New software for finding transition states by probing accessible, or ergodic, regions
Scott Marcus Woodley, Andrew M Walker

To cite this version:
Scott Marcus Woodley, Andrew M Walker. New software for finding transition states by probing accessible, or ergodic, regions. Molecular Simulation, Taylor Francis, 2007, 33 (15), pp.1229-1231. 10.1080/08927020701714579. hal-00515021

HAL Id: hal-00515021
https://hal.archives-ouvertes.fr/hal-00515021
Submitted on 4 Sep 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
New software for finding transition states by probing accessible, or ergodic, regions

<table>
<thead>
<tr>
<th>Journal:</th>
<th>Molecular Simulation/Journal of Experimental Nanoscience</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manuscript ID:</td>
<td>GMOS-2007-0115.R1</td>
</tr>
<tr>
<td>Journal:</td>
<td>Molecular Simulation</td>
</tr>
<tr>
<td>Date Submitted by the Author:</td>
<td>28-Sep-2007</td>
</tr>
<tr>
<td>Complete List of Authors:</td>
<td>Woodley, Scott; UCL, Chemistry</td>
</tr>
<tr>
<td>Complete List of Authors:</td>
<td>Walker, Andrew; University of Cambridge, Department of Earth Sciences</td>
</tr>
<tr>
<td>Keywords:</td>
<td>Ionic diffusion, Ergodic regions, Software, Transition states</td>
</tr>
</tbody>
</table>
New software for finding transition states by probing accessible, or ergodic, regions

S. M. WOODLEY*† and A. M. WALKER‡

†DFRL, Department of Chemistry, University College London, UK
‡Department of Earth Sciences, University of Cambridge, UK

Keywords: Ionic diffusion; Ergodic regions, Software; Transition states

Running title: Finding transition states

An automated, iterative approach to finding the lowest energy, ionic diffusion paths through a periodic structure has been developed within our new code (written in FORTRAN 77 and named Bubble). The approach is quite general in that it can be applied to find the accessible (ergodic) regions, at a chosen temperature, of a hyper-surface, which is defined across a uniform grid. We describe both our implementation within the Bubble code and its application to locating the approximate transition states for Mg interstitial diffusion in forsterite, which can then be refined using standard transition state searching [1].

In computational studies of diffusion within ionic crystals, it is necessary to be able to identify transition states, which are the least stable atomic configurations through which the system must evolve during an individual hopping event. Transition states, or saddle points, are represented by particular first-order turning points on a (3N-3) dimensional energy hyper-surface, dimensions of which are the atomic coordinates. We initially require the approximate location and configuration of the relevant turning point, defined on a three-dimensional subspace where the diffusing atom is held fixed and the other atoms relaxed so as to minimise the lattice energy. For vacancy diffusion the approximate configuration of the transition state is known (broadly, an interstitial located somewhere between two lattice sites), but for interstitial diffusion the configuration of the transition state can be less obvious. Thus we evaluate a large area of the potential energy surface, as a function of the moving interstitial magnesium ion position. More precisely, the diffusing Mg(II) interstitial was fixed on a uniform grid covering the whole symmetry irreducible portion of the unit cell, whilst the ions in the extended neighbourhood were relaxed. Once this potential energy surface has been mapped, the problem becomes one of finding suitable approximate saddle points: the grid of energy points becomes the input data for our Bubble code to search.
A flow chart of the Bubble code is shown in figure 1. For ease of explanation consider an analogous problem on a two dimensional topography, discretised over a regular n×m grid, where height represents the energy. In our example the grid spans the 3x3 supercell of the crystal. The lowest point, or global minimum (GM), in the central unit cell is located by comparing all data. Next the code finds the lowest paths to the equivalent points in the adjacent unit cells; the highest point along such a path being the transition point. Imagine the effect of pouring water onto the GM of the central unit cell. As the basin containing the GM fills, water may spill out into higher basins. Eventually, one of the surrounding equivalent lowest basins will flood. The vector from the central GM to one of the newly wet GM is the direction of diffusion. At this stage, one of the highest wet points is the transition state from one GM to the other. Adding more water will allow the transition states in the other directions to be found.

In our simulation, initially all grid points, apart from the lowest point in the central unit cell, are labelled as dry. Each time the water level is raised, the relative height of dry grid points that are nearest neighbours to wet grid points are checked; if below the water level then it is labelled as wet. Even for the smallest rise in water level, it is important to check the nearest neighbouring dry grid points to any newly wet points as flooding into neighbouring basins may occur. Moreover, if the height of the dry and wet grid point is less and higher than the previous water level, respectively, then the coordinates of the wet grid point are stored as it represents a saddle point. Each time the water level is raised only the wet grid points that define the current edge of the water pool need to be considered. To improve the efficiency of our code (CPU time requirements), the latter coordinates are stored in memory so that loops over all grid points are reduced to loops over the grid points at the water’s edge. Note that the edge of the expanding pool of water marks the boundary of the most stable ergodic region, i.e. accessible area for the diffusing ion at a defined water depth, or temperature. (By pouring water onto a different local minimum, we can effectively investigate the shape/size of higher energy ergodic regions). In the actual case of diffusion in a crystal, the system is periodic in three directions and we thus simulate an expanding bubble on a 3D grid and the code loops over grid points that define the surface of this bubble (see figure 2).
The speed of the *Bubble* code depends upon both the number of grid points (which in turn depends on the required accuracy and system size) and the step size for the water level. Upon flooding into a basin containing one of the periodic images of the GM, the previous surface of the bubble can be reloaded and a smaller step taken so to reduce the number of candidate points. (Alternatively, although less efficient, as the code requires less than a minute on a typical desktop computer, successive runs of the code with ever decreasing step size could be performed). Once the approximate transition states are located, the second order transition state search can be implemented to allow for refinement of the transition structure with motion of the diffusing ion away from the grid points. In our example, a grid of 20×40×20 points across the unit cell for forsterite (two neighbouring grid points 0.25 Å apart) was sufficient. The Bubble code can also employ the appropriate space group symmetry, so reducing the 16000 energy calculations to a more manageable 2000.

From experimental data [2], the mechanism of diffusion in forsterite is uncertain. Using *Bubble*, as well as finding the migration paths for diffusion, it was found that interstitial Mg diffusion has a higher activation barrier than O vacancy diffusion in all three crystallographic directions, (e.g. 3.13 compared to 0.72 eV in the [001] direction) so suggesting the mechanism is by vacancy diffusion [3].


Bubble: start
Read in Parameters
  e.g. step size for water level, dE
- Read in Potential (data E_i)
  for selected grid points (gps) within reduced unit cell
  E(nondefined gp in reduced cell) = arbitrary high value
Expand definition of Potential across 3x3x3 (or 5x5x5) supercell
NB crystal space group symmetry used + code checks compatibility of input data
Find Global Minimum (GM) in central unit cell
  Label GM "wet" and all other gps as "dry"
Set water level, E_{wp} = E_{GM}
Rise the water level, i.e. E_{wp} = E_{wp} + dE

Define Surface; gps b_i=1 to N
  i.e. set N=1, label GM gp as b_1
Examined all surface points?
  i.e. does N = i ?
  i = i + 1
Unmark possible saddle points
Is E(nn) dry?
Check for flooding
  Is E(nn) < [E_{wp} – dE] ?
Expand surface by including gp(nn)
  i.e. b_{N+1}=gp(nn), N=N+1
Label gp(nn) as wet
Note if b_i listed as possible saddle point
Check height of nn
  Is E(nn) < E_{wp}?
Mark as possible saddle point
Expand surface by including gp(nn)
  i.e. b_{N+1}=gp(nn), N=N+1
Label gp(nn) as wet

Effect of water level increase
  For Surface gp b_i: Examine each nearest neighbour (nn) gp in turn
    NB typically, pbc across supercell not applied, so number of nn <= 6
Are all nn gps wet or outside supercell?
Remove of any obsolete points in set b_i
  Reduce number of surface points
    i.e. remove b_i, N=N-1
Has any images of the GM become wet?
  Report water level and direction of image
  Output possible saddle points (that induced flooding)
  Checks And Reports
Report: current water level, number of recently flooded gps, number of dry (and wet) gps
And, if requested, output coordinates of accessible grid point
Bubble finished: Stop
Are all images of the GM wet?

Figure 1: Flow chart of the Bubble code: solid arrowheads of solid (broken) lines indicate direction of yes (no).
Figure 2: The growing bubble marks the accessible volume for hydrogen diffusion in forsterite crystal structure at a temperature (a) near 0K, (b) just below and (c) above the transition temperature.
New software for finding transition states by probing accessible, or ergodic, regions

S. M. WOODLEY† and A. M. WALKER‡

†DFRL, Department of Chemistry, University College London, UK
‡Department of Earth Sciences, University of Cambridge, UK

Keywords: Ionic diffusion; Ergodic regions, Software; Transition states

Running title: Finding transition states

An automated, iterative approach to finding the lowest energy, ionic diffusion paths through a periodic structure has been developed within our new code (written in FORTRAN 77 and named Bubble). The approach is quite general in that it can be applied to find, at a chosen temperature, the accessible (ergodic) regions of a hyper-surface, which is defined across a uniform grid [1]. We describe both our implementation within the Bubble code and its application to locating the approximate transition states for Mg interstitial diffusion in forsterite, which can then be refined using standard transition state searching [2].

In computational studies of diffusion within ionic crystals, it is necessary to be able to identify transition states, which are the least stable atomic configurations through which the system must evolve during an individual hopping event. Transition states, or saddle points, are represented by particular first-order turning points on a (3N-3) dimensional energy hyper-surface, dimensions of which are the atomic coordinates. We initially require the approximate location and configuration of the relevant turning point, defined on a three-dimensional subspace where the diffusing atom is held fixed and the other atoms relaxed so as to minimise the lattice energy. For vacancy diffusion the approximate configuration of the transition state is known (broadly, an interstitial located somewhere between two lattice sites), but for interstitial diffusion the configuration of the transition state can be less obvious. Thus we evaluate a large area of the potential energy surface, as a function of the moving interstitial magnesium ion position. More precisely, the diffusing Mg(II) interstitial was fixed on a uniform grid covering the whole symmetry irreducible portion of the unit cell, whilst the ions in the extended neighbourhood were relaxed. Once this potential energy surface has been mapped, the problem becomes one of finding suitable approximate saddle points: the grid of energy points becomes the input data for our Bubble code to search.
A flow chart of the Bubble code is shown in figure 1. For ease of explanation consider an analogous problem on a two dimensional topography, discretised over a regular n×m grid, where height represents the energy. In our example the grid spans the 3x3 supercell of the crystal. The lowest point, or global minimum (GM), in the central unit cell is located by comparing all data. Next the code finds the lowest paths to the equivalent points in the adjacent unit cells; the highest point along such a path being the transition point. Imagine the effect of pouring water onto the GM of the central unit cell. As the basin containing the GM fills, water may spill out into higher basins. Eventually, one of the surrounding equivalent lowest basins will flood. The vector from the central GM to one of the newly wet GM is the direction of diffusion. At this stage, one of the highest wet points is the transition state from one GM to the other. Adding more water will allow the transition states in the other directions to be found.

In our simulation, initially all grid points, apart from the lowest point in the central unit cell, are labelled as dry. Each time the water level is raised, the relative height of dry grid points that are nearest neighbours to wet grid points are checked; if below the water level then it is labelled as wet. Even for the smallest rise in water level, it is important to check the nearest neighbouring dry grid points to any newly wet points as flooding into neighbouring basins may occur. Moreover, grid points with a height that is higher than the previous water level and that have at least one neighbouring dry grid point with a height lower than the previous water level are labelled (coordinates stored) as possible key saddle points. Each time the water level is raised only the wet grid points that define the current edge of the water pool need to be considered. To improve the efficiency of our code (CPU time requirements), the latter coordinates are stored in memory so that loops over all grid points are reduced to loops over the grid points at the water’s edge. Note that the edge of the expanding pool of water marks the boundary of the most stable ergodic region, i.e. accessible area for the diffusing ion at a defined water depth, or temperature. (By pouring water onto a different local minimum, we can effectively investigate the shape/size of higher energy ergodic regions). In the actual case of diffusion in a crystal, the system is periodic in three directions and we thus simulate an expanding bubble on a 3D grid and the code loops over grid points that define the surface of this bubble (see figure 2).
The speed of the Bubble code depends upon both the number of grid points (which in turn depends on the required accuracy and system size) and the step size for the water level. Upon flooding into a basin containing one of the periodic images of the GM, the previous surface of the bubble can be reloaded and a smaller step taken so to reduce the number of candidate points. (Alternatively, although less efficient, as the code requires less than a minute on a typical desktop computer, successive runs of the code with ever decreasing step size could be performed). Once the approximate transition states are located, the second order transition state search can be implemented to allow for refinement of the transition structure with motion of the diffusing ion away from the grid points. In our example, a grid of 20×40×20 points across the unit cell for forsterite (two neighbouring grid points 0.25 Å apart) was sufficient. The Bubble code can also employ the appropriate space group symmetry, so reducing the 16000 energy calculations to a more manageable 2000.

From experimental data \cite{Chakraborty1997}, the mechanism of diffusion in forsterite is uncertain. Using Bubble, as well as finding the migration paths for diffusion, it was found that interstitial Mg diffusion has a higher activation barrier than O vacancy diffusion in all three crystallographic directions, (e.g. 3.13 compared to 0.72 eV in the [001] direction) so suggesting the mechanism is by vacancy diffusion \cite{Walker2018}.


Figure 1: Flow chart of the Bubble code: solid arrowheads of solid (broken) lines indicate direction of yes (no).
Figure 2: The growing bubble marks the accessible volume for hydrogen diffusion in forsterite crystal structure at a temperature (a) near 0K, (b) just below and (c) above the transition temperature.