967.

[225] W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling. *Numerical Recipes in C.* Cambridge University Press, Cambridge, 1988.

[226] J. Lima-de-Faria. *Structural Mineralogy: An Introduction.* Kluwer, Dordrecht, 1994.

[227] H.A. Miers. *Mineralogy.* MacMillian, New York, 1902.

[228] L.G. Berny and B. Mason. *Mineralogy.* Freeman, San Franscisco, 1959.

[229] E.H. Kraus, W.F. Hunt and L.S. Ramsdell. *Mineralogy.* McGraw-Hill, New York, 1959.

[230] M.H. Battey. *Mineralogy for Students.* Longman, New York, 1981.

[231] M.J. Davies, P.R. Kenway, P.J. Lawrence, S.C. Parker, P.W. Tasker and W.C. Mackrodt, *J. Chem. Soc. Faraday Trans.* **85** (1989) 555.

[232] M.L. Connolly, *J. Appl. Cryst.* **16** (1983) 548.

[233] D.R. Lide (ed.). *CRC Handbook of Chemistry and Physics.* CRC Press, Boca Raton, Florida, 1995.

[234] M. Levitt, M. Hirshberg, R. Sharon, K.E. Laidig and V. Daggett, *J. Phys. Chem.* **101** (1997) 5051.

[235] A. Laaksonen, P.G. Kusalik and I.M. Svishchev, *J. Phys. Chem. A* **101** (1997) 5910.

[236] R.L. Mancera, A.D. Buckingham and N.T. Skipper, *J. Chem. Soc., Faraday Trans.* **93** (1997) 2263.

[237] L.X. Dang, *J. Am. Chem. Soc.* **117** (1995) 6954.

[238] A. Tongraar, K.R. Liedl and B.M. Rode, *J. Phys. Chem. A* **101** (1997) 6299.

[239] P.G. Smith, H.R. Watling and P. Crew, *Colloids and Surfaces A: Physico. Eng. Aspects* **111** (1996) 119.

[240] S. Barlow, A.L. Rohl, S.G. Shi, C.M. Freeman and D. O'Hare, *J. Am. Chem. Soc.* **118** (1996) 75.

[241] D.M. Heyes, *J. Chem. Phys.* **74** (1981) 1924.

[242] A.S. Russell, J.D. Edwards and C.S. Taylor, *J. Met.* **7** (1955) 1123.

[243] D. Frenkel and B. Smit. *Understanding Molecular Simulation.* CA, San Diego, 1996.

[244] B.I. Whittington, T.M. Fallows and M.J. Willing, *Int. J. Miner. Process.* **49** (1997) 1.

[245] K.J. Laidler and J.H. Meiser. *Physical Chemistry.* Houghton Mifflin, Boston, 1995.

[246] M. Moskovits and K.H. Michaelian, *J. Am. Chem. Soc.* **102** (1980) 2209.

[247] S.K. Sharma and S.C. Kashyap, *J. Inorg. Nucl. Chem.* **34** (1972) 3623.

[248] P.D. Fawell and H.R. Watling, *Appl. Spectrosc.* **52** (1998) 1115.

Appendix A

Computer Codes

A brief description of the major codes developed during the course of this project is presented. Where appropriate, suitable input and output samples are also shown.

A.1 C

A.1.1 Dynamics Initializer

Description

This program (MDI) was designed to create a complete liquid simulation cell, given the individual constituents and the amounts of each required. The total number of lattice points (molecular sites) is determined from a specified parameter L, so that the total number of molecules in the box is L^3. Thus, L is simply the dimension of the simulation cube in lattice points. The actual physical dimensions of the box is automatically determined when the constituents are read in at run time. The radius of the largest constituent is used to ensure that no overlap occurs. The essential idea of the program is to separate the constituents into one solvent and multiple solute particles. The default particle at each lattice point is thus a solvent molecule. The box is then filled, one solute molecule at a time, so that each particle is maximally dispersed from all previously placed particles. This is accomplished by attempting to maximize the shell of solvent that will be around the newly inserted particle. Typically, there are a number of possible locations for placement, one of which is selected at random. In addition, before the solute molecule is placed in the lattice, it is randomly rotated.

The program is executed as "MDI <INPUT_FILE>", where a typical input file consists of lines such as:

```
boxdim 5
solv water.car
comp aluminate.car 5
output soup.car
```

The first line is the dimensions of the cube in lattice points. The second line indicates the BIOSYM coordinates data file to be used as solvent. The third line indicates a solute BIOSYM file plus the number of such species desired in the lattice. This command may be repeated when multiple solute species are required. The last line indicates the BIOSYM filename desired for the resultant simulation cell. Note that the MSI code used in the molecular dynamics studies also requires an MDF (molecular data file) indicating molecular connectivities. A separate Perl script is available that creates a suitable MDF file for the output simulation cell. After the program has executed, an output summary is presented:

```
chosen angstrom separation = 5.000000
Filling box: 5 x 5 x 5

aluminate.car : solvated 5 molecules
Number of solvent (water.car) molecules: 120
Output written successfully to: soup.car
```

Code listing

```
/* MDI - Molecular dynamics inititaliser */
/* v 0.2 */

#include <math.h>
#include <stdio.h>
#include <string.h>
#include <sys/times.h>
#include "const.h"
#include "struc.h"

int numcomp=0;    /* number of components to "dissolve"                */
                  /* NB: doesn't include solvent itself                */
int boxdim;       /* box dimension (ie side length in lattice pts)     */
float lattsep;    /* angstrom distance between lattice pts             */
int *compdat;     /* array indicating component type at every location */
int mkmdf=0;      /* make an mdf file?  1=yes 0=no                     */

char outfile[30];
char *get_line();
char *get_item(char *, int, int);

main (int argc, char *argv[])
{
```

```
int i,j;

if (argc<2)
  {
  printf("\n*** Molecular Dynamics Initializer v2.0 ***\n\n");
  printf("Description:\n-----------\n\n");
  printf("Creates a cubic lattice with specified components randomly \n");
  printf("placed at the lattice points. Some effort has been taken to \n");
  printf("ensure that the different components are reasonably dispersed,\n");
  printf("simulating solvation.\n\n");
  printf("Useage:\n------\n\t mdi input_file\n\n");
  printf("Keywords:  (input file)\n--------\n\n");
  printf("boxdim <integer>          = dimension of cube\n");
  printf("solv <filename>           = molecule to use as the solvent (BIOSYM car file)\n");
  printf("comp <filename> <integer> = solvate a specified number of molecules\n");
  printf("output <filename>         = destination file (BIOSYM arc format)\n\n");
  printf("Notes:\n-----\n\n");
  printf("1. comp is the only keyword that may appear more than once\n");
  printf("2. If you need an mdf file you can run mkmdf afterwards\n\n");
  return;
  }

/* load all the data */

if (read_inpfile(argc, argv) == 2 || read_datfile() == 2)
  {
  printf("Run unsuccessful!\n");
  return;
  }
else
  {

/* main array */

compdat = (int *) malloc(boxdim*boxdim*boxdim*sizeof(int));
if (!compdat)
  {
  printf("Unable to allocate memory for compdat!\n");
  return;
  }

/* Initialise random number generator */

j=times(buffer);
srand(j);
j=j % 87;
for (i=0 ; i<j ; i++)
  {
  rand();
  }
srand(rand());

/* Main routine - attempt to fill vacant lattice points */

  if (fill() == 2)
    {
    printf("Run unsuccessful!\n");
```

```
    return;
    }
  }

/* create car file */

write_dat();

/* create mdf file if requested */
if (mkmdf)
  {
  write_mdf();
  }

/* Print a run summary */

j=0;
for (i=1 ; i<=numcomp ; i++)
  {
  printf("%s : solvated %d molecules\n",comp[i].filename,comp[i].done);
  j += comp[i].done;
  }
i = boxdim*boxdim*boxdim - j;

printf("Number of solvent (%s) molecules: %d\n",comp[0].filename,i);
printf("Output written successfully to: %s\n",outfile);
}

/*******************/
/* Main subroutine */
/*******************/

int fill()
{
int numsites;    /* number of filled points */
int i,j,k,p,s,sitematch,midmatch;
int nummid,numcand,candpt[MAXMID];
int r2,r2min[MAXMID],rmaxmin,dx,dy,dz;
int reqtot,ci,ri,pos;

float f;
float ran2();

/* NB: initially every point in box is 0 => solvent */

printf("Filling box: %d x %d x %d\n",boxdim,boxdim,boxdim);

numsites = 0;
for (i=0 ; i<2 ; i++)
  {
  for (j=0 ; j<2 ; j++)
    {
    for (k=0 ; k<2 ; k++)
      {
      site[numsites].x = i*(boxdim-1);
      site[numsites].y = j*(boxdim-1);
      site[numsites].z = k*(boxdim-1);
```

```
      numsites++;
      }
    }
  }

/* force positioning of 1st component (in exact center of box) */

/* site update */
site[numsites].component = 1;
site[numsites].x = (int) boxdim/2;
site[numsites].y = (int) boxdim/2;
site[numsites].z = (int) boxdim/2;
comp[1].done++;

/* array update */
pos = site[numsites].x + (boxdim*site[numsites].y) + (boxdim*boxdim*site[numsites].z);
*(compdat+pos) = 1;

numsites++;

/* compute total number of components to dissolve */
reqtot=0;
for (i=1 ; i<=numcomp ; i++)    /* omit solvent */
  {
  reqtot += comp[i].reqnum;
  }

ci = 1;          /* index - components placed */

/* loop over total number to dissolve */
for (ri=1 ; ri<reqtot ; ri++)   /* start at 1, since already done 1 (above) */
  {

/* select a component at random */
/* if we've not exceeded the required number of that compinent */

  if (comp[ci].done == comp[ci].reqnum) ci++;   /* next component if done */
  comp[ci].done++;

/* loop1 generate midpoints */

k=0;   /* num unique midpoints */

/* loop over all unique pairs (i,j) of already dissolved components */
for (i=0 ; i<numsites-1 ; i++)
  {
  for (j=i+1 ; j<numsites ; j++)
    {
    midpt[k].x = (int) (site[i].x + site[j].x)/2;
    midpt[k].y = (int) (site[i].y + site[j].y)/2;
    midpt[k].z = (int) (site[i].z + site[j].z)/2;

/* check midpoint against existing sites */
    for (s=0 ; s<numsites ; s++)
      {
      sitematch = 0;
      if (midpt[k].x == site[s].x) sitematch++;
```

```
      if (midpt[k].y == site[s].y) sitematch++;
      if (midpt[k].z == site[s].z) sitematch++;
      if (sitematch == 3) break;
      }

/* valid mid point only if not equal to existing sites */
    if (sitematch < 3)
      {
      midmatch=0;              /* necessary for case k=0 */
      for (p=0 ; p<k ; p++)
        {
        midmatch = 0;
        if (midpt[k].x == midpt[p].x) midmatch++;
        if (midpt[k].y == midpt[p].y) midmatch++;
        if (midpt[k].z == midpt[p].z) midmatch++;
        if (midmatch == 3) break;
/* unique mid point only if not equal to existing midpoints */
        }
      if (midmatch < 3) k++;     /* valid => save as kth midpoint */
      }
    }
  }
nummid = k;

/* loop2 generate distances(sq'd) */

rmaxmin=0;

for (i=0 ; i<nummid ; i++)  /* midpoint */
  {
  r2min[i] = boxdim*boxdim;

  for (j=0 ; j<numsites ; j++) /* sites */
    {
/* correct for periodic nature of box */
    dx = abs(midpt[i].x - site[j].x);
    if (dx > boxdim/2)
      {
      dx -= boxdim;
      dx = abs(dx);
      }
    dy = abs(midpt[i].y - site[j].y);
    if (dy > boxdim/2)
      {
      dy -= boxdim;
      dy = abs(dy);
      }
    dz = abs(midpt[i].z - site[j].z);
    if (dz > boxdim/2)
      {
      dz -= boxdim;
      dz = abs(dz);
      }

/* 'reduced' distance squared */
    r2 = dx*dx;
    r2 += dy*dy;
```

```
    r2 += dz*dz;

/* local midpoint min */
    if (r2 < r2min[i]) r2min[i] = r2;
    }

/* determine the largest of the local midpoint minima */
  if (r2min[i] > rmaxmin) rmaxmin = r2min[i];
  }

/* examine all midpoints and only admit those */
/* with the largest local minima as candidate sites */

k=0;
for (i=0 ; i<nummid ; i++)
  if (r2min[i] == rmaxmin) candpt[k++] = i;

numcand = k;
if (!numcand)
  {
  printf("No candidate sites found!\n");
  return 2;
  }

/* Select (at random) a candidate site into */
/* which the current component will be placed */

f = ran2();
s = (int) (0.5 + numcand * f);

/* do site update */
site[numsites].component = ci;
site[numsites].x = midpt[candpt[s]].x;
site[numsites].y = midpt[candpt[s]].y;
site[numsites].z = midpt[candpt[s]].z;

/* do array data update */
pos = site[numsites].x + (boxdim*site[numsites].y) +
                         (boxdim*boxdim*site[numsites].z);
*(compdat+pos) = ci;
numsites++;

  }      /* for total number of component */

/* output data */

return 0;
}

/*********************************/
/* read setup data from input file */
/*********************************/

int read_inpfile(int argc, char *argv[])
{
int num;
```

```
char item1[10],item2[10],item3[10];

FILE *fp;

fp = fopen(argv[1],"r");
if (!fp)
  {
  printf("Can't open %s!\n",argv[1]);
  return 2;
  }

num = 0;
for(;;)
  {
  num = fscanf(fp,"%s",&item1);
  if (num == -1) break;

  if (strcmp(item1,"comp") == 0)
    {
    num = fscanf(fp,"%s %s",&item2,&item3);
    if (num != 2)
      {
      printf("Incomplete component line: %s\n",item1);
      return 2;
      }
    numcomp++;
    strcpy(comp[numcomp].filename, item2);
    comp[numcomp].reqnum = atoi(item3);
    comp[numcomp].done = 0;
    }
  else if (strcmp(item1,"solv") == 0)
    {
    num = fscanf(fp,"%s",&item2);
    if (num != 1)
      {
      printf("Incomplete component line: %s\n",item1);
      return 2;
      }
    strcpy(comp[0].filename, item2);
    }
  else if (strcmp(item1,"boxdim") == 0)
    {
    num = fscanf(fp,"%s",&item2);
    if (num != 1)
      {
      printf("Incomplete component line: %s\n",item1);
      return 2;
      }
    boxdim = atoi(item2);  /* side length */
    }
  else if (strcmp(item1,"output") == 0)
    {
    num = fscanf(fp,"%s",&item2);
    if (num != 1)
      {
      printf("Incomplete component line: %s\n",item1);
      return 2;
```

```
      }
    strcpy(outfile,  item2);
    }
  else if (strcmp(item1,"mdf") == 0)
    {
    mkmdf=1;
    }
  }          /* while */
}

/*****************/
/* Get .car data */
/*****************/

int read_datfile()
{
FILE *fp;
char *buff, dummy, data[MAXLINELEN];
int i,j,k,m,n,idx,eof_flag;

int r;
float r2,r2max;

buff = (char *) malloc(MAXITEMLEN*sizeof(char));

/* cycle through all components and load their car data */

r2max = 0;    /* global maximum (ie dimensions of largest molecule) */

for (idx=0 ; idx<=numcomp ; idx++)
  {
  fp = fopen(comp[idx].filename,"r");
  if (!fp)
    {
    printf("Error opening data file: %s\n",comp[idx].filename);
    return 2;
    }

/* read past header (assumes 4 lines - FIX ME) */
  i = 0;
  while(i<4)
    {
    fread(&dummy,sizeof(char),1,fp);
    if (dummy == '\n') i++;
    }

/* read .car data until end encountered - assume eof */
  i = j = eof_flag = 0;
  while(!eof_flag)
    {

/* if we read a line with no data before eof => bad data file */
    if (!fread(&data[i],sizeof(char),1,fp))
      {
      printf("Unexpected EOF!\n");
      if (!j)
        return 2;
```

```
      }

 /* process a line of data  - ie either a CR  */
 /* or max line length encountered - FIX ME */
     if (data[i] == '\n' || i==MAXLINELEN)
       {
       buff = get_item(data,1,1);
       if (!strncmp("end",buff,3*sizeof(char)))
         {
         eof_flag = 1;
         }
       else
         {
         if( buff[0] != '\n' )
           {
           k = 0;
           while(k < 8)
             {
             comp[idx].atom[j][k] = buff[k];
             k++;
             }
           buff = get_item(data,2,1);
           sscanf(buff,"%f12",&comp[idx].x[j]);
           buff = get_item(data,3,1);
           sscanf(buff,"%f12",&comp[idx].y[j]);
           buff = get_item(data,4,1);
           sscanf(buff,"%f12",&comp[idx].z[j]);
           buff = get_item(data,5,0);
           strcpy(comp[idx].tail[j],buff);
           j++;
           i = 0;
           }          /* if buff */
         }            /* if end */
       }              /* if (get line) */
     else
       i++;
     }                /* while not eof */
   close(fp);
   comp[idx].numatoms = j;

 /* calc. max distance between all atoms in component i */

   for (n=0 ; n<comp[idx].numatoms-1 ; n++)
     {
     for (m=n+1 ; m<comp[idx].numatoms ; m++)
       {
       r2 = (comp[idx].x[n]-comp[idx].x[m])*(comp[idx].x[n]-comp[idx].x[m]);
       r2 += (comp[idx].y[n]-comp[idx].y[m])*(comp[idx].y[n]-comp[idx].y[m]);
       r2 += (comp[idx].z[n]-comp[idx].z[m])*(comp[idx].z[n]-comp[idx].z[m]);
       if (r2 > r2max) r2max = r2;
       }
     }
   }                  /* for each component */

 r = (int) (1.0 + sqrt(r2max));
 lattsep = (float) r;
```

```
printf("chosen angstrom separation = %f\n",lattsep);

return 0;
}

/************************/
/* Save box to car file */
/************************/

int write_dat()
{
FILE *fp;

int f,s,n,pos;
float x,y,z,phi,theta,psi,start,stop,step;
float q[3],boxlen;

float ran2();

fp = fopen(outfile,"wt");

boxlen = boxdim * lattsep;

fprintf(fp,"!BIOSYM archive 3\nPBC=ON\n");
fprintf(fp,"Multiple component MD Initialisation\n!DATE\n");
fprintf(fp,"PBC %f %f %f 90.0 90.0 90.0 (P 1)\n",boxlen,boxlen,boxlen);

start = lattsep/2;
stop = start + boxdim*lattsep ;
step = lattsep;

pos=0; /* pointer - current array position */

for (z=start ; z<stop ; z+=step)
  {
  for (y=start ; y<stop ; y+=step)
    {
    for (x=start ; x<stop ; x+=step)
      {
      s = *(compdat+pos);        /* get component at this (x,y,z) pos'n */
      pos++;

/* create some random rotations */
      phi = 2.0*PI*ran2();
      theta = PI - 2.0*PI*ran2();
      psi = 2.0*PI*ran2();

      for (n=0 ; n<comp[s].numatoms ; n++)
        {
/* do random rotation of coords */
        q[0] = comp[s].x[n];
        q[1] = comp[s].y[n];
        q[2] = comp[s].z[n];
        rot(&q[0],phi,theta,psi);
/* do translation of coords */
        q[0] += x;
        q[1] += y;
```

```
        q[2] += z;
/* print adjusted coords to file */
        for (f=0 ; f<8 ; f++)
          {
          if (comp[s].atom[n][f] != '\0')
            fprintf(fp,"%c",comp[s].atom[n][f]);
          else
            fprintf(fp," ");
          }
        fprintf(fp," ");

        if (q[0] > -0.000001 && q[0] < 10.0)
          fprintf(fp," ");
        fprintf(fp,"%f11    ",q[0]);

        if (q[1] > -0.000001 && q[1] < 10.0)
          fprintf(fp," ");
        fprintf(fp,"%f11    ",q[1]);

        if (q[2] > -0.000001 && q[2] < 10.0)
          fprintf(fp," ");
        fprintf(fp,"%f11 ",q[2]);

        fprintf(fp,"%s\n",comp[s].tail[n]);
        }
/* arc file format */
      fprintf(fp,"end\n");

      }
    }
  }

/* arc file format */
fprintf(fp,"end\n");
/* car file format - makes it difficult to create mdf file after */
/* fprintf(fp,"end\nend\n"); */
close(fp);

}

/*********************/
/* mdf file creation */
/*********************/

write_mdf()
{
printf("To make an mdf file you will have to type:\n\n");
printf("\tmkmdf input_file\n\n");
}

/************************************************/
/* read a line of data from specified file pointer */
/************************************************/

char *get_line(FILE *fp)

{
```

```
char *data;
int i;

data = (char *) malloc(sizeof(char)*MAXLEN);

if (data == NULL)
  {
  printf("get_line() : out of memory!\n");
  }

i=0;
do
  {
  if(fread(data+i,sizeof(char),1,fp) != 1)
    {
    printf("An error occurred reading file!\n");

printf("%d\n",i);
printf("%s\n",data);

    return;
    }
  }
while(data[i] != '\n');

data[i] = '\0';

return(data);
}

/************************************************/
/* Find the num-th item in an input line of text */
/************************************************/

/* FIXME - this fails with core dump if *line is an empty string */

char *get_item(char *line, int num, int op)

/* op = 1 - return num item only                      */
/* op = 0 - return num,num+1,num+2,... items as well */

{
int i,j,n,len;
char *item;

item = (char *) malloc(MAXITEMLEN*sizeof(char));

len = str_len(line);

if (item == NULL)
{
printf("Not enough memory!\n");
return;
}

j = 0;
for (n=0 ; n<num ; n++)
```

```
  {
  i = j;
  while(line[i] == ' ')
    i++;

  j = i;
  while(line[j] != ' ' && j<len)
    j++;
  }

/* Crude fix for finding the end of the input data */

if (!op)
  {
  j=80;
  while (line[j] >= '0' && line[j] <= '9')
    j++;
  }

strncpy(item,&line[i],j-i);

item[j-i+1] = '\0';
return item;
}

/*************************/
/* get length of a string */
/*************************/

int str_len(char *data)

{
int i=0;

do
  {
  i++;
  }
while(*(data+i) != '\0');

return(i);
}

/*********************/
/* My random routine */
/*********************/

float ran2()
{
int i,j;
float f;

/* generate 2 random numbers */

i = rand();
j = rand();
```

```
/* make range 0-1 */

if (i < j)
  f = (float) i / (float) j;
else
  f = (float) j / (float) i;

return f;
}

/******************/
/* Debugging aids */
/******************/

test1()
{
int i,j;

for (i=0 ; i<=numcomp ; i++)
  {
  printf("component %d (%s) : require %d\n",i,comp[i].filename,comp[i].reqnum);
  printf("\tcontains %d atoms:\n",comp[i].numatoms);
  for (j=0 ; j<comp[i].numatoms ; j++)
    {
    printf("\t\tatom %d : %s at (%f,%f,%f)\n",j,comp[i].atom[j],comp[i].x[j],
                                              comp[i].y[j],comp[i].z[j]);
    }
  }
}

test2(int num)
{
int i;

printf("Filled sites: %d\n",num);
for (i=0 ; i<num ; i++)
  {
  printf("(%d,%d,%d)\n",site[i].x,site[i].y,site[i].z);
  }
}
```

A.2 Perl

A.2.1 Surfaces

Description

The COMPSHIFT program was written to automatically determine the valid shifts for a given crystal structure and list of faces to examine. The program relies on the MARVIN code for creating an initial surface, with a default shift of 0. The z-coordinates of all atoms (and molecule centroids) are then used to

generate a range of candidate shifts. Each cut is then tested using the MARVIN code to see if a perpendicular dipole moment exists. The program is run as "COMPSHIFT <INPUT_FILE.MVN> <MILLER_PLANES>" with the first file a suitable MARVIN input file, and the second a list of faces (one per line) to examine. For gibbsite, the output from the program is:

(hkl)	D(hkl)	Area	Shift	Esurf(un)	Eatt(un)	Eslice(un)
(002)	4.8675	43.9896	0.0000	17.6236	-8.4374	-18.2832
(002)	4.8675	43.9896	0.2500	0.3234	-0.1472	-26.5734
(200)	4.3810	48.8751	0.0000	1.7830	-0.9036	-25.8170
(200)	4.3810	48.8751	0.2500	5.3936	-2.8628	-23.8578
(110)	4.3384	98.7106	0.0000	3.6785	-1.8987	-24.8219
(110)	4.3384	98.7106	0.3795	5.9341	-3.0316	-23.6890
(110)	4.3384	98.7106	0.5001	4.6427	-2.4077	-24.3128
(110)	4.3384	98.7106	0.6207	5.9341	-3.0316	-23.6890
(101)	6.2006	69.0644	0.0000	1.1829	-0.4233	-26.2972
(101)	6.2006	69.0644	0.2456	2.8744	-1.0316	-25.6890
(101)	6.2006	69.0644	0.5000	1.2882	-0.4616	-26.2590
(101)	6.2006	69.0644	0.7544	2.8744	-1.0316	-25.6890
$(10\bar{1})$	6.8769	62.2723	0.0000	1.2858	-0.4160	-26.3046
$(10\bar{1})$	6.8769	62.2723	0.2913	3.1978	-1.0349	-25.6857
$(10\bar{1})$	6.8769	62.2723	0.5000	1.3491	-0.4366	-26.2840
$(10\bar{1})$	6.8769	62.2723	0.7088	3.1978	-1.0349	-25.6857
(112)	3.1591	135.5587	0.0000	1.1828	-0.8442	-25.8764
(112)	3.1591	135.5587	0.5001	2.9475	-2.0783	-24.6422
$(11\bar{2})$	3.3246	128.8102	0.0000	1.3448	-0.9110	-25.8096
$(11\bar{2})$	3.3246	128.8102	0.5001	3.5930	-2.4193	-24.3012

Code listing

```
#! /usr/bin/perl

# compshift v0.1

# computes possible shifts for a list of faces by
# eliminating those with non zero z dipole moment

$MARVIN = "/usr/local/bin/marvin";

$dz = 0.00025;        # z coord + this = cut
$dcut = 0.0005;       # unique shift separation
$dzdip = 0.005;       # judge zero dipole
```

```
$DEBUG=0;

if (!@ARGV[1])
  {
  print "Usage:  compshift <marvin_file> <plane_file>\n\n";
  print "plane_file should contain miller indices (one face per line).\n";
  }

$inp_file = $ARGV[0];
$hkl_file = $ARGV[1];
print "(h k l)    D(hkl)   Area     Shift    Esurf(unre) Eatt(unre)  Eslice(unre)\n";

$_ = $inp_file;
s/\./ /g;
split(" ");
$inp_base = $_[0];

open (INP,"$hkl_file");

while(<INP>)
  {
  s/,/ /g;
  chop;
  $miller = $_;
  s/ //g;
  if (!$_)
    {
    exit;
    }

# get smallest common (nonzero) multiple in (hkl)
  $scm = &simplify;

# get all possible shifts for this face
  $num_shifts = &get_shifts;

# which shifts have zero z-dipole
  &comp_zdip;

# compute info for those shifts
  &comp_data;

# output results
  print
"-------------------------------------------------------------------------------\n";
  for ($i=0 ; $i<$num_zd ; $i++)
    {
    print "($miller)  ";
    $zd_dspace[$i] = $zd_dhkl[$i]*$zd_latt[$i]/$scm;
    printf "%8.4f  ",$zd_dspace[$i];
    printf "%7.4f  ",$zd_area[$i];
    printf "%7.4f  ",$zd_shift[$i];
    printf "%7.4f  ",$zd_esurf[$i];
    printf "%10.4f  ",$zd_eatt[$i];
    printf "%10.4f  ",$zd_eslice[$i];
    print "\n";
```

```
    }

  }

######################

sub simplify
  {
  $_ = $miller;
  split(' ');
  $a = &scf($_[0],$_[1]);
  $b = &scf($_[1],$_[2]);
  $c = &scf($a,$b);

if ($DEBUG)
  {
  print "Doing ($miller)\n";
  print "Smallest common factor: $c\n";
  }

  return($c);
  }

######################

sub scf
  {
  if ($_[1] == 0)
    {
    return(abs($_[0]));
    }
  else
    {
    return(&scf($_[1],$_[0]%$_[1]));
    }
  }

######################

sub get_shifts
  {
  $tmp_file = $inp_base."_1.mvn";

  system("cp $inp_file $tmp_file\n");

  open (OUT,">>$tmp_file");
  print OUT "surface\n";
  print OUT "miller a $miller\n";
  print OUT "region 1 0 0 0 \n";
  print OUT "shift 0.0\n";
  print OUT "noenergy\n";
  print OUT "print molecule\n";
  close OUT;

  open (MVN,"-|") || exec("$MARVIN $tmp_file");
  while (<MVN>)
    {
```

```
    if (/Layer Thickness/)
      {
      split(' ');
      $dhkl = $_[2];
      }
    if (/BLOCK A  REGION 1/)
      {
      $i=0;
      $nummols=0;
      while(<MVN>)
        {
        split(' ');
        if ($_[2] eq "core" || $_[2] eq "shel" || $_[2] eq "shell")
          {
# print "$_[1]  ($_[3] $_[4] $_[5])  $_[7]\n";
          $mol[$i] = $_[7];

# init for molecule calculation
          $zcent[$mol[$i]] = 0;
          $molsize[$mol[$i]] = 0;

          if ($mol[$i] > $nummols)
            {
            $nummols = $mol[$i];
            }
          $z[$i] = $_[5];
          $i++;
          }
        }         # while MVN
      }           # if BLOCK A  REGION 1
    }             # while MVN

# finished with input file
  unlink $tmp_file;

  $numatoms = $i;

if ($DEBUG)
  {
  print "Found $numatoms atoms\n";
  print "Found $nummols molecules\n";
  }

# get zcuts for lone atoms and start centroid calculations for molecules
  $j=0;
  $zmax = -9999;
  for ($i=0 ; $i<$numatoms ; $i++)
    {
    if ($mol[$i])
      {
      $zcent[$mol[$i]] += $z[$i];
      $molsize[$mol[$i]]++;
      }
    else
      {
      $zcut[$j] = $z[$i];
      if ($zcut[$j] > $zmax)
```

```
        {
        $zmax = $zcut[$j];
        }
      $j++;
      }
    }

# complete molecule centroid calculations
  for ($i=1 ; $i<=$nummols ; $i++)
    {
    $zcut[$j] = $zcent[$i]/$molsize[$i];
    if ($zcut[$j] > $zmax)
      {
      $zmax = $zcut[$j];
      }
    $j++;
    }
  $numcuts = $j;

if ($DEBUG)
  {
  print "Found $numcuts possible cuts\n";
  print "Maximum z value: $zmax\n";
  print "Sorting...\n";
  }

  &zsort;

# convert from physical z coords to a shift value
  for ($i=0 ; $i<$numcuts ; $i++)
     {
# +ve shift direction is in -ve z direction (ie invert)
     $cut[$i] = -1.0*(($zcut[$i] - $dz)/$dhkl);
     while ($cut[$i] < 0.0)
       {
       $cut[$i] += 1.0;
       }
     }

# eliminate any equivalent cuts -> shift□
  $shift[0] = 0.0;      # add this as a special case
  $shift[1] = $cut[0];
  $j=$totcuts=2;
  $ref=0;
  for ($i=1 ; $i<$numcuts ; $i++)
    {
    if ( ($cut[$i] - $cut[$ref]) > $dcut )
      {
#print "shift: $cut[$i]\n";
      $shift[$j++] = $cut[$i];
      $totcuts++;
      $ref = $i;
      }
    }

if ($DEBUG)
  {
```

```
  print "Found $totcuts unique shift values\n";
  }

# some more cut processing

  $flag=0;
  for ($i=$totcuts-1 ; $i>1 ; $i--)
    {
# make cuts a little nicer by taking midpoints
    $sh = ($shift[$i] + $shift[$i-1])/2.0;
# eliminate any cuts due to common multiple in h k l
    if ($sh < 1.0/$scm && !$flag)
      {
      $mark = $i+1;
      $flag=1;
      }
    $shift[$i] = $sh;
    }
# were any eliminated?
  if ($mark < $totcuts)
    {
    $totcuts=$mark;
    }

if ($DEBUG)
  {
  print "Found $totcuts symmetry reduced shift values\n";
  }

  }               # end sub

#############

sub zsort
  {
  $swap=1;
  while ($swap)
    {
    $swap=0;
    for ($i=1 ; $i<$numcuts ; $i++)
      {
      if ($zcut[$i-1] < $zcut[$i])
        {
        $tmp = $zcut[$i-1];
        $zcut[$i-1] = $zcut[$i];
        $zcut[$i] = $tmp;
        $swap=1;
        }
      }
    }
  }

##############

sub comp_zdip
  {
# compute number of zero z-dipole cuts
```

```
  $j=0;

  for ($i=0 ; $i<$totcuts ; $i++)
    {
    $tmp_file = $inp_base."_$i.mvn";
    system("cp $inp_file $tmp_file\n");
    open (OUT,">>$tmp_file");
    print OUT "surface\n";
    print OUT "miller a $miller\n";
    print OUT "region 1 0 0 0 \n";
# NEW
    print OUT "depth a 10.0\n";
    print OUT "shift $shift[$i]\n";
    print OUT "noenergy\n";
    print OUT "print molecule\n";
    close OUT;

    open (MVN,"-|") || exec("$MARVIN $tmp_file");
    while (<MVN>)
      {
      if (/Surface Dipole/)
        {
        split(' ');
        $zdip = abs($_[4]);
        if ($zdip < $dzdip)
          {
          $zd_shift[$j] = $shift[$i];
          $zd_dipz[$j] = $_[4];
          $j++;
          }
        }
      }
# finished with file
    unlink $tmp_file;
    }

  $num_zd = $j;
  }

######################

sub comp_data
  {

  for ($i=0 ; $i<$num_zd ; $i++)
    {
    $tmp_file = $inp_base."_$i.mvn";
    system("cp $inp_file $tmp_file\n");
    open (OUT,">>$tmp_file");
    print OUT "surface\n";
    print OUT "miller a $miller\n";
    print OUT "region 1 0 0 0 \n";
# NEW
    print OUT "depth a 10.0\n";
    printf OUT "shift %7.4f\n",$zd_shift[$i];
    print OUT "nominimize\n";
    print OUT "print molecule\n";
```

```
        close OUT;

        open (MVN,"-|") || exec("$MARVIN $tmp_file");
        while (<MVN>)
          {
          if (/Surface Area/)
            {
            split(' ');
            $zd_area[$i] = $_[2];
            }
          if (/Lattice Constant/)
            {
            split(' ');
            $zd_latt[$i] = $_[2];
            }
          if (/Layer Thickness/)
            {
            split(' ');
            $zd_dhkl[$i] = $_[2];
            }
          if (/Surface Energy/)
            {
            split(' ');
            $zd_esurf[$i] = $_[3];
            }
          if (/Attachment Energy/)
            {
            split(' ');
            $zd_eatt[$i] = $_[3];
            }
          if (/Slice Energy/)
            {
            split(' ');
            $zd_eslice[$i] = $_[3];
            }
          }

    # finished with input file
        unlink $tmp_file;
        }
      }
```

A.2.2 Twinning

Description

The GTWIN program was written to automate the procedure of computing the attachment energy of a growth slice, then inverting that same growth slice and recalculating the attachment energy. The difference between the two was used to determine if twinning was energetically feasible. The program uses the SIMPLEX minimization method to find the most favourable location of the reflected growth slice on the crystal surface. Thus, the function to be minimized is the total energy

from a MARVIN run, which is computed for the reflected growth slice at some position (x, y, z) above the surface. To attempt to enforce a more rigorous search for the global minimum, the program runs several times with a different starting position of the reflected slice. These starting positions are mesh points on a grid that spans the repeat surface of the crystal. The number of mesh points per repeat vector length is an adjustable program parameter.

A sample invocation of the program "GTWIN GIBBSITE.MVN RESULTS_FILE 002, 0.0", (the last two entries are the miller index and the corresponding shift to use) produces the following output file:

```
*** RESULTS FILE ***

Label: none
Miller index: (002)
          Delta twin energy: -1.6182 eV/unit
    Total E for normal slice: -1282.5871 eV/unit
Detected number of molecules: 24
         Translation vector: (2.98914,-1.10708,-0.08651) Angs.
```

An additional program AUTOTWIN, is also available which accepts one input file specifying a list of faces, and shift values. It then invokes as many GTWIN calls as required, thus completely automating the computation of twinning energies.

Code listing

```
#! /usr/bin/perl
# performs norm & twin comparison using specified MARVIN file
# starts at different locations to try and seek the global min.

# v0.99
# latest - elegant fix to SG/Linux NULL 1st char problem

if (!$ARGV[1])
  {
  print "Useage: gtwin marvin_file output_file [autotwin data]\n";
  exit;
  }

# [autotwin data] = <miller>, <fractional_cutoff>, <label>

# miller
if (!$ARGV[2])
  {
  $ARGV[2] = "not specified";
  }
# cutoff
```

```
if (!$ARGV[3])
  {
  $ARGV[3] = "0";
  }
# label
if (!$ARGV[4])
  {
  $ARGV[4] = "none";
  }

########################
# INITIALISE CONSTANTS #
########################

# FIXED - should no longer be necessary
#computer system
# 0 - linux
# 1 - unix
# $OS=1;

#numerical parameters
$NUM_DIM = 3;
$NUM_PTS = 4;
$TOL_MIN = 0.005;
$ALPHA = 1.0;
$BETA = 0.5;
$GAMMA = 2.0;
$MIN_ENERGY = 7777777.77;
$GRID_SIZE = 3;
@BVEC = (0.0,0.0,0.3, 0.3,0.3,0.2, 0.3,-0.3,0.2, -0.3,0.0,0.2);

#temporary filenames
$MVN_ORIG = "base.mvn";
$SRC_ORIG = "base.mvn-r";
$MVN_TMP = "base_xlat.mvn";
$SRC_TMP = "base_xlat.mvn-r";
$MOT_TMP = "base_xlat.mot";

#misc data
@pattern = ("coordinates 1 A","coordinates 1 B","end","ENERGY BREAKDOWN",
            "Surface Vectors:","Layer Thickness:","BLOCK B BASIS","-----");

####################
# SET UP FILENAMES #
####################

$inp_file = $src_file = $src_norm = $src_twin = $base = @ARGV[0];
$out_file = @ARGV[1];
$inp_file =~ s/.mvn/_temp.mvn/;
$src_file =~ s/.mvn/.mvn-r/;
$src_norm =~ s/.mvn/_norm.mvn-r/;
$src_twin =~ s/.mvn/_twin.mvn-r/;
$base =~ s/.mvn//;
$before = join("",$base,"_before");
$after = join("",$base,"_after");
$bulk = join("",$base,"_bulk");
```

```
####################################
# GENERATE MARVIN OUTPUT FOR SOURCE #
####################################

system("cp @ARGV[0] $inp_file");
open (OUT, ">>$inp_file") || die "Can't open input file!";
print OUT "\n";
print OUT "region 1 1 1 1\n";
# comment out 1 line below if manually modified .mvn-r file is required
print OUT "output before marvin $base\n";
close OUT;
system("/usr/local/bin/marvin $inp_file garbage.mot");

####################################
# GET THE SYSTEM'S LATTICE VECTORS #
####################################

open(INP,"garbage.mot");

while(<INP>)
  {
# scan for surface vectors
  if (/$pattern[4]/)
    {
    @vec = split(' ');
    $lattice_mag[0] = sqrt($vec[2]*$vec[2]+$vec[3]*$vec[3]);
    $_ = <INP>;
    @vec = split(' ');
    $lattice_mag[1] = sqrt($vec[0]*$vec[0]+$vec[1]*$vec[1]);
    }
# scan for layer thickness
  if (/$pattern[5]/)
    {
    @vec = split(' ');
    $lattice_mag[2] = sqrt($vec[2]*$vec[2]);
    }
  }
close (INP);

# Setup initialisation vector BVEC
# and automate the Z displacement

for ($j=0 ; $j<$NUM_PTS ; $j++)
  {
  for ($i=0 ; $i<$NUM_DIM-1 ; $i++)
    {
    $BVEC[$j*$NUM_DIM+$i] *= $lattice_mag[$i]/$GRID_SIZE;
    }

#Z coord must be done separately (not part of grid)
    $BVEC[$j*$NUM_DIM+$NUM_DIM-1] *= $lattice_mag[$NUM_DIM-1];

  }

################################################
# PARSE TO GENERATE NORMAL AND TWIN SOURCE FILE #
################################################
```

```
open (INP, $src_file);
open (OUT1,">$src_norm");
open (OUT2,">$src_twin");

# establish cutoff for trimming

$cutoff = $lattice_mag[$NUM_DIM-1]*$ARGV[3];

while (<INP>)
  {
  if (/$pattern[1]/)
    {
# keep 1 B as it is
    print OUT1 "coordinates 1 B\n";
    $_ = <INP>;
    until (/$pattern[2]/)
      {
      @line = split(' ');
      $_ = join(" ",$line[0],$line[1],$line[2],
                    $line[3],$line[4]);

      if ($cutoff)
        {
        if ($line[4] <= $cutoff)
          {
          print OUT1 "$_\n";
          }
        }
      else
        {
        print OUT1 "$_\n";
        }

      $_ = <INP>;
      }
    }
  if (/$pattern[0]/)
    {
# do twin source file - reflect 1 A
    print OUT2 "coordinates 1 B\n";
    $_ = <INP>;
    until (/$pattern[2]/)
      {
      @line = split(' ');
# reflect
      $line[4] *= -1.0;

      $_ = join(" ",$line[0],$line[1],$line[2],
                    $line[3],$line[4]);

      if ($cutoff)
        {
        if ($line[4] <= $cutoff)
          {
          print OUT2 "$_\n";
          }
```

```
        }
      else
        {
        print OUT2 "$_\n";
        }

      $_ = <INP>;
      }
    }
  }
print OUT1 "end\n";
print OUT2 "end\n";
close INP;
close OUT1;
close OUT2;

##########################################
# CYCLE THROUGH GRID OF STARTING LOCATIONS #
##########################################

for ($gx=0 ; $gx<$GRID_SIZE ; $gx++)
  {
  for ($gy=0 ; $gy<$GRID_SIZE ; $gy++)
    {

$gvec[0] = $lattice_mag[0]*$gx/$GRID_SIZE;
$gvec[1] = $lattice_mag[1]*$gy/$GRID_SIZE;

@init_vec =
($BVEC[0]+$gvec[0],$BVEC[1]+$gvec[1],$BVEC[2],
$BVEC[3]+$gvec[0],$BVEC[4]+$gvec[1],$BVEC[5],
$BVEC[6]+$gvec[0],$BVEC[7]+$gvec[1],$BVEC[8],
$BVEC[9]+$gvec[0],$BVEC[10]+$gvec[1],$BVEC[11]);

########################
# SIMPLEX MINIMISATION #
########################

# generate initial function values

system("cp $src_twin $SRC_ORIG");
system("cp @ARGV[0] $MVN_ORIG");

@p = @init_vec;

for ($i=0 ; $i<$NUM_PTS ; $i++)
   {
   $y[$i] = &function(@p[3*$i,3*$i+1,3*$i+2]);
   }

# generate sum

for ($i=0 ; $i<$NUM_DIM ; $i++)
  {
  $sum = 0.0;
  for ($j=0 ; $j<$NUM_PTS ; $j++)
    {
```

```
    $sum += $p[$NUM_DIM*$j+$i];
    }
  $psum[$i] = $sum;
  }

# main loop

$flag = 0;
while (!$flag)
  {

# flag the highest, next hightest and lowest values

   if ($y[0] < $y[1])
     {
     $hi_ptr = 1;
     $nhi_ptr = 0;
     $lo_ptr = 0;
     }
   else
     {
     $hi_ptr = 0;
     $nhi_ptr = 1;
     $lo_ptr = 1;
     }

   for ($i=2 ; $i<$NUM_PTS ; $i++)
     {
     if ($y[$i] < $y[$lo_ptr])
       {
       $lo_ptr = $i;
       }
     if ($y[$i] > $y[$hi_ptr])
       {
       $nhi_ptr = $hi_ptr;
       $hi_ptr = $i;
       }
     if ($y[$i] > $y[$nhi_ptr] && $i != $hi_ptr)
       {
       $nhi_ptr = $i;
       }
     }

# Now that points are ranked - compute difference

  $ydiff = $y[$hi_ptr] - $y[$lo_ptr];
  $rtol = &abs($ydiff);

# Test if difference satisfies tolerance

  if ($rtol < $TOL_MIN)
    {
    last;  # yes! break out of while loop
    }

  $y_test = &reflect(-$ALPHA);
```

```
  if ($y_test <= $y[$lo_ptr])
    {
    $y_test = &reflect($GAMMA);
    }
  elsif ($y_test >= $y[$nhi_ptr])
    {
    $y_save = $y[$hi_ptr];
    $y_test = &reflect($BETA);
    if ($y_test >= $y_save)
      {
      for ($i=0 ; $i<$NUM_PTS ; $i++)
        {
        if ($i != $lo_ptr)
           {
           for ($j=0 ; $j<$NUM_DIM ; $j++)
             {
             $psum[$j] = 0.5*($p[$NUM_DIM*$i+$j]+$p[$NUM_DIM*$lo_ptr+$j]);
             $p[$NUM_DIM*$i+$j] = $psum[$j];
             }
           $y[$i] = &function(@psum);
           }
        }

# recalculate psum

      for ($i=0 ; $i<$NUM_DIM ; $i++)
        {
        $sum = 0.0;
        for ($j=0 ; $j<$NUM_PTS ; $j++)
          {
          $sum += $p[$NUM_DIM*$j+$i];
          }
        $psum[$i] = $sum;
        }
      }
    }
  } # end while

# calculate twinned energy

system("cp $src_twin $SRC_ORIG");
system("cp @ARGV[0] $MVN_ORIG");

$y[0] = &function(@p[3*$lo_ptr,3*$lo_ptr+1,3*$lo_ptr+2]);

#DEBUG###
#print "Minimum energy for this cycle = $temp\n";
#print "Translation vector = $p[3*$lo_ptr] $p[3*$lo_ptr+1] $p[3*$lo_ptr+2]\n";
#DEBUG###

if ($y[0] < $MIN_ENERGY)
  {
  $MIN_ENERGY = $y[0];
  @min_vec = @p[3*$lo_ptr,3*$lo_ptr+1,3*$lo_ptr+2];
  }

##################
```

```
# END OF SIMPEX #
#################

  } # $gy loop
} # $gx loop

###################################
# GENERATE TWIN 'BEFORE' .arc FILE #
###################################

system("cp $src_twin $SRC_ORIG");
system("cp @ARGV[0] $MVN_ORIG");
open (OUT,">>$MVN_ORIG");
print OUT "\n";
print OUT "nooutput 2 a\noutput before biosym $before 2\n";
close OUT;
$dummy = &function(@BVEC[0,1,2]);

############################################
# Parse for number of molecules in region B #
############################################

open (INP,$MOT_TMP);
$_=<INP>;
#find appropriate section
until (/$pattern[6]/)
  {
  $_=<INP>;
  }
#skip header
for ($i=0 ; $i<4 ; $i++)
  {
  $_=<INP>;
  }
#read until end
until (/@pattern[7]/)
  {
  $save = $_;
  $_ = <INP>;
  }

$_ = $save;
@last = split(' ');
$twin_mols = $last[6];

###################################
# GENERATE TWIN 'AFTER' .arc FILE #
###################################

system("cp $src_twin $SRC_ORIG");
system("cp @ARGV[0] $MVN_ORIG");
open (OUT,">>$MVN_ORIG");
print OUT "\n";
print OUT "nooutput 2 a\noutput before biosym $after 2\n";
close OUT;
$ytwin = &function(@min_vec[0,1,2]);
```

```
############################
# GENERATE BULK .arc FILE #
############################

system("cp $src_norm $SRC_ORIG");
system("cp @ARGV[0] $MVN_ORIG");

open (OUT,">>$MVN_ORIG");
print OUT "\n";
print OUT "region 1 1 1 1\n";
print OUT "output before biosym $bulk 2\n";
close OUT;

$dummy = &function(0,0,0);

########################
# GENERATE NORM ENERGY #
########################

system("cp $src_norm $SRC_ORIG");
system("cp @ARGV[0] $MVN_ORIG");

$ynorm = &function(0,0,0);

##########################################
# Parse for number of molecules in region B #
##########################################

open (INP,$MOT_TMP);
$_=<INP>;
#find appropriate section
until (/$pattern[6]/)
  {
  $_=<INP>;
  }
#skip header
for ($i=0 ; $i<4 ; $i++)
  {
  $_=<INP>;
  }
#read until end
until (/@pattern[7]/)
  {
  $save = $_;
  $_ = <INP>;
  }

$_ = $save;
@last = split(' ');
$norm_mols = $last[6];

################
# SAVE RESULTS #
################

# Note: prints to $out_file must APPEND, since we are lumping
# all the results in the one file.
```

```
open(OUT,">>$out_file");
print OUT "Label: $ARGV[4]\n";
print OUT "Miller index: ($ARGV[2])\n";

# do some checks
if ($twin_mols != $norm_mols)
  {
  print OUT "\n***Warning: the number of molecules in region B for    ***\n";
  print OUT   "***         twinned and normal structures are unequal! ***\n";
  }
if ($twin_mols <= 0)
  {
#  print OUT "\n***Warning: no molecules in twin found! ***\n";
  $twin_mols = 0;
  }
if ($norm_mols <= 0)
  {
#  print OUT "\n***Warning: no molecules in norm found! ***\n";
  $norm_mols = 0;
  }

# compute difference in total region 1 energy
$diff = ($ynorm - $ytwin);

printf OUT "           Delta twin energy: %5.4f eV/unit\n",$diff;
printf OUT "    Total E for normal slice: %7.4f eV/unit\n",$ynorm;
printf OUT "Detected number of molecules: %d\n",$twin_mols;
printf OUT "          Translation vector: (%2.5f,%2.5f,%2.5f) Angs.\n\n",@min_vec[0,1,2];
close OUT;

# tidy up
#unlink $MVN_ORIG, $SRC_ORIG, $MVN_TMP, $SRC_TMP, $MOT_TMP,
#       $inp_file, $src_file, $src_norm, $src_twin, 'garbage.mot';

exit;

#-----------------------------------------

sub reflect
# this function extrapolates through the face of the simplex
# across from the worst point to obtain a trial replacement point

{
$fac = @_[0];
$fac1 = (1.0 - $fac)/$NUM_DIM;
$fac2 = $fac1 - $fac;

#print "reflect: fac = $fac $fac1 $fac2\n";
#print "reflect: psum = @psum\n";

for ($k=0 ; $k<$NUM_DIM ; $k++)
  {
  $p_test[$k] = $psum[$k]*$fac1-$p[3*$hi_ptr+$k]*$fac2;
  }

#print "reflect: p_test = @p_test\n";
```

```
$y_test = &function(@p_test);

if ($y_test < $y[$hi_ptr])
  {
  $y[$hi_ptr] = $y_test;
  for ($k=0 ; $k<$NUM_DIM ; $k++)
    {
    $psum[$k] += $p_test[$k] - $p[3*$hi_ptr+$k];
    $p[3*$hi_ptr+$k] = $p_test[$k];
    }
  }
$y_test *= 1;
}

#-----------------------------------------

sub function
# this accepts as input the x,y,z values and
# returns the resultant energy from a marvin run
# assumes the existance of marvin and source
# coordinates files
{

system("cp $MVN_ORIG $MVN_TMP");

# get translation vector
@vec = @_;

open(OUT,">>$MVN_TMP");
print OUT "\nsource $SRC_ORIG\n";
print OUT "translate b $vec[0] $vec[1] $vec[2]\n";
close OUT;
system("/usr/local/bin/marvin $MVN_TMP $MOT_TMP");

#parse output file for the total energy

$flg = 0;
open (INP,$MOT_TMP) || die "can't open $MOT_TMP\n";

while (<INP>)
  {
  if (/$pattern[3]/)
    {
    $flg = 1;
    for ($skip=0 ; $skip<10 ; $skip++)
      {
      $_ = <INP>;
      }
    @line = split(' ');
    last;
    }
  }
if (!$flg)
  {
  print "ERROR: ENERGY NOT FOUND\n";
  exit;
  }
```

```
$energy = $line[3];
}
#-----------------------------------
sub abs
{
if (@_[0] < 0)
  {
  @_[0] *= -1.0;
  }
else
  {
  @_[0] *= 1.0;
  }
}
#-----------------------------------
```

A.2.3 Accessibility

Description

The SCURF code was written to facilitate the determination of which hydrogen atoms (in a given slab of material) should be regarded as belonging to the surface. The essential idea is to loop over all atoms in the supplied slab, selecting only the desired atoms for further testing. Each selected atom then has a probe atom rolled over its surface. If the probe intersects the surface of any other atom (except the one being scanned), that particular contact point is assumed to be inaccessible. Note that the appropriate ionic radius was used for the probe, and the van der Waals radius was used for all other atoms.

The program is executed using "SCURF <INPUT FILE>". A typical input file should specify the coordinates data file (BIOSYM or MARVIN) of the surface slab, the atom type to test for accessibility, region cutoffs (optional), and the probe atom to use. A sample input is presented below:

```
input marvin gibb_r002_2x2.mvn-r
type H
probe Na
```

Upon execution, this produces the following output summary:

```
Parsing: gibb_r002_2x2.mvn-r
Found 1344 atoms
Found 24 atoms to test
Testing 1 atom type(s):  H
Probe atom type: NA
```

```
---------------------------
 OUTPUT SUMMARY
---------------------------
Type: H  Scanned: 24  Successful:  24

---------------------------
 OUTPUT SPECIFICS
---------------------------
H #142 : 30.56% accessible surface
H #286 : 27.63% accessible surface
H #21022 : 37.87% accessible surface
H #21166 : 31.43% accessible surface
H #140 : 31.14% accessible surface
H #284 : 31.29% accessible surface
H #21020 : 24.56% accessible surface
H #21164 : 24.71% accessible surface
H #143 : 24.56% accessible surface
H #287 : 32.46% accessible surface
H #21023 : 29.97% accessible surface
H #21167 : 38.89% accessible surface
H #141 : 22.95% accessible surface
H #285 : 22.95% accessible surface
H #21021 : 26.17% accessible surface
H #21165 : 26.17% accessible surface
H #144 : 32.16% accessible surface
H #288 : 25.73% accessible surface
H #21024 :  8.63% accessible surface
H #21168 :  1.02% accessible surface
H #0 :  0.44% accessible surface
H #283 :  0.44% accessible surface
H #21019 :  0.44% accessible surface
H #21163 :  0.44% accessible surface
```

Since a surface composed of four normal repeat units was used to generate the above listing, each set of four subsequent hydrogens are equivalent. The different accessibilities that occur are due to the artificial increase in exposure a hydrogen will gain if near an edge or corner. Hence, for all surfaces, a 2x2 construction was used for the purpose of accessibility determination. The true accessibility for

each unique hydrogen is the lowest of the four equivalents.

Code listing

```
#! /usr/bin/perl

# SCURF, scan surface
# determines what atoms in a .car/.mvn-r file are "accessible"
# to a given probe atom

if (!$ARGV[0])
  {
  print "Useage: scurf <input file>\n";
  exit;
  }

# radii - from marvin.par
# ionic for probes, VdW for all others
%size = ('AL',2.01,'O',1.40,'H',1.17,'NA',0.97,'K',1.33);

$probe = "NA";       # probe to scan the VdW surface (default)

# compute desired radius to use for surface accessability scan
# TODO - ionic vs vdw automatically determined?
foreach $atom (keys %size)
  {
  $rad{$atom} = $size{$atom};
  }

$pi = 3.141592654;
$da = $pi/18.0;      # angle coarseness - extended sphere sweep

# parse input file
$test_types=0;
$xlim=$ylim=$zlim=0;
$if=$nl=0;

&parse_input($ARGV[0]);

# check results of parse
if (!$if)
  {
  print "No input file specified!\n";
  exit;
  }

# open .arc data file
# possibility of parse MARVIN file?

$n=0;
$test_count=0;
if ($if == 1)
  {
  &parse_biosym($inp_file);
  }
if ($if == 2)
```

```
  {
  &parse_marvin($inp_file);
  }

# inform user
print "Parsing: $inp_file\n";
print "Found $n atoms\n";
print "Found $test_count atoms to test\n";
print "Testing $test_types atom type(s):  ";
for ($i=0 ; $i<$test_types ; $i++)
  {
  print "$ttype[$i]  ";
  $totnum{$ttype[$i]} = 0;
  $totok{$ttype[$i]} = 0;
  }
print "\n";
print "Probe atom type: $probe\n";

# loop over all atoms to test
for ($i=0 ; $i<$n ; $i++)
  {
  if (!$test_atom[$i])
    {
    next;
    }
  $freesurf[$i] = 0;
  $totsurf[$i] = 0;

#  print "Testing (# $i): $atype[$i]  $x[$i] $y[$i] $z[$i]\n";
  $candi=0;    # assume i not a cadidate

  $totnum{$atype[$i]}++;

# radius of extended VdW sphere
  $rexi = $rad{$atype[$i]} + $rad{$probe};

# loop over surface of the sphere
  for ($theta=0.0 ; $theta<$pi ; $theta+=$da)
    {
    for ($phi=0.0 ; $phi<2.0*$pi ; $phi+=$da)
      {
# convert spherical coords to cartesian
      $xi = $x[$i] + $rexi*sin($theta)*cos($phi);
      $yi = $y[$i] + $rexi*sin($theta)*sin($phi);
      $zi = $z[$i] + $rexi*cos($theta);
# loop over all other atoms
      for ($j=0 ; $j<$n ; $j++)
        {
        if ($i == $j)
          {
          next;
          }
# error trap
        if (!$rad{$atype[$j]})
          {
          print "Uknown atom type (# $j): ";
          print " ($atype[$j])";
```

```
          printf " at (%5.2f,%5.2f,%5.2f)\n",$x[$j],$y[$j],$z[$j];
          exit;
          }
# atom j's radius
        $rexj = $rad{$atype[$j]} + $rad{$probe};
# make squared + fudge factor
        $rexj_sq = $rexj*$rexj;

# distance to atom j
        $dist_sq = ($xi-$x[$j])*($xi-$x[$j]);
        $dist_sq += ($yi-$y[$j])*($yi-$y[$j]);
        $dist_sq += ($zi-$z[$j])*($zi-$z[$j]);
# distance to j is closer than j's extended radius
# point (theta,phi) is bad - skip to next (theta,phi)

#printf "(%5.2f,%5.2f,%5.2f) - (%5.2f,%5.2f,%5.2f)\n",$xi,$yi,$zi,$x[$j],$y[$j],$z[$j];
#print "$dist_sq  <  $rexj_sq\n";

        if ($dist_sq < $rexj_sq)
          {
          goto skip;
          }
        }    # for j

# no overlap with any jth atom => success!
$candi=1;
# measure % surface "free"
$freesurf[$i]++;

skip:
      $totsurf[$i]++;
      }      # for phi
    }        # for theta

bypass:
  if ($candi)
    {
#    print "OK!\n";
    $totok{$atype[$i]}++;
    }
  }          # for i

# OUTPUT SUMMARY

# CURR

print "\n";
print "---------------------------\n";
print " OUTPUT SUMMARY \n";
print "---------------------------\n";
for ($i=0 ; $i<$test_types ; $i++)
  {
  print "Type: $ttype[$i]  ";
  print "Scanned: $totnum{$ttype[$i]}  ";
  print "Successful:  $totok{$ttype[$i]}\n";
  }
print "\n";
```

```
print "--------------------------\n";
print " OUTPUT SPECIFICS \n";
print "--------------------------\n";

for ($i=0 ; $i<$n ; $i++)
  {
  if ($test_atom[$i])
    {
#  print "Freesurf: $freesurf[$i]\n";
#  print "Totsurf: $totsurf[$i]\n";
    $pcfree = 100 * $freesurf[$i]/$totsurf[$i];
# if marvin file print molecule number
    if ($if == 2)
      {
      print "$atype[$i] #$marnum[$i] : ";
      }
# otherwise use internal indexing
    else
      {
      print "$atype[$i] #$i : ";
      }
    printf "%5.2f",$pcfree;
    print "% accessible surface\n";
    }
  }
print "\n";

#---------------------------------------------

sub parse_input

{
open (INP,"$_[0]") || die "Can't open $_[0] for reading.\n";

while(<INP>)
  {
  if (/^input/)
    {
    split;
    if ($_[1] eq "biosym")
      {
      $inp_file = $_[2];
      $if=1;
      }
    if ($_[1] eq "marvin")
      {
      $inp_file = $_[2];
      $if=2;
      }
    }
  if (/^type/)
    {
    split;
    $i=1;
    while ($_[$i])
      {
      $ttype[$test_types] = $_[$i++];
```

```
      $ttype[$test_types++] =~ tr/a-z/A-Z/;
      }
    }
  if (/probe/)
    {
    split;
    $probe = $_[1];
    $probe =~ tr/a-z/A-Z/;
    }
  if (/^xlim/)
    {
    split;
    $xstart = $_[1];
    $xend = $_[2];
    $xlim=1;
    $nl++;
    }
  if (/^ylim/)
    {
    split;
    $ystart = $_[1];
    $yend = $_[2];
    $ylim=1;
    $nl++;
    }
  if (/^zlim/)
    {
    split;
    $zstart = $_[1];
    $zend = $_[2];
    $zlim=1;
    $nl++;
    }

  }
}

#-----------------------------------------------

sub parse_biosym
{

open (INP,"$_[0]") || die "Can't open $_[0] for reading.\n";

# skip BIOSYM header

do
  {
  $_ = <INP>;
  if ($fail++ > 20)
    {
# TODO better termination?
    print "Error: unknown BIOSYM header or unexpected EOF.\n";
    exit;
    }
  }
while(!/^!DATE/);
```

```
$_ = <INP>;
if (/^PBC/)
  {
# record data in PBC line
  split;
  $limit[0] = $_[1];
  $limit[1] = $_[2];
  $limit[2] = $_[3];
  print "PBC: $limit[0] $limit[1] $limit[2]\n";
  $_ = <INP>;
  }

# parse all atom position data

$n=0;
do
  {
  $readflag=1;
  $skip=0;
  split;
# blank line check
  if (!$_[0])
    {
    $skip=1;
    }
# ignore shells and lines starting with #
  $test = substr($_[4],-1,1);
  $test2 = substr($_[0],0,1);
  if ($test eq "S" || $test2 eq "#")
    {
    $skip=1;
    }

# termination of frame on double end only
  if (!$skip)
    {
  $exit=0;
  if (/end/)
    {
    $exit++;
# get next
    $_ = <INP>;
    if (/end/)   # double end - frame termination
      {
      $exit++;
      }
    else
      {
      $readflag=0;
      }
    }
  else
    {
# get atom data
    $x[$n] = $_[1];
    $y[$n] = $_[2];
```

```
    $z[$n] = $_[3];
    $atype[$n] = $_[7];
    $atype[$n] =~ tr/a-z/A-Z/;
# test if atom is of the type we want to test
    $match=0;
    if ($test_types)
      {
      for ($i=0 ; $i<$test_types ; $i++)
        {
        if ($atype[$n] eq $ttype[$i])
          {
          $match=1;
          }
        }
      }
# test if atom is within any boundaries
    if ($xlim)
      {
      if ($x[$n] > $xend || $x[$n] < $xstart) { $match = 0; }
      }
    if ($ylim)
      {
      if ($y[$n] > $yend || $y[$n] < $ystart) { $match = 0; }
      }
    if ($zlim)
      {
      if ($z[$n] > $zend || $z[$n] < $zstart) { $match = 0; }
      }
# match=1 => this atom is one we will test
    if ($match)
      {
      $test_atom[$n]=1;
      }
    else
      {
      $test_atom[$n]=0;
      }
# next atom
    $n++;
    }

    }   # if skip

  if ($readflag)
    {
    $_ = <INP>;
    }
  }
while($exit < 2);

}

#--------------------------------------------

sub parse_marvin
{
```

```
open (INP,"$_[0]") || die "Can't open $_[0] for reading.\n";

while(<INP>)
  {
# get rid of any lines with text only
  $save = $_;
  tr/a-z//d;
  tr/A-Z//d;
  tr/#//d;
  split;

# any data?
  if ($_[3])
    {
# restore (as data lines have text and numbers)
    $_ = $save;

    split(' ');
    $atype[$n] = $_[0];
    $atype[$n] =~ tr/0-9//d;
    $atype[$n] =~ tr/a-z/A-Z/;

    $marnum[$n] = $_[5];

    $x[$n] = $_[2];
    $y[$n] = $_[3];
    $z[$n] = $_[4];
# determine if atom is of the type(s) we want to test
    $match=0;
    if ($test_types)
      {
      for ($i=0 ; $i<$test_types ; $i++)
        {
        if ($atype[$n] eq $ttype[$i])
          {
          $match=1;
          }
        }
      }
# all are matches
    else
      {
      $match=1;
      }
# test if atom is within any boundaries
    if ($xlim)
      {
      if ($x[$n] > $xend || $x[$n] < $xstart) { $match = 0; }
      }
    if ($ylim)
      {
      if ($y[$n] > $yend || $y[$n] < $ystart) { $match = 0; }
      }
    if ($zlim)
      {
      if ($z[$n] > $zend || $z[$n] < $zstart) { $match = 0; }
      }
```

```
# match=1 => this atom is one we will test
    if ($match)
      {
      $test_atom[$n]=1;
      $test_count++;
      }
    $n++;
    }       # if !ignore
  }         # while
}
```