Chapter 7

Molecular Dynamics

This chapter contains results generated using molecular dynamics (MD) [128], a different technique from the static energy minimisation used in the previous chapters. With the MD technique it is possible to simulate the dynamic, thermal behaviour of atoms in a solid. By performing simulations at different temperatures and studying the displacements of the ions as a function of time we can predict diffusion coefficients and potentially fit Arrhenius equations.

Within the strict Born description of ionic solids the technique can only be used to perform simulations lasting a few nano-seconds on clusters containing a few thousand atoms. Previous studies on UO_2 have typically employed a periodic boundary geometry [129, 130]. The range of the calculations could be extended by using performance enhancing modifications to the model such as potential cut-off (see e.g. [131]) and symmetry rules. However, it is difficult to cut off the Coulombic interaction between ions where the sum of the forces involved does not converge with distance and the introduction of symmetry rules may distort the results. As such we will not employ such techniques here.

Since predictions for activation energies and experimentally obtained diffusion coefficients are available for oxygen migration in UO_2, it is very interesting to see what the MD technique will predict. It is well known that experimental techniques cannot always reliably predict the pre-exponential terms in the diffusion equation and studies are known to differ by orders of magnitude, depending on how the grain size was taken into account (see the discussion about the Booth model in Section 3.4.3 and literature data in e.g. references [7, 74] or [62, 63]). Predictions for diffusion coefficients derived from MD simulations are absolute and no assumptions need to be made, as the individual ranges of the ions can be investigated directly. Simulations are, however, subject to other approximations inherent in their description of forces within the Born ionic model and also simulation parameters such as the simulation size or boundary conditions.

7.1 Methodology

The MD simulation code used in this study is "penicillin" by A. Dornford-Smith [30]. "Penicillin" [1] performs MD simulations of nano clusters of ions in vacuum. The core of Penicillin consists of a Gear 5^{th} order predictor corrector algorithm [132]. The product of an MD simulation is a sequence of ion position and momentum information as a function of time.

Programs to investigate the MD output were initially developed by V. Bulatov to

[1] "Penicillin" refers to the Penicillin molecular dynamics methodology and algorithm. "penicillin" refers to the implementation written in C. There are some fundamental problems in the usability of penicillin which require an extensive rewrite of the software. Even though the way the code works has been changed significantly it still complies with the Penicillin methodology and test cases predict the same results.

study the high temperature behaviour of LaF_3 [31]. Since both the "penicillin" code and the programmes to investigate the resulting data were not originally designed to simulate large (>10Å) clusters of ions, the codes were overhauled extensively during this study. Great care was taken to ensure that the individual codes actually do what they advertise and possible pitfalls have been documented in the source code. Appendix A contains further descriptions and usage notes for future users of these codes.

During this study the main simulation code was re-written to incorporate parallel algorithms which allow "penicillin" to benefit from parallel computer systems. The first version of the parallel code operates by starting multiple processes which share memory and can therefore not be used on large scale message passing machines. However this method scales well on multi-processor Intel based machines. The second version uses a full message passing algorithm (based on the MPI 1.1 standard [133, 134]) and can run on any type of supercomputer, multi-processor server or so called "Beowulf" clusters, which are commodity machines connected by cheap Internet protocol network hardware. The power of MPI based programmes is that they can, in fact, run over a network of heterogeneous machines.

To make the simulation of clusters easier, codes were developed which facilitate the cleaving of virtual crystallites from "CASCADE" output files. For the purpose of evaluating diffusion coefficients a code was added to the tools suite by V. Bulatov which evaluates the distribution of displacements. Finally an application was developed which allows "penicillin" users to visualize molecular dynamics results on-screen.

Since the exact method of setting up an MD simulation with Penicillin has not been published before, the simulation runs will be reported in detail in this chapter.

Addional descriptions of the algorithms and methods used in this chapter may be found in Appendix A and are referenced throughout the text.

7.2 Penicillin algorithm

The molecular dynamics technique is a computer simulation method that predicts the motion of the ions in a lattice. Both the vibrational motion and the displacement of ions results from the iterative solution of Newton's equations of motion.

7.2.1 Gear 5^{th} order predictor-corrector algorithm

The method used to solve Newton's equation via a predictor (fifth order polynomial) and a corrector (the correction to the acceleration of the ions as a result of interionic forces) is formally known as the Nordsieck method. Gear [132] derived a model which allows us to correct the terms of the polynomial when the correction to the second order term is known.

Predictor

We use a Taylor expansion to predict, at time $(t_0 + \Delta t)$, the position of an atom (e.g., $r_0(t_0 + \Delta t)$), if, at time t_0, the position of a particle was $r_0(t_0)$, its velocity was $r_1(t_0)$, its acceleration was $r_2(t_0)$, etc. to the order n where $r_n(t_0) = \frac{d^n r}{dt^n}$

$$r_0(t_0 + \Delta t) = r_0(t_0) + \Delta t r_1(t_0) + 1/2\Delta t^2 r_2(t_0) + 1/6\Delta t^3 r_3(t_0) + ... \qquad (7.1)$$

Similarly we can predict expressions for velocity, acceleration and higher order terms:

$$r_1(t_0 + \Delta t) = r_1(t_0) + \Delta t r_2(t_0) + 1/2\Delta t^2 r_3(t_0) + ..., \tag{7.2}$$

$$r_2(t_0 + \Delta t) = r_2(t_0) + \Delta t r_3(t_0) + ... \tag{7.3}$$

The accuracy of these expressions is improved if more higher order terms are included. Experience suggests that, for the present purposes, we can safely terminate after the fifth order. The accuracy is further defined by the size of the time step Δt; the smaller the size of Δt, the more accurate the prediction. Of course, smaller time steps will require more iterations to perform a simulation for a fixed length of time. A typical time step in these calculations is 10^{-15} seconds, although in Penicillin this value is not necessarily fixed.

Corrector

After each predictor step the total force experienced by each ion due to its interaction with all other ions in the cluster is calculated. The acceleration of each ion is compared to the acceleration assumed in the second order term of the predictor. The terms of the predictor can now be corrected with the difference between the second order predictor and the real acceleration due to the combined force, F, using the following formula (see Gear [132], where the Nordsieck method is discussed in detail):

$$\Delta r_2 = \frac{F}{m} - r_2, \quad \text{where} \quad r_n = r_n + \Delta r_2 G_n, \quad n \in \{0, 1, 2, 3, 4, 5\} \tag{7.4}$$

where G_n are the Gear coefficients which depend on the order and type of system being solved and m is the atomic mass.

7.2.2 The total energy

Although penicillin supports many types of potential forms, only the Buckingham form was used in this study. The potential energy (in Joules) of a crystallite is therefore:

$$E_{pot} = 10^{-7} c^2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} \frac{q_i q_j}{r_{ij}} + A_{ij} \exp \frac{-r_{ij}}{\rho_{ij}} - \frac{C_{ij}}{r_{ij}^6} \tag{7.5}$$

where c is the speed of light in vacuum, n is the number of ions in the simulation, r_{ij} is the distance between ions i and j, A_{ij}, ρ_{ij} and C_{ij} are adjustable parameters of the Buckingham potential function and q_i and q_j are the ion charges in Coulombs.

Of course, since the positions of the ions change with time, the lattice energy will vary about a mean value. The balance is maintained by the fluctuating kinetic energy, which can be calculated from the simulation data.

Conservation of energy can be used to check the accuracy of the simulation. If the total energy of the system varies by more than a predetermined amount between successive time steps (typically 0.001 eV), the time step must be reduced and the iteration repeated until energy is sufficiently conserved. This time-step adjustment is a particular feature of "Penicillin". In penicillin version 5 the energy conservation rule will be applied to the average energy per ion, to ensure simulation scale independence.

7.2.3 Rescale factors

During the equilibration stages of a simulation the kinetic energy of the ions is continually being adjusted to match the set temperature. The factor applied to all ion velocities is the rescale factor, R.

The effective temperature is defined as:

$$T = \frac{2E_k}{3k_B n}, \tag{7.6}$$

where n is the number of ions in the simulation and E_k is the total kinetic energy.

The rescale factor R is calculated from the current temperature and the required temperature:

$$R = \sqrt{T_{set}/T(t)}. \tag{7.7}$$

7.3 Inter ionic potentials

The inter ionic potentials used in this study are the same potentials as were used in the previous sections. The core-shell model is removed, as Penicillin does not include a time dependent description of the core-shell separation.

In this MD study both UO_2 and U_3O_8 are studied. After the core-shell model is removed from the ions, a static simulation is performed with "CASCADE". At this point the potentials could be modified if the predicted parameters such as elastic constants, static dielectric constants and lattice parameters are significantly affected by the lack of a core-shell interaction, although this proved unnecessary here. Of course, the high frequency dielectric constants will be unity due to the lack of shell polarisation.

7.4 Constructing a Cluster

The resulting output from perfect lattice simulations performed using "CASCADE" can be used to construct input files for "penicillin". The advantage of taking the ion positions from an energy minimisation simulation is that the ions will be in their

equilibrium positions for the potential set used. Once a cluster of ions has been cut and the MD simulation is started, the surface will relax, but since the simulation will have started with sensible ion positions, the cluster should equilibrate more easily. Nevertheless, a desired feature in Penicillin would be the ability to perform energy minimisation of the cluster before starting a MD simulation. [2]

The code used to cut crystallites is called 'cas2pen' and is described in Section A.2.7. Charge neutrality is an important condition as the Coulombic interaction is dominant in ionic systems. For example, the addition of a single electron charge in the centre of a cluster significantly alters the potential energy given by equation 7.5. Cutting a crystallite with a number of symmetric planes does not, by default, give us a neutral crystallite. The UNIX command "grep -c" can be used to count the number of different ions and determine the charge of the crystallite, although penicillin will warn the user when the total charge is not neutral. If the charge is not neutral, the cutting planes need to be adjusted. When a set of planes has been found which yield a number of ions close to the target number but with a small excess charge, the remaining charge can be removed either by hand or by defining additional cutting planes which remove corners. For the purpose of reproducibility it is advisable to add additional cutting planes. Visual inspection of the cluster with 'casplot' (see Section A.2.8) can assist users of penicillin with the above tasks.

Finally the resulting ion positions are added to a penicillin file with instructions and potentials (see Section A.1.1 for an example).

[2] This feature has been added to penicillin release 5.0. The features in that release are largely the result of experiences with the code during this work.

7.5 Running penicillin

Once the input file is finished, the simulation can start. Typically the simulated cluster will not yet be in equilibrium and an equilibration run must be performed at a set temperature. After an equilibration time of a few picoseconds the rescale factor will converge to one and the cluster will be equilibrated. Typically an equilibration run will be extended to 50-100 ps to ensure that any re-adjustments of the cluster configuration have taken place.

Depending on how large the system to be studied is, we have a choice of three 'penicillin' program versions:

- The first choice is the normal single processor program for small clusters up to 1000 atoms. The basic operation is described in the appendix A.1.1
- The second program allows the full use of a dual or quad processor shared memory machine. This version of penicillin always guarantees either faster operation or the ability to simulate more atoms at the same speed. See appendix A.1.2 for details.
- The third version is the MPI version (See appendix A.1.2) which is designed to simulate larger numbers of atoms over networked computers. Performing fast simulation of small numbers of ions is possible with the third program only if interconnects between the processors are fast and have a low latency.

Section A.1.1 explains how "penicillin" is run. "Penicillin" runs can produce a multitude of output files the most important being the ".res" and ".arc" files. The ".res" file contains a snapshot of all the simulation parameters and is updated at regular intervals. The ".arc" file contains frames of ion positions with time and time

step stamps. All other output that penicillin is able to create can be derived from the ".arc" file and it is not necessary to instruct 'penicillin' to produce anything other than ".arc" output, with the exception of the "rescale" file which is useful for determining when the ions have reached a certain temperature during equilibration. A welcome addition would be to add ion momentum data to the output options but that feature is not yet supported.

7.6 Surface and bulk

Selection criteria for distinguishing between surface and bulk atoms were developed by Bulatov *et al.* [31]. The criterion used here to distinguish between surface and bulk ions in UO_2 specifically is based on counting the number of ions within a sphere of 4.83 Å surrounding an individual ion. If the number is below the selection threshold (for UO_2 this is 35 atoms) the ion is considered surface, otherwise it is bulk. Using these parameters, a bulk atom is always at least 3-4 Å removed from the surface. Since U_3O_8 has a lower density than UO_2, the threshold is set at 60 ions within a radius of 6.5Å. Table 7.1 shows the number of bulk and surface ions for the three clusters used in this study based on the parameters in this section as a selection criterion with the "arc2surf" program (See Section A.2.3).

During a simulation, especially at high temperatures, ions which are in the interior of the cluster may migrate to the surface. Effectively this causes an inaccuracy in the predicted standard deviation, s (see below), because it limits the maximum range of the ions. The validity of our selection criterion should therefore be tested by comparing the range distribution with the size of the nano cluster.

Number of surface and bulk ions at 1000K						
	Radius	Threshold	Cluster	Bulk	Surface	Cluster Radius
UO_2	4.83	35	324 Atoms	85	239	10.0
UO_2	4.83	35	3129 Atoms	1364	1765	21.7
U_3O_8	6.5	60	2750 Atoms	1106	1644	20.8

Table 7.1: Number of surface and bulk ions. Ions which have more than the threshold number of neighbours within the selection radius are bulk ions. All others are considered surface ions.

7.7 RMSD analysis

The diffusion coefficient of a type of ion is a simple function of the Mean Square Displacement (MSD) of the ions and the time interval used to obtain the MSD. This result is derived from the diffusion equation and the assumption that the concentration of diffusing ions follows a normal distribution. The diffusion equation is:

$$\frac{dc(x,t)}{dt} - D\frac{d^2c(x,t)}{dx^2} = 0, \tag{7.8}$$

where $c(x,t)$ is the time and position dependent concentration of the diffusing species and D is the diffusion coefficient. The equation and the relationship between the MSD and diffusion coefficient is therefore valid for a *continuous* concentration distribution. Since the number of atoms in our simulations is relatively small, it may be necessary to investigate this matter further. From equation 7.8 it follows that, if we exchange $c(x,t)$ for a Gaussian distribution, the variance of the Gaussian distribution progresses with time as:

$$VAR(t) = 2Dt. \tag{7.9}$$

This is a useful result, even though it is derived from the assumption that the distribution range of the ions is continuous. To test whether our ions relate to this model we can compare the Root Mean Square Distribution of our simulations (RMSD) to a Gaussian function.

The continuous distribution of radial ion displacements can be derived from the Gaussian one-dimensional case and is equal to:

$$c''(r) = \frac{nr^2}{s^3\sqrt{\pi/2}} \exp(-\frac{r^2}{2s^2}), \tag{7.10}$$

where $c''(r)$ is the radial distribution of displacements which is *not* the same as the concentration distribution of ion displacements and s is the standard deviation which is equal to $\sqrt{VAR(t)}$.

$c''(r)$ data is presented in the following results section. In essence this is the $n(r)/N$ data where $n(r)$ is the number of ions at range r, and N is the total number of ions of the type considered in the simulation.

Dividing expression 7.10 by $4\pi r^2$ yields the concentration of the distribution of displacements at diffusion length r and has the additional advantage that the fitting variable r is only present under the exponent.

We can obtain a distribution of displacements from a ".arc" file with the "arc2dist" programme (Section A.2.2) which will also perform the required classification and division to allow direct fitting to the displacement concentration equation:

$$c(r) = \frac{n\sqrt{8}}{s^3\sqrt{\pi^3}} \exp(-\frac{r^2}{2s^2}). \tag{7.11}$$

Better still, the logarithm of the classified displacement data can be taken and a linear fit can be performed to yield the s parameter.

7.7.1 Random walk

When does a discrete distribution of displacements for a migrating ion in a lattice come to resemble the continuous case represented by equation 7.10 ? It would be extremely laborious to investigate this question by running increasingly large and long MD simulations. Much more effective is the prediction of the occupancy of lattice sites visited by a random walk particle. Random walk theory was initiated by a question, inspired by the game of golf, in an advertisement in Nature [135,136]. The site occupancy predicted by random walk simulation can be compared with either the continuous result (equation 7.10) or with distribution profiles from actual MD simulations.

We can calculate the probability that an ion is in a certain position relative to the origin after n time steps. Figure 7.1 shows the range distribution of a random walk after 1, 11 and 21 steps and the distribution function (equation 7.10) fitted to it. From the figure it is clear that after 10 random jumps equation 7.10 describes the distribution reasonably well.

Converting the simulated random walk distribution data to concentration data makes it possible to compare the simulation with equation 7.11. Figure 7.2 shows equation 7.11 plotted against concentration distribution data using the same s parameters as were used in Figure 7.1. Clearly the 10 and 20 time step simulations only describe the behaviour of the data at larger displacements. This is due to the fact that the numerical error in the calculation of concentrations close to the origin is large.

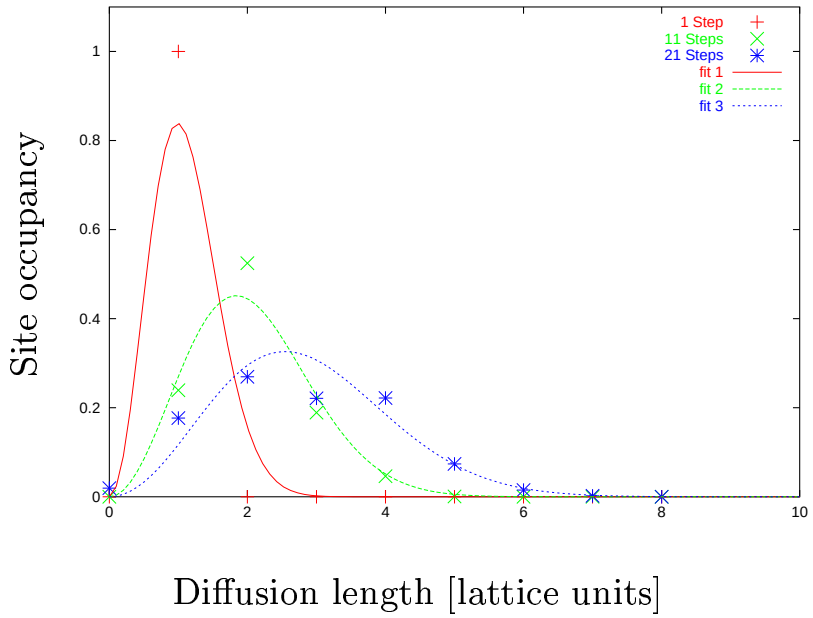

Figure 7.1: The probability distribution of a random walk particle in a cubic lattice and fitted continuous distribution functions.

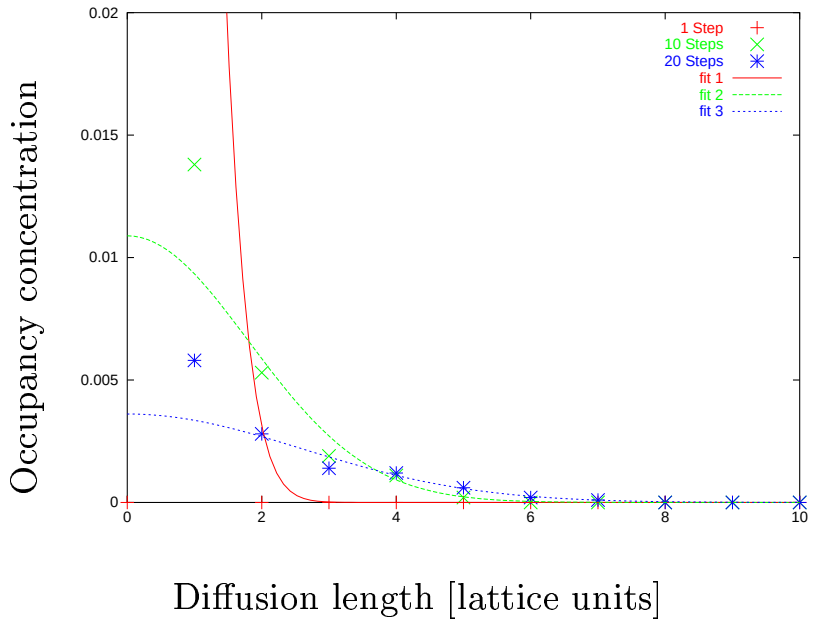

Figure 7.2: The concentration distribution of a random walk particle in a cubic lattice and fitted continuous concentration distribution functions.

The number of jumps

The jump frequency, p, of an ion in a lattice [137] is equal to :

$$p = \nu \exp(\frac{-E_M}{k_B T}), \tag{7.12}$$

where ν is the attempt frequency which is of the order of 10^{11}-10^{14} s^{-1}. For a group of n random walk particles the predicted number of jumps will therefore be a binomial distribution, which ranges from 0 to n jumps and has a maximum at $n/2$.

For simple interstitial or vacancy assisted diffusion the attempt frequency can be derived from the diffusion coefficient as in the Arrhenius diffusion equation D_0 equals νd^2:

$$D(T) = \nu d^2 \exp(\frac{-\Delta H_m}{k_B T}), \tag{7.13}$$

where d is the jump length which is of the order of the distance between the ions. ΔH_m is the Arrhenius migration enthalpy. Note that we are ignoring any geometric or configurational entropy factor here.

For oxygen vacancies in UO_{2-x} the activation energy taken from experiment [61] is 0.51 eV and the pre-exponential, D_0, equals 2.2×10^{-8} $m^2 s^{-1}$. Using equation 7.13 and assuming a jump length of 2.7Å, an attempt frequency of 3×10^{11} is predicted. Equation 7.12 therefore predicts that, on average, an oxygen vacancy in UO_{2-x} will make one jump per nano second at 1000K and about 6-7 jumps per nano second at 2000K.

Unfortunately our current computing facilities are only able to simulate an appreciable number of ions, approximately 3000 up to 5 nano second per year. However, reaching the required number of diffusion steps to make accurate predictions is only a matter of time as computer power currently doubles every 18 months [138].

7.7.2 Simple estimation of diffusion coefficient

Equation 7.12 can be used to estimate diffusion coefficients and hence the attempt frequency and the activation energy for migration in another way. The relative number of ions that have performed a migration step will be:

$$\frac{n}{N} = \nu \Delta t \exp(\frac{-E_M}{k_B T}), \tag{7.14}$$

which is a Arrhenius type equation which can be used to derive ν and E_M if we can obtain n/N from the output data.

7.8 Simulations

7.8.1 Small UO_2 cluster

The first simulation run was performed on a small 324 ion cluster, arbitrarily cut from CASCADE simulation output data. This cluster was equilibrated for 200 ps and then left to run for another 800 ps. Table 7.1 shows the number of bulk ions present in this cluster at 1000 K, just after equilibration.

Figures 7.3 and 7.4 show the distribution of bulk and surface ion displacements over the final 800 ps of the simulation at constant temperatures 1000, 1500, 2000 and 3000 K.

The bulk distribution in Figure 7.3 does not show enough ion mobility to be used for predicting diffusion coefficients. However, comparison with surface ion mobility in Figure 7.4 shows that the selection criterion for bulk and surface ions works. No bulk ions travel further than 12 Å, even at 3000 K.

Given the arbitrary nature of the choice of surface cuts in this small UO_2 cluster, Figure 7.4 shows some interesting behaviour which was also apparent in earlier tests.

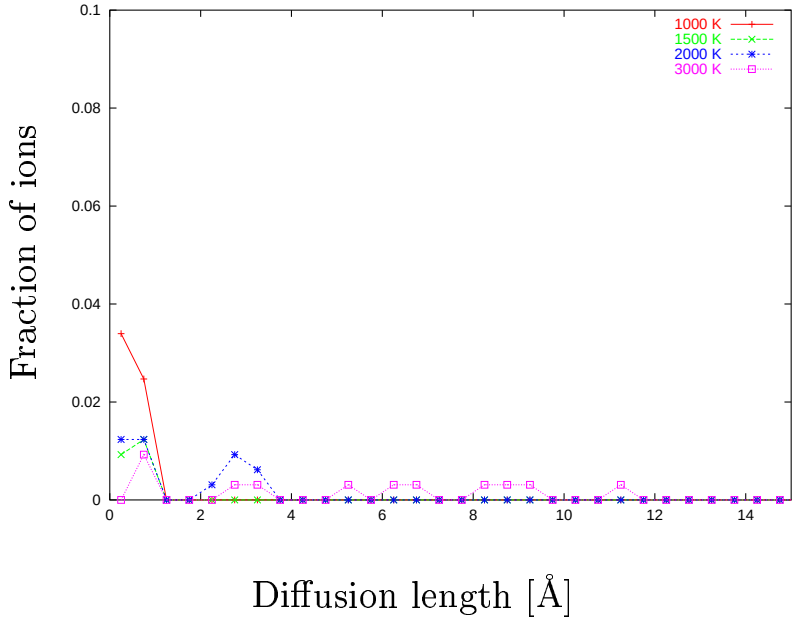

Figure 7.3: Displacement distribution of bulk oxygen ions in a 324 ion UO_2 cluster.

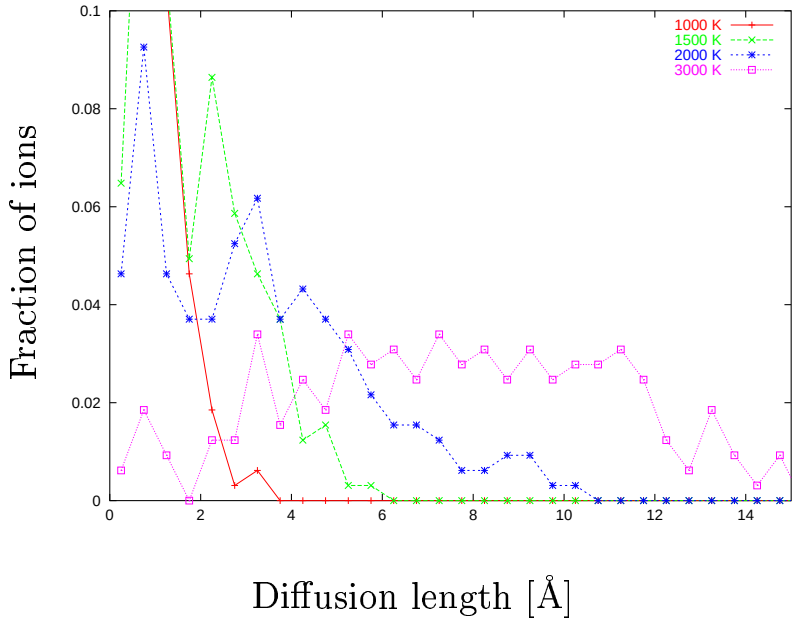

Figure 7.4: Displacement distribution of surface oxygen ions in a 324 ion UO_2 cluster.

The 1000 K (red) graph shows large elastic displacements which are seen again at 1500 K (green) graph. At 2000 K long range displacements are observed but the vibrational motion seems no larger than at 1500 K. At 3000 K the vibrational motion collapses and large scale motion is observed.

These phenomena can be explained if we assume that the original cluster did not have a stable surface and either surface melting or reconstruction is able to take place rapidly between 2000 and 3000 K. Similar behaviour was observed recently in clusters of CaF_2 [139]. This theory could be tested by cooling the cluster down after the 3000 K simulation and comparing the displacements of ions after a further 800 ps at constant temperature. Important here is that the bulk behaviour is significantly shielded from surface phenomena. However, electrostatic effects from an un-equilibrated surface can still affect bulk ions, so care must be taken when drawing conclusions from these small cluster results.

Figure 7.5 shows that there is virtually no uranium mobility within the bulk of the cluster. Even at 3000 K only vibrational motion is detected.

However, on the surface (see Figure 7.6) significant uranium mobility with diffusion lengths up to 7 Å is observed at 3000 K. It is possible that the large oxygen mobility at the surface assists the migration of uranium, but it would seem more likely that surface reconstruction is taking place. Again the arbitrary choice of surface cuts may well mean that the surface will never equilibrate in the short timescales studied here.

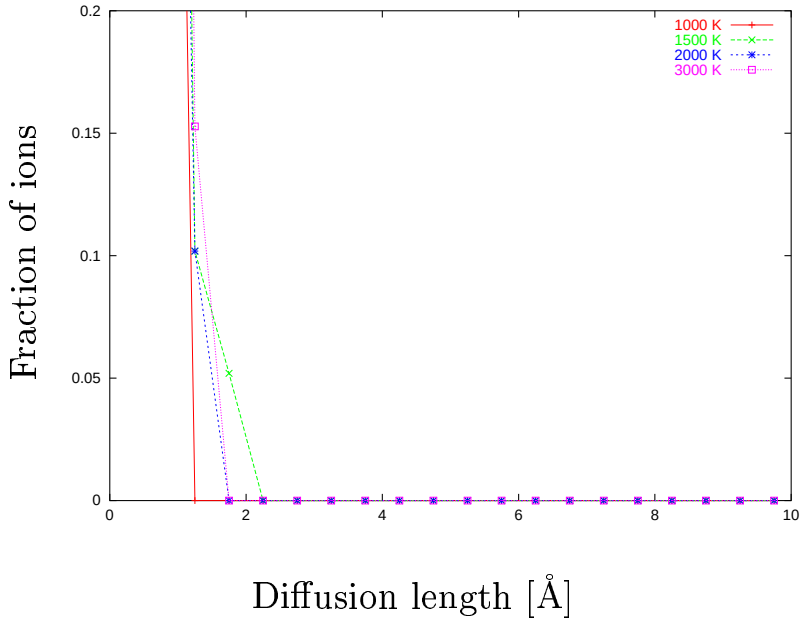

Figure 7.5: Displacement distribution of bulk uranium ions in a 324 ion UO_2 cluster. As expected, no migration takes place in the bulk.

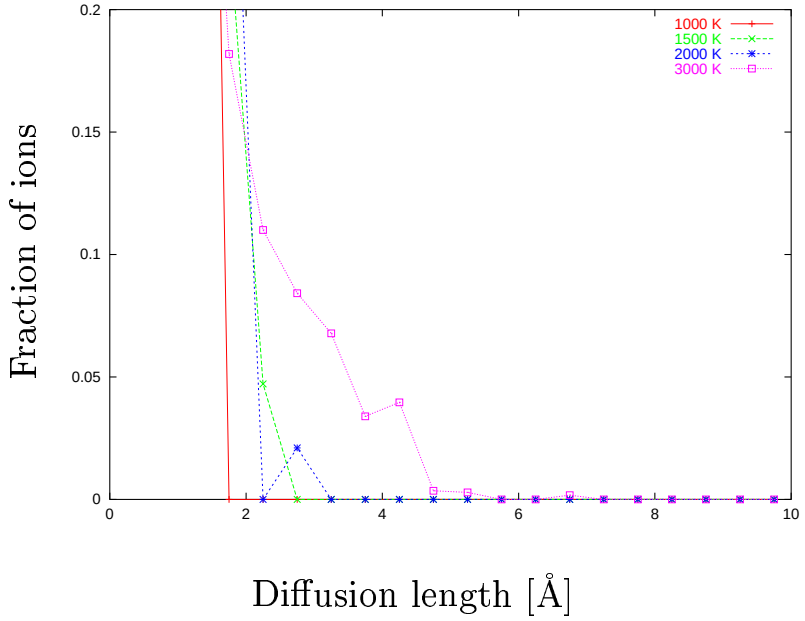

Figure 7.6: Displacement distribution of surface uranium ions in a 324 ion UO_2 cluster. The long range migration at high temperatures can be attributed to surface reconstruction.

7.8.2 Large UO_2 cluster

A crystallite of 3129 ions of UO_2 was simulated at 1500, 2000 and 3000 K for 150 ps. The crystallite used in this simulation was cut along the stable (111) plane [140], with minor corrections at the plane intersections to assure charge neutrality. For pure stoichiometric UO_2 with no surface hydroxylation the particle morphology has been predicted to be an octahedron bound by (111) surfaces [141].

Figures 7.7 and 7.8 show the displacement concentrations of the bulk and surface oxygen ions respectively. In this large cluster the significant numbers of ions with large displacements at the surface as seen in the small UO_2 cluster (Figure 7.4) are not observed. The timescale of the large cluster simulation is shorter than the 324-ion cluster but even if that was an explanation for the lack of surface instability, reconstruction would still be expected at high temperatures. Nevertheless, large diffusion lengths for oxygen ions are observed at the surface at 3000 K. Undoubtedly this can be attributed to some form of surface melting or surface sub lattice melting as a result of under coordinated surface ions.

Figure 7.9 shows uranium mobility at the surface, especially at 3000 K. Compared with the small UO_2 cluster results (Figure 7.6) there appears to be less of an increase in diffusivity when the temperature is increased.

Figure 7.10 indicates that bulk uranium exhibits even less mobility than in the 324 ion simulation (Figure 7.5). The instability of the surface in the smaller cluster influences the behaviour of uranium ions deep within the cluster. The stable surfaces of the larger cluster do not show this behaviour. This emphasises how important it is to carry out simulations with a sufficiently large cluster.

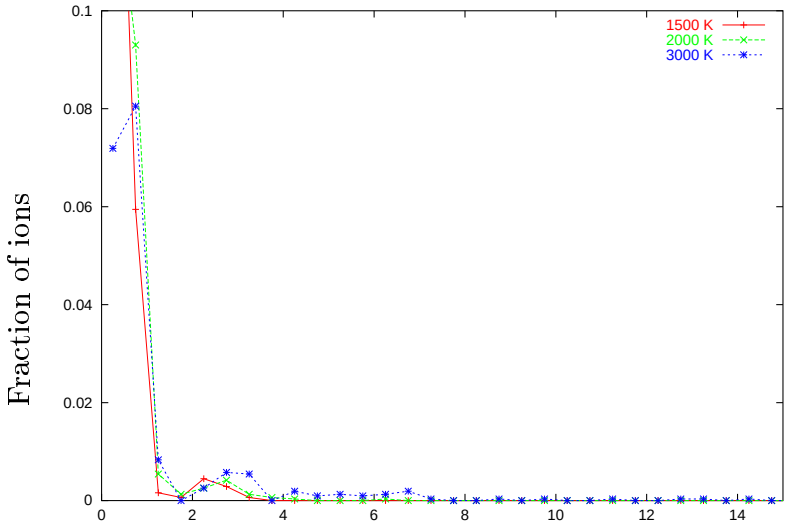

Figure 7.7: Displacement distribution of bulk oxygen ions in a 3129 ion UO_2 cluster.

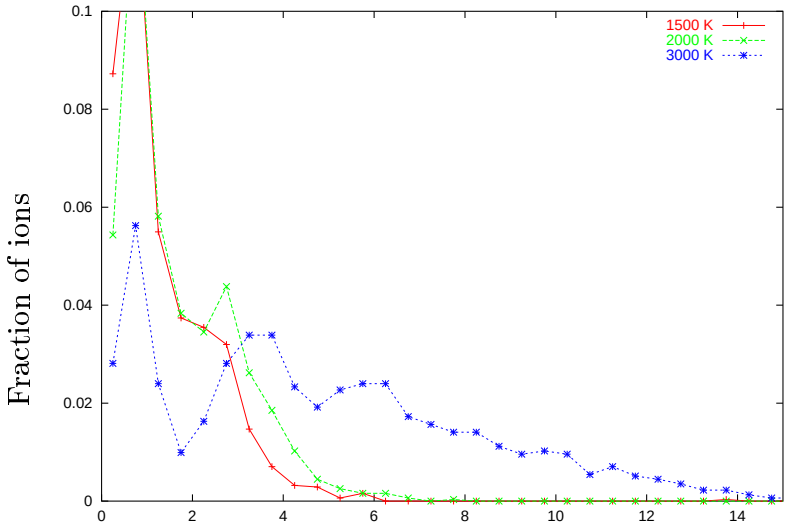

Figure 7.8: Displacement distribution of surface oxygen ions in a 3192 ion UO_2 cluster.

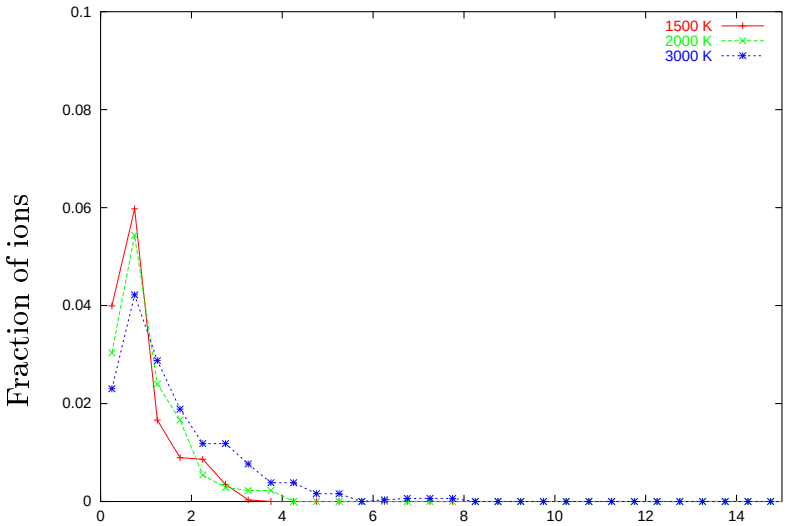

Figure 7.9: Displacement distribution of surface uranium ions in a 3192 ion UO_2 cluster.

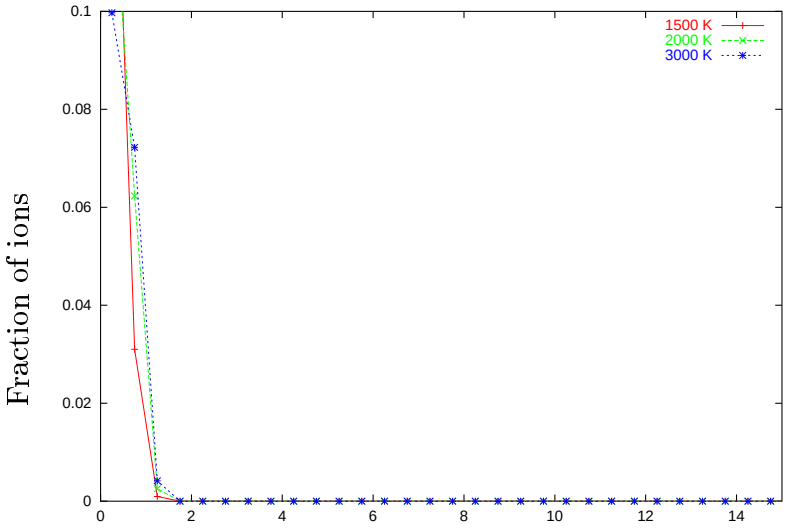

Figure 7.10: Displacement distribution of bulk uranium ions in a 3129 ion UO_2 cluster.

7.8.3 U_3O_8

The lessons learnt from the simulations on UO_2 have been applied to the simulation of U_3O_8. A cluster of 2750 ions was cut from CASCADE simulation output and care was taken to ensure that only neutral surfaces of the {0001} and {11$\bar{2}$0} families face the vacuum.

Simulations were performed at 500, 700, 1000, 1500 and 2000 K. The equilibration time for each cluster was 100 ps. Because these simulations were started on a variety of machines across the Internet, the simulation times vary. Here we have chosen to analyse 300 ps of data after the initial equilibration for consistency although some of the simulations had reached 1 ns of simulation time.

The bulk/surface selection criterion was adjusted for U_3O_8 as described in Section 7.6. Using this criterion approximately 40% of the ions are classified as bulk ions (see Table 7.1).

Oxygen behaviour

Figures 7.11 and 7.12 show the density distribution of migrating bulk and surface oxygen ions respectively. In U_3O_8 the degrees of freedom of migrating oxygen ions are much greater than in UO_2 and hence the distribution function looks more continuous. The behaviour as far as diffusion length is concerned is similar to that observed in the large UO_2 cluster. However, the simulation times are shorter for the U_3O_8 phase and we therefore expect the diffusion coefficients to be qualitatively higher than in UO_2.

Surface migration of oxygen ions in U_3O_8 shown in Figure 7.12 seems to occur much less frequently than in the large UO_2 cluster (Figure 7.8). This again indicates that the UO_2 cluster surfaces are not entirely stable, or possibly the migration takes

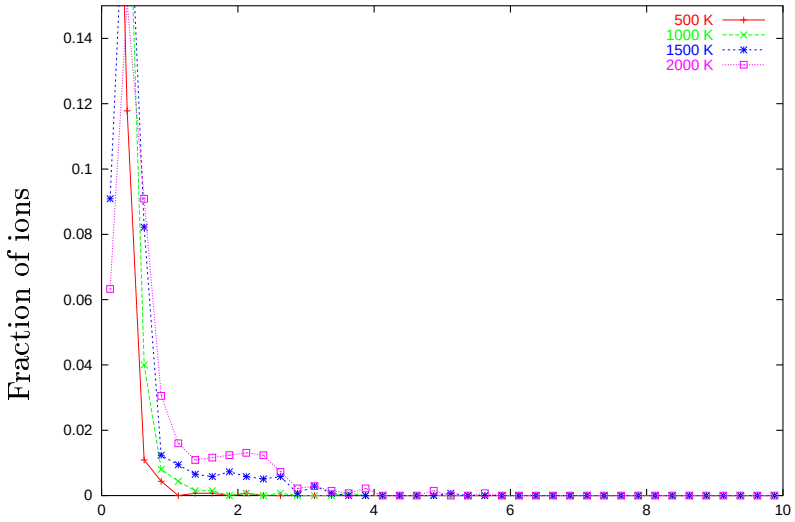

Figure 7.11: Displacement distribution of bulk oxygen ions in a 2750 ion U_3O_8 cluster.

place on the edges where the surface planes meet. The expected behaviour on surfaces of fast ion conducting systems is a reduction in the number of ions migrating due to the loss of one degree of freedom and the electrostatic surface termination which makes movement out-of-plane energetically expensive. The dipolar type of the surface plays a role here, but this has not been investigated further.

Uranium behaviour

Comparison of Figures 7.13 and 7.10 indicates that uranium shows little or no migration in this short simulation time interval. Comparison of Figure 7.14 with the UO_2 case in Figure 7.9 suggests that the surfaces in the U_3O_8 cluster are well behaved.

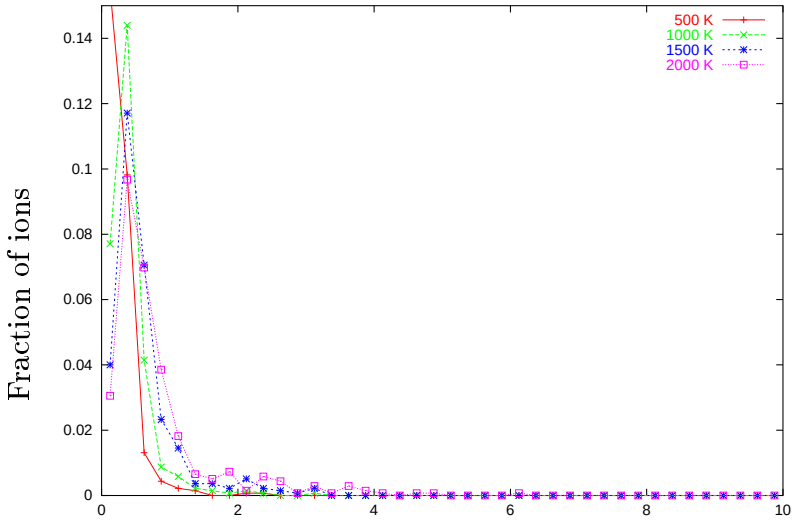

Figure 7.12: Displacement distribution of surface oxygen ions in a 2750 ion U_3O_8 cluster.

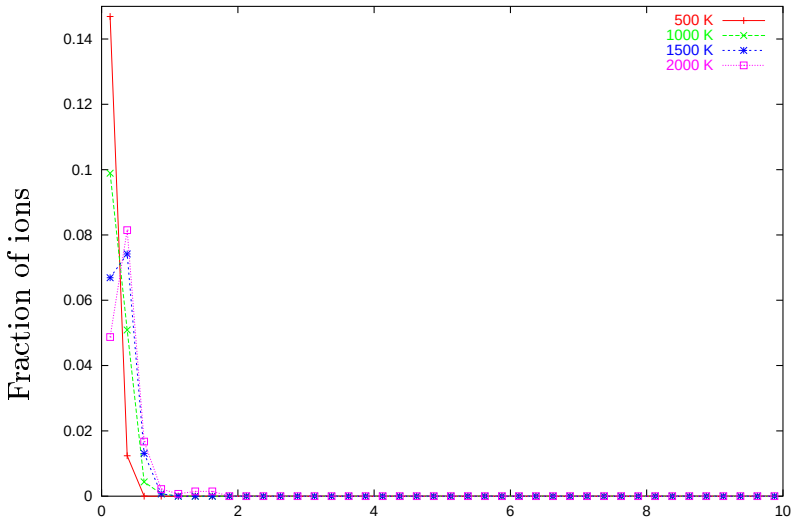

Figure 7.13: Displacement distribution of bulk uranium ions in a 2750 ion U_3O_8 cluster.

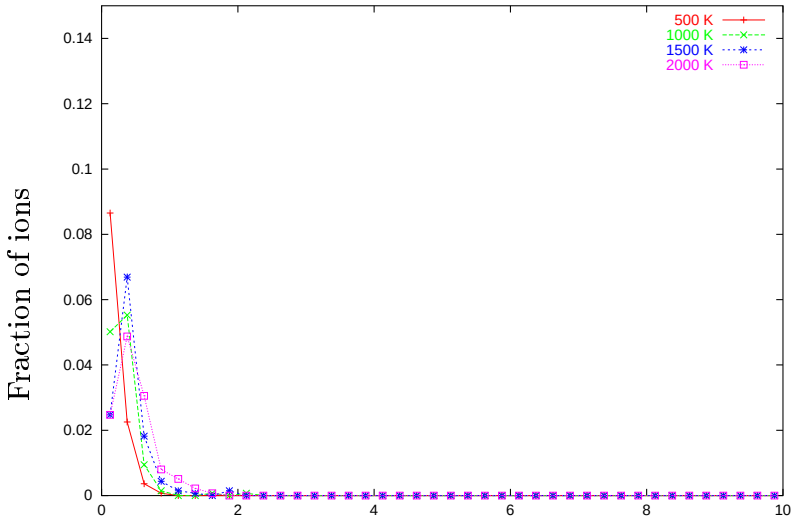

Figure 7.14: Displacement distribution of surface uranium ions in a 2750 ion U_3O_8 cluster.

Comparing different oxygen types

In the U_3O_8 phase two crystallographically distinct oxygen types are present; planar oxygen (O) and chain oxygen (OX). Figure 7.15 shows the migration behaviour of planar oxygen and Figure 7.16 shows the chain oxygen behaviour. Clearly planar oxygen migrates more easily than chain oxygen in the bulk. It is interesting to note that the chain-type oxygen Frenkel defect is predicted to have a significantly higher formation enthalpy via static simulations than the planar oxygen ion.

The distinction is less clear at the surface, although long diffusion lengths can be seen for planar oxygen when Figure 7.17 and Figure 7.18 are compared.

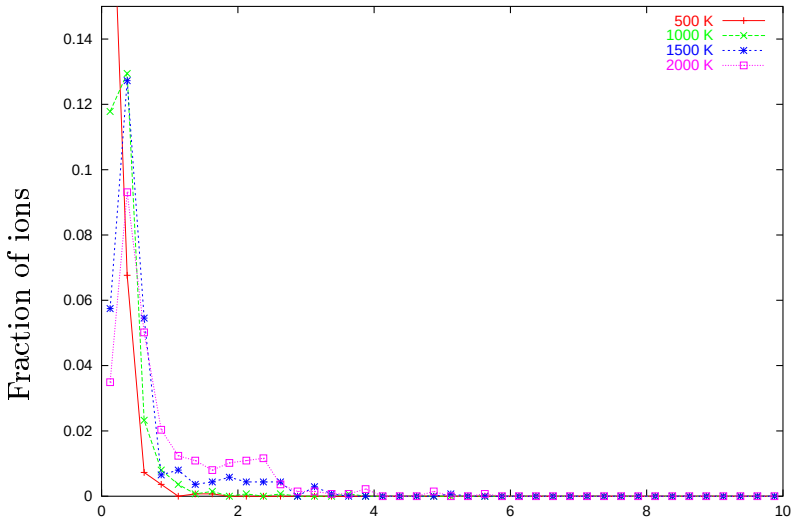

Figure 7.15: Displacement distribution of bulk planar oxygen ions in a 2750 ion U_3O_8 cluster.

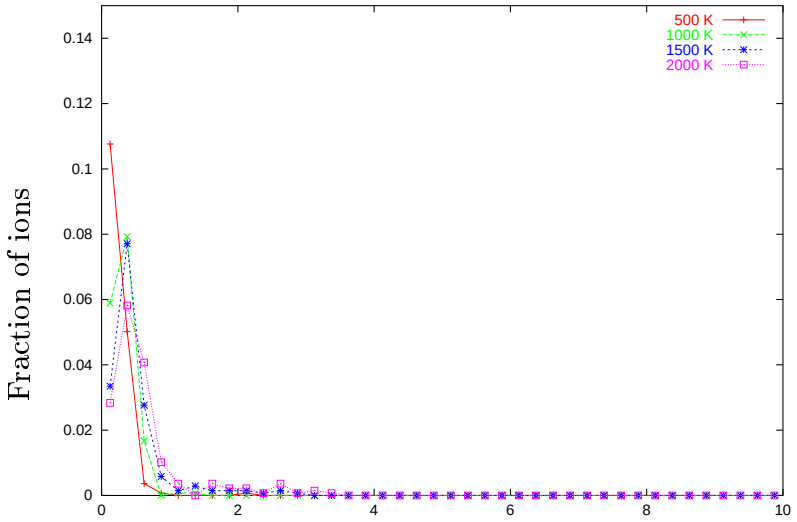

Figure 7.16: Displacement distribution of bulk chain oxygen ions in a 2750 ion U_3O_8 cluster.

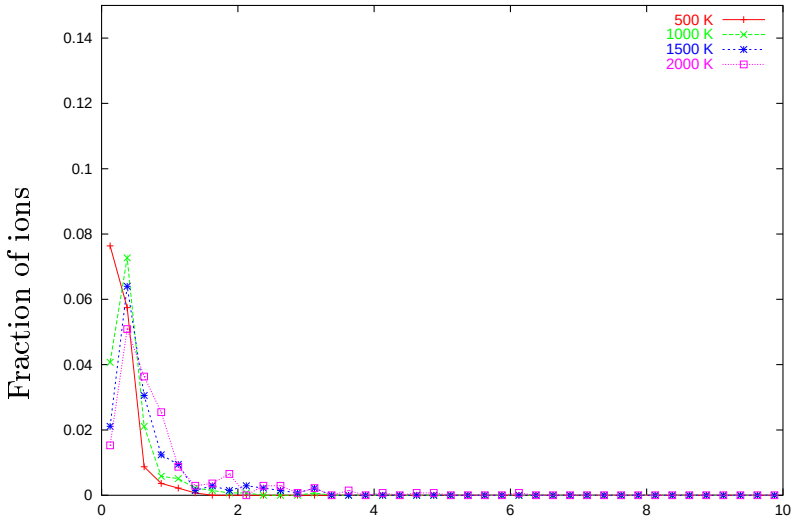

Figure 7.17: Displacement concentration of surface planar oxygen ions in a 2750 ion U_3O_8 cluster.

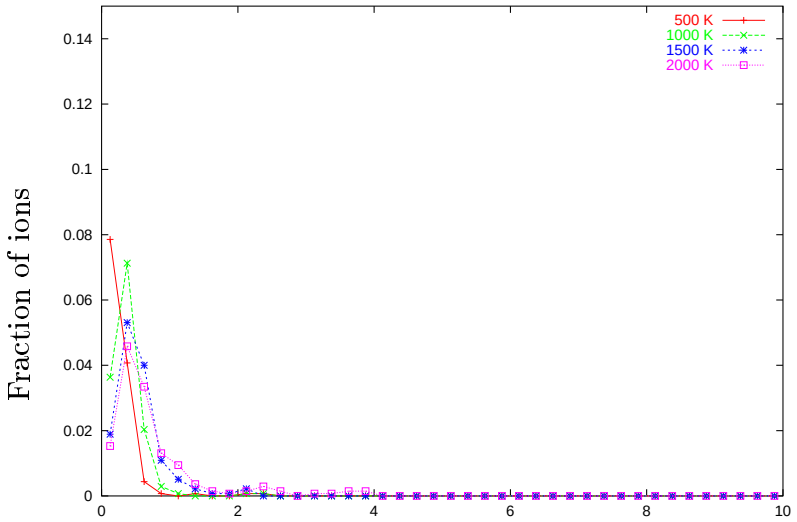

Figure 7.18: Displacement concentration of surface chain oxygen ions in a 2750 ion U_3O_8 cluster.

7.9 Diffusion Coefficients

In this section the diffusion coeficients are derived using the distribution results presented in the previous section, various processing steps and equation 7.11.

7.9.1 The small UO_2 cluster

Figure 7.19 shows the derived diffusion coefficients for the small 324 ion UO_2 cluster. Results obtained at 1000 K or lower are ignored here. The results seem to be trailing off towards lower temperatures, which is understandable, since any error or noise will result in an over estimation of the diffusion coefficient when the number of ion displacements is small. Indeed, Figure 7.3 only showed appreciable migration of oxygen ions when temparatures of 2000 K were reached.

7.9.2 The large UO_2 cluster

Figure 7.20 shows a plot of the diffusion coefficients of oxygen derived from the 3129 ion UO_2 cluster results. Here the surface migration results are clearly invalid. The bulk oxygen results show the same behaviour as the 324-ion cluster, where the diffusion coefficient seems higher than expected at lower temperatures.

Although the diffusion length of uranium is almost certainly too short to perform any reliable analysis, derived diffusion coefficients are presented in Figure 7.21. What is measured here is a widening of the distribution plots due to thermal motion of the ions. The predicted ΔH_m values do not possess any physical reality as far as diffusion phenomena are concerned.

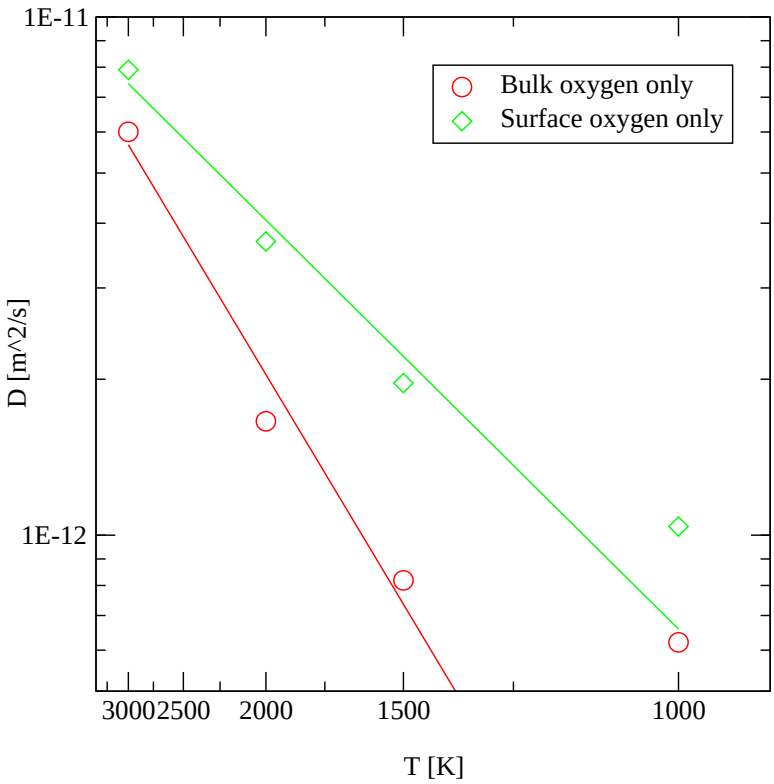

Figure 7.19: Summary of predicted oxygen diffusion coefficients in UO_2 derived from the small 324 ion UO_2 cluster. The fitted graphs correspond with an activation energy for diffusion of 0.53 and 0.31 eV.

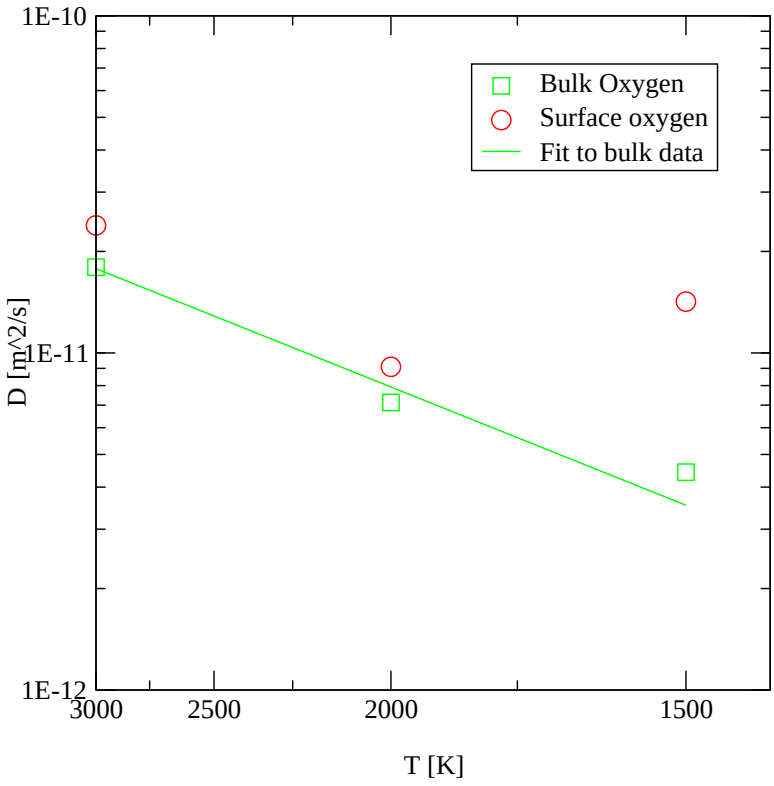

Figure 7.20: Summary of predicted oxygen diffusion coefficients in UO_2 derived from the large 3129 ion UO_2 cluster. The fitted line represents an activation energy of 0.42 eV.

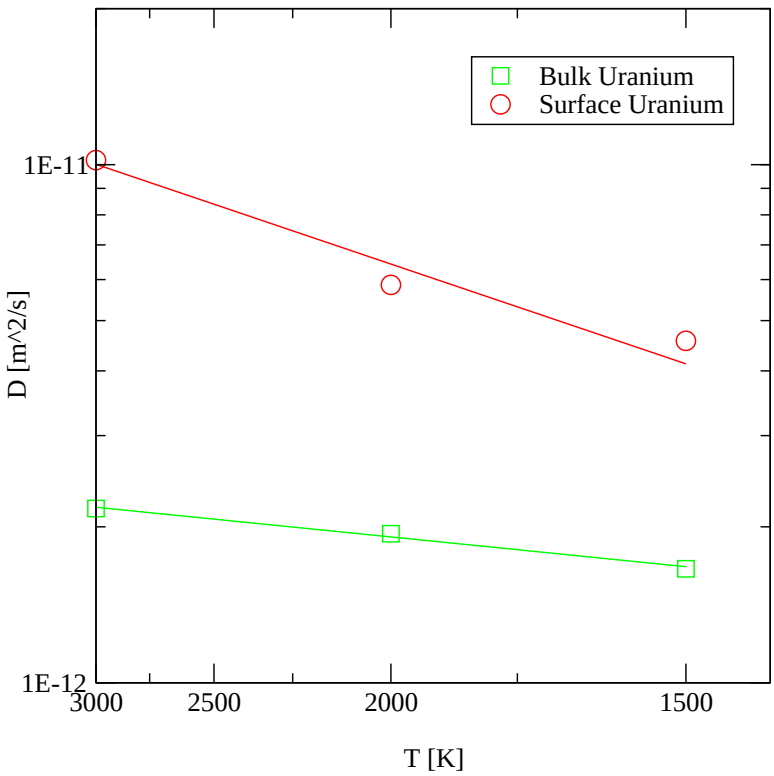

Figure 7.21: Summary of predicted uranium diffusion coefficients in UO_2 derived from the large 3129 ion UO_2 cluster. Due to the very short diffusion lengths measured for uranium over the course of this 300 ps simulation, these results have very little to do with diffusion and represent the increased variance in the position due to thermal vibration.

7.9.3 The U_3O_8 cluster

Figure 7.22 contains the diffusion coefficients derived from the U_3O_8 cluster simulation. Again the uranium results are a result of atomic vibration rather than actual migration. A very low activation energy for oxygen diffusion is derived from these results.

Since basal plane and chain oxygen ions are expected to behave slightly differently, the results for basal plane ions and chain ions have been separated in Figure 7.23. There seems to be a grouping of chain and basal oxygen, but a large amount of scatter in the simulation results distorts the picture.

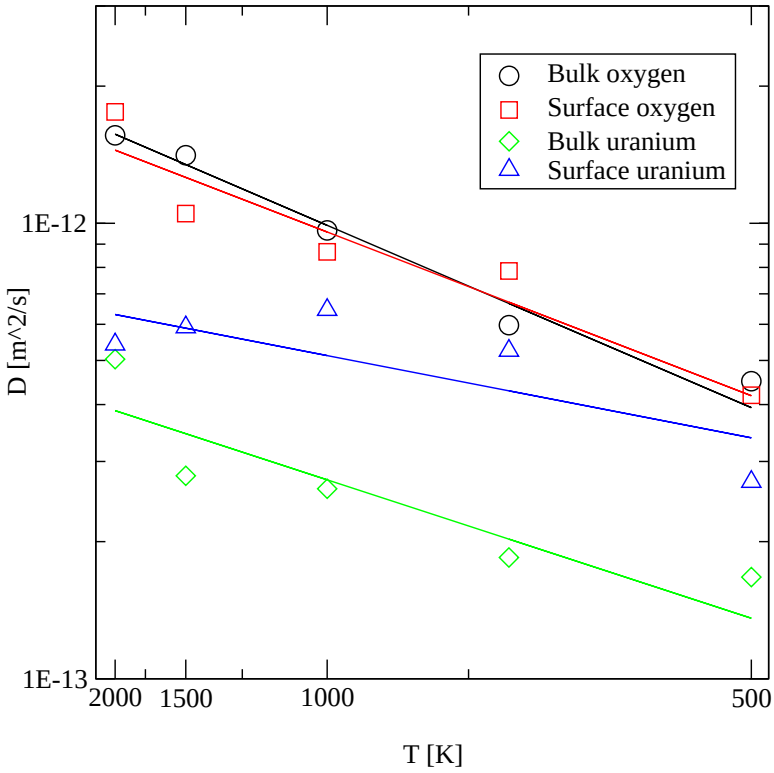

Figure 7.22: Summary of predicted oxygen and uranium diffusion coefficients in U_3O_8. Both the bulk and surface oxygen migration activation energies are lower than 0.1 eV

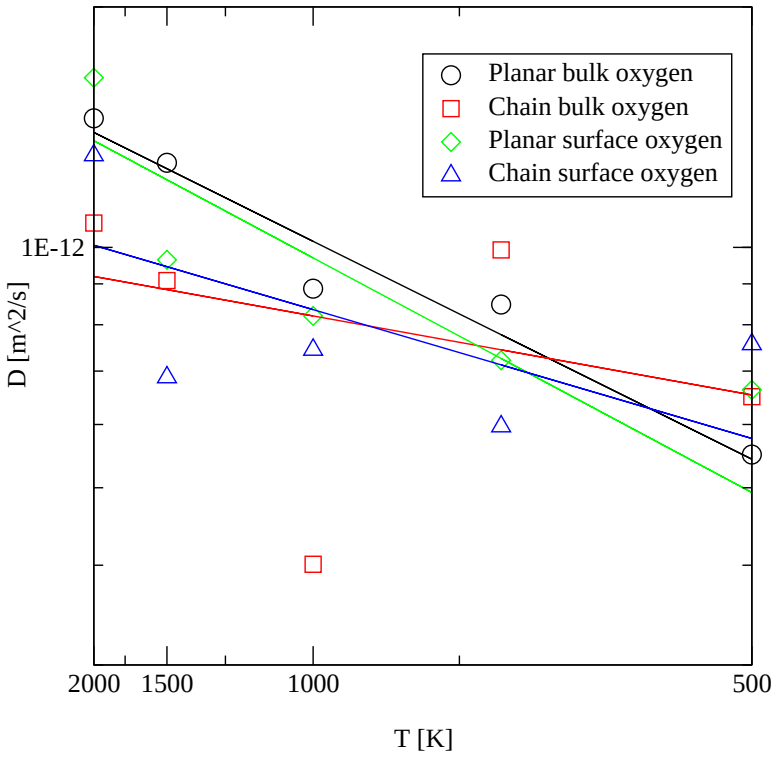

Figure 7.23: Summary of oxygen diffusion coefficients in U_3O_8. Here the behaviour of planar and chain type oxygen ions is compared. Bulk and surface ions are shown separately.

Ion	ΔH_m [eV]	D_0 [cm^2/s]
324 ion UO_2 cluster		
Bulk O	0.53	4.37 x 10^{-11}
Surface O	0.314	2.50 x 10^{-11}
3129 ion UO_2 cluster		
Bulk O	0.41	8.95 x 10^{-11}
2750 ion U_3O_8 cluster		
Bulk O	0.080	6.61x10^{-12}
Surface O	0.071	5.84x10^{-12}
Bulk planar O	0.054	5.09x10^{-12}
Bulk chain O	0.020	2.75x10^{-12}
Surface planar O	0.058	5.09x10^{-12}
Surface chain O	0.032	3.23x10^{-12}

Table 7.2: Summary of diffusion coefficients derived from MD results presented in Figures 7.19, 7.20, 7.21, 7.22 and 7.23.

Summary

The activation energies and the pre-exponentials for the Arrhenius equation derived from molecular dynamics simulations are presented in Table 7.9.3. Results for uranium have been excluded because the average diffusion lengths are too short to derive any meaning from these fitted diffusion equations.

From the table it is clear that the predictions for the pre-exponential factor are all in the 10^{-12}-10^{-10} range. From a theoretical point of view this is a least 2-4 orders of magnitude too low (see Section 7.7.1).

However, results for UO_2 are close to experimental results for UO_{2-x} [61]. This

encourages us to develop this approach and refine the data further.

(It is interesting to note that the uranium simulations also result in pre-exponentials in this range, although one could argue that the methodology should describe the increase in the amplitude with increasing temperature of an ion trapped at an equilibrium position because this motion has a pre-exponential related to the Debye frequency. The meaning of the "activation energy" is unknown within this context.)

Why is the oxygen migration activation energy lower in the larger UO_2 cluster than in the small cluster? This can be explained if we assume that the low temperature results introduce overestimations of the diffusion coefficient. Especially with a short timescale simulation the diffusion length will be short at low temperature and the derived diffusion coefficient may be influenced significantly by the thermal vibration. The estimation of errors into this methodology should give us limits as to how long simulations need to run to provide us with meaningful results.