UPTEC X 02 012
MAR 2002
ISSN 1401-2138

PETRA WENNERSTRÖM

A new constant pressure algorithm and periodic boundary conditions in molecular dynamics free energy calculations.

Master’s degree project

Molecular Biotechnology Programme
Uppsala University School of Engineering

UPTEC X 02 012	**Date of issue 2002-03**

Author

Petra Wennerström

Title (English)

A new constant pressure algorithm and periodic boundary conditions in molecular dynamics free energy calculations

Title (Swedish)

Abstract

Periodic boundary conditions were implemented in a already existing molecular dynamics software. In addition a new constant pressure algorithm was implemented. The new algorithm is a combination of molecular dynamics and Monte Carlo volume sampling. The results are promising but the algorithm needs more testing.

Keywords

Molecular dynamics, periodic boundary conditions, isothermal- isobaric ensemble, Monte Carlo, constant pressure.

Supervisor

Johan Åqvist
Uppsala University

Examiner

David van der Spoel
Uppsala University

Project name	Sponsors
Language **English**	Security
ISSN 1401-2138	Classification
Supplementary bibliographical information	Pages **20**

Biology Education Centre Biomedical Center Husargatan 3 Uppsala
Box 592 S-75124 Uppsala Tel +46 (0)18 4710000 Fax +46 (0)18 555217

A new constant pressure algorithm and periodic boundary conditions in molecular dynamics free energy calculations

Petra Wennerström

Populärvetenskaplig sammanfattning

Ett läkemedel består ofta av en liten organisk molekyl, en ligand, som binder till ett större protein och på så sätt ändrar proteinets funktion. Att utveckla ett sådant läkemedel är en tidskrävande och mödosam process som kräver enorma resurser. Med datorns hjälp kan man simulera hur liganden och proteinet beter sig. Detta kallas molekyldynamik och kan vara en hjälp i utvecklingsarbetet mot en färdig medicin.

Detta projekt syftar till att vidareutveckla ett befintligt molekyldynamikprogram. I gamla versionen av programmet är beräkningarna begränsade till den del av proteinet där liganden är fästad. Detta för att minska på simuleringstiden. Uppgiften är att implementera funktioner för att ta med hela proteinet i simuleringen. Simuleringstiden kommer att öka men å andra sidan kan man få fram information som annars skulle ha förbigåtts.

I arbetet ingår också att implementera en ny metod för att simulera förhållandena vid konstant tryck. De flesta experiment utförs vid konstant tryck och för att kunna jämföra simuleringar och laboratorieexperiment är det viktigt att 'försöken' har utförts under liknande betingelser. De metoder som är i allmänt bruk idag för att efterlikna konstant tryck är både svåra att implementera och har lång simuleringstid.

Den nya metoden går ut på att man då och då under simuleringen slumpmässigt ändrar volymen, dvs drar ut eller trycker ihop alla atomer och molekyler i simuleringen. De preliminära resultaten är lovande men mer jobb behöver göras.

Examensarbete 20 p i Molekylär bioteknikprogrammet

Uppsala universitet mars 2002

Contents

1 Introduction

In molecular dynamics (MD) the behavior, the movement in space, of atoms and molecules is simulated. By using this method a close up view of what really happens on the molecular level can be obtained.

MD can be applied to virtually all kinds of molecular systems, but the software used in this project is essentially made to handle large biomolecules in water solution. A typical example would be the monitoring of the binding of a small organic molecule, the ligand, to a protein. The ligand is often a lead compound to some future drug.

Other features that can be studied with molecular dynamics are conformational changes in proteins, such as protein folding and phase transitions. A number of different physical properties (energy, binding constant etc.) can be derived and it is also possible to examine processes not accessible via experiments, either because they are impossible to perform or just too dangerous.

The aim of this work is to add a number of features to an already existing molecular dynamics program. The additions include the implementation of periodic boundary conditions and a fairly untried method for sampling in the isothermal-isobaric ensemble.

2 Molecular dynamics

Molecular dynamics is simulated by integrating Newtons equation of motion

$$\mathbf{F} = m \cdot \mathbf{a} \ . \tag{1}$$

One common integration algorithm is the leap-frog version of Verlet,

$$\begin{aligned} \mathbf{v}_i \left(t + \frac{\Delta t}{2} \right) =& \mathbf{v}_i \left(t - \frac{\Delta t}{2} \right) + \frac{\mathbf{F}_i(t)}{m_i} \\ \mathbf{r}_i(t + \Delta t) =& \mathbf{r}_i + \mathbf{v}_i \left(t + \frac{\Delta t}{2} \Delta t \right), \end{aligned} \tag{2}$$

where $\mathbf{v}_i$ is the velocity of particle i, Δt is the time step, $\mathbf{r}_i$ is the position of particle i and $\mathbf{F}_i$ is the force acting on particle i.

The particles' initial positions and velocities are known as well as their masses. The force $\mathbf{F}$ is derived from a *force field* and thus the acceleration can be computed. The simulation is progressed with small steps of time, Δt. The time step must be approximately one order of magnitude smaller than the shortest motion in the system. The highest frequency is due to the stretching of bonds between hydrogen atoms and other atoms, which have a period of $\sim$10 femtoseconds [1]. It is common though, to perform simulations with rigid bonds and a time step of $\sim$1-2 fs.

The force field is a collection of parameters describing the particles in the system, for example charges and van der Waals parameters, and also bonds between different particles in terms of bond length, bonding constants, angles etc. The

force on each particle is a function of these parameters and the particle's position relative to each other. The force is calculated via the potential energy U_{pot} since

$$\mathbf{F} = -\frac{dU_{pot}(\mathbf{x})}{d\mathbf{x}} \ . \tag{3}$$

For each time step the potential energy is computed as

$$\begin{aligned} U_{pot} = & \sum_{bonds} \frac{1}{2} k_b (r - r_0)^2 + \sum_{angles} \frac{1}{2} k_\theta (\theta - \theta_0)^2 \\ & + \sum_{torsions} K_\varphi [1 + \cos(n\varphi - \delta)] \\ & + \sum_{impropers} \frac{1}{2} k_\xi (\xi - \xi_0)^2 \\ & + \sum_{atom\ pairs\ i,j} \frac{1}{4\pi\epsilon_0} \cdot \frac{q_i q_j}{r_{ij}} + \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} \ . \end{aligned} \tag{4}$$

The first four summations are the energy due to bonded interactions among the atoms. The bond-stretching force constant of the bond in question is denoted k_b, r_0 is the reference bond length, k_θ is the angle-bending force constant and θ_0 the reference angle. K_φ is the rotation force constant of the dihedral angle φ, n is the multiplicity of that angle and δ the phase shift. The out of plane bending constant of improper dihedral ξ is k_ξ and ξ_0 the reference angle.

The last sum is for the energy from non-bonded interactions. The partial charges are denoted $q_{i,j}$, r_{ij} is the distance between atom i and j, A_{ij} and B_{ij} are the Lennard–Jones parameters.

All of the above mentioned parameters constitute the force field. There are several kinds of force fields with different parameterisations, some on another form than given in Eq. (4). This variety is of course due to the fact that no force field is totally accurate, all of them are approximations. Thus all results are approximations too.

2.1 Boundaries and cut-offs

Eq. (4) takes a long time to compute. In particular the last summation over atom pairs is slow because every atom is supposed to interact with every other atom leading to a time consuming double loop. To speed up the computations a cut-off is introduced. In this way, a single atom only interacts with other atoms within a specified distance, often in the order of 10 Å. Of course this affects the simulation and cut-offs are know to introduce inaccuracies [2].

Another limiting factor of a dynamics simulation is the amount of particles taking part in the simulation. Not all particles into infinity can be explicitly modelled since the computer would run short of memory and the computation time increase. Somewhere a line has to be drawn and thus a boundary appears between vacuum and explicit particles. This is of course a highly unnatural state and there are a number of ways to cope with it.

In the *continuum model*, the atoms and molecules far away are replaced with a dielectric continuum. This might be a good idea if the surroundings are homogenous and the behaviour of the particles of the boundary does not differ too much from the ones inside the bulk.

Spherical boundary conditions

A somewhat more complex way of treating the boundary is the *spherical boundary conditions* [3, 4]. This is done by restricting the dynamics to a spherical zone and restraining the particles and molecules closest to the boundary. Restraining means that an external force, which does not originate from the force field, is exerted on to the molecules.

First of all the molecules are restrained radially to avoid evaporation from the system. Secondly, the orientation of the particles closest to the boundary is controlled. If for example the boundary cuts through water all water molecules on the surface of the sphere would be pointing with their hydrogens in the same direction, causing unwanted polarization.

The particles outside the sphere, if any, are held virtually fixed at their original positions. When calculating the potential energy these atoms only contribute to the bonded interactions, that is, only the first four summations of Eq. (4) are considered. Finally, to give a smooth transition between the fixed and the movable regions, the outermost part of the sphere is lightly restrained too.

The spherical boundary conditions may be combined with a continuum outside the sphere to handle long-range interactions.

3 Periodic boundary conditions

The ideal situation in molecular dynamics would be to simulate the interesting molecules in a bulk solution, that is, with solvent molecules filling a space extending to infinity. The continuum model does extend to infinity, but is not filled with molecules. Spherical conditions are the opposite, interactions are described with real molecules but they do not extend to infinity. *Periodic boundary conditions* might be an alternative.

The basic idea of periodic boundary conditions is not difficult to grasp. Imagine a box filled with particles. If one of the particles, during dynamics, is passing through one of the borders of the box it suddenly appears on the opposite side, as seen in Fig. 1. The system could be viewed as an infinite number of copies of the box placed side by side extending into space. The copy of a specific particle in a neighbouring box, is called it's *periodic image*.

The underlying geometry does not need to be a cubic box. All shapes that fill the space when translated are valid. There are seven possible shapes: cube, parallelepiped, hexagonal prism, truncated octahedron, rhombic dodecahedron and elongated dodecahedron [1]. Different shapes makes it possible to minimize the number of solvent molecules needed to solvate a protein. In this work we

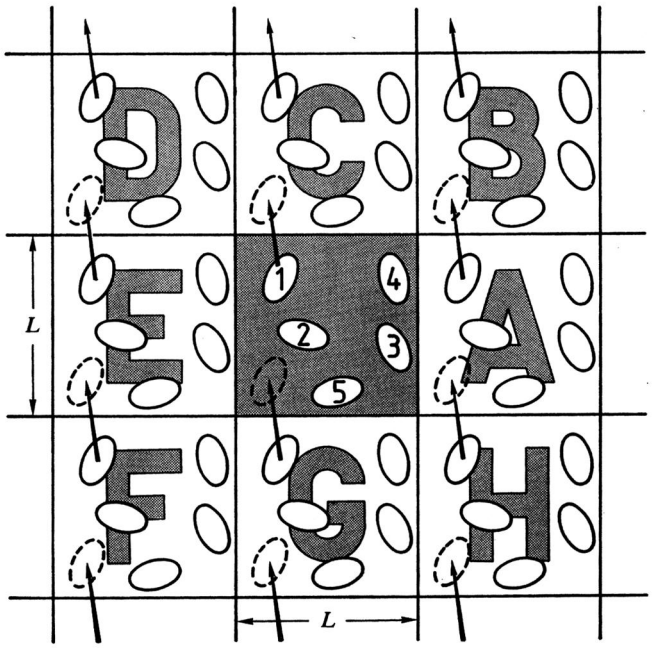


FIGURE 1: The basic idea of periodic boundary conditions in two dimensions. When particle 1 in the central box cross a boundary its periodic image enters. From [5].

have chosen the rectangular box. The corners have right angles but adjacent sides do not need to be of equal length (opposite sides do of course).

A periodic system is a system with infinite size and infinite number of particles. But all boxes are the same, with the same finite number of particles at the same relative positions. In this way only the coordinates of the particles in the central box need to be stored, other coordinates can be easily deduced from the box size.

When computing the energy of the system you focus on the central box, that is, only the energy in the central box is determined. This means that the first four summations in Eq. (4) (the bonded energy) do not run to infinity.

The standard way of using a cut-off in periodic boundary conditions is to use the *minimal image convention,* Fig. 2. Then you count a particle's interaction with the closest periodic image of all the other particles. This gives a cut-off radius of maximum $\frac{1}{2}$ the box length. In my implementation the cut-off is a user defined variable and since rectangular boxes are possible, the cut-off is maximum $\frac{1}{2}$ the shortest side of the box.

Finding the closest periodic image of a particle is not a difficult task. After computing the distance to the particle within the central box, dx, dy and dz,

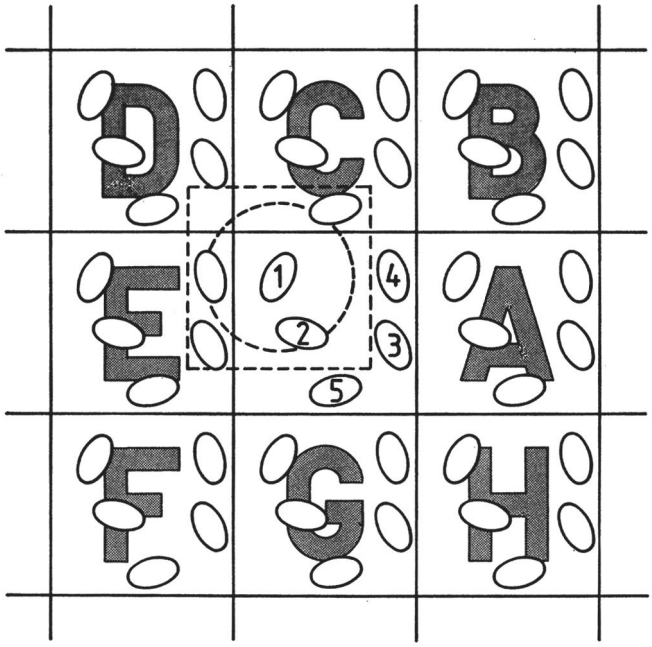


FIGURE 2: The minimal image convention. A single particle interacts only with one copy of each other particle, either it is the periodic image or the actual central box particle. From [5]

the distance to the closest image is

$$\begin{aligned} dx &= dx - l_x \cdot \texttt{ANINT}(dx/l_x) \\ dy &= dy - l_y \cdot \texttt{ANINT}(dy/l_y) \\ dz &= dz - l_z \cdot \texttt{ANINT}(dz/l_z) \ , \end{aligned} \tag{5}$$

where $l_{x,y,z}$ are the box side length in each direction. The function `ANINT(x)` returns the nearest integer to x.

Periodic vs. spherical boundary conditions

The most apparent difference between spherical and periodic systems is the runtime. When it comes to large solvated biomolecules, the periodic conditions almost inevitably gives longer runtime. To solvate a whole protein a lot more explicitly modelled solvent molecules are required than when placing a droplet over the active site. The spherical boundary conditions also have the advantage of not forcing any artificial symmetry on the system, neither is a cut-off always needed because the number of particles taking part in the simulation is often low enough to let them all interact with each other.

Periodic boundary conditions, on the other hand, do not introduce any boundary to vacuum or continuum, nor is any part of the simulated system fixed, which makes it possible to detect large conformational changes.

4 Constant pressure algorithm

Most molecular dynamics simulations are performed using the canonical ensemble, where the volume (V), the temperature (T) and the number of particles (N) are constant, while the energy, pressure etc. are variables. In some cases it is more favorable to use another ensemble, usually the isothermal-isobaric, where N,T and pressure (P) are fixed. This resembles the conditions during experiments, which facilitates comparisons between experimental and computed results.

The standard way to keep a fixed pressure is to use a so called 'pressure bath'. The volume of the system is scaled to obtain a constant pressure close to that of the target pressure, P_0, of the bath. The rate of change of pressure in a system is

$$\frac{d\mathrm{P(t)}}{d\mathrm{t}} = \frac{1}{\tau_p}(\mathrm{P}_0 - \mathrm{P(t)}) \ , \tag{6}$$

where τ_p is a coupling constant describing how strong the coupling between the bath and the system should be. P(t) is the actual pressure at time t. The volume of the system is scaled with a factor λ, defined as

$$\lambda = 1 - \kappa\frac{\delta t}{\tau_p}(\mathrm{P} - \mathrm{P}_0) \ , \tag{7}$$

where κ is the isothermal compressibility and δt is the timestep. The new positions of particle i in the the system is given by

$$\mathbf{r}_i' = \lambda^{1/3}\mathbf{r}_i \ . \tag{8}$$

Unfortunately, this method gives very large fluctuations of the pressure, often in the order of several hundred bars from the target value (se for example [2], Table II). In addition the method requires an explicit calculation of the pressure, which is computationally demanding.

To work around these obstacles I have tried an algorithm introduced 1995 by Chow and Ferguson [6] and to my knowledge not used since (the method is mentioned in [5] but not further developed). The basic concept is to run an ordinary molecular dynamics simulation combined with Monte Carlo volume-space sampling. Chow and Ferguson applied this algorithm to neon and in this work I extend it to biomolecular systems of much greater complexity.

The Monte Carlo method (MC), or Metropolis algorithm, is an energy minimization method, particularly suitable if one wants to avoid local minima. A small change is randomly applied to the system and the energy of the new state is calculated. If the new energy is lower than the previous, the state is accepted. If the new energy is higher, the state is accepted with a certain probability that depends on the size of the change of energy. This means that unfavorable steps might be accepted, giving the possibility for the algorithm to step out of a local minimum.

A straight forward application of the Metropolis algorithm is the one where particles are moved. A new state is given by randomly moving a particle. In

this work though, we focus on isothermal-isobaric Monte Carlo where a change of state means a change of volume.

The novelty of this is the combination with standard molecular dynamics. During the dynamics simulation changes of the volume are made according to the Metropolis algorithm.

A change in the volume is chosen randomly

$$\Delta V = n_{rand} \cdot \Delta V_{max} \ , \tag{9}$$

where n_{rand} is a random number between -1 and 1 and ΔV_{max} is the maximum allowed volume displacement in one move. The new volume is defined as

$$V' = V + \Delta V \ , \tag{10}$$

prime indicating the new configuration. The coordinates are then changed, the system is contracted or expanded. The scaling factor for the side length of the box, $l_{x,y,z}$ is

$$l_i' = l_i \sqrt[3]{\frac{V'}{V}} \ . \tag{11}$$

The proportions of the box are maintained, meaning that a rectangular box stays rectangular. The coordinates, $r_{x,y,z}$ of each molecules center of mass, are scaled according to

$$r_i' = (r_i - c_i)\frac{l_i'}{l_i} + c_i \ , \tag{12}$$

where c_i is the coordinate of the centre of the box. This variable is included to handle the case when the box center does not coincide with the origin of the coordinate system. The contraction or expansion is in terms of molecules, not atoms, which means that all intramolecular distances are kept fixed.

After the new configuration has been set, the potential is recalculated. Only the non-bonded interactions need to be taken in to account because the interior of molecules are not changed. The Metropolis sampling equation is

$$\Delta W = (U_{pot}' - U_{pot}) + P_0(V' - V) \ , \tag{13}$$

where P_0 is the target pressure. The new configuration is accepted with probability

$$P(\Delta V) = \begin{cases} e^{-\frac{\Delta W}{kT}} & \Delta W > 0 \\ 1 & \Delta W \leq 0 \end{cases} \ . \tag{14}$$

If ΔW is zero or negative the move is always accepted. Otherwise a new random number, $n \in [0, 1]$, is generated and the configuration is accepted if $n \leq e^{-\frac{\Delta W}{kt}}$. If a configuration is rejected, all coordinates are restored to the state before the trial move.

One important feature of Monte Carlo simulations is the acceptance ratio *number of accepted changes/total number of changes*. A ratio of ∼0.40-0.50 is preferable [5], and it is adjusted with ΔV_{max}, the larger ΔV_{max} the smaller ratio and vice versa.

One more thing worth noticing, is that the Metropolis algorithm by itself is time independent. Time as such does not exist in a Monte Carlo simulation. In our case the essence of time is solely contributed by the MD-algorithm.

5 The program Q

In 1998, a new molecular dynamics program called Q [7], was released by the Åqvist group at the Departement of Cell and Molecular Biology at Uppsala University. The program uses the mode of integration of Eq. (2) and is compatible with most force fields as long as they have the form showed in Eq. (4). Q is designed to perform free energy perturbation simulations (FEP) (se section 6.1, empirical valence bond calculations(EVB) of reaction free energies and linear interaction energy calculations (LIE) of receptor-ligand interactions. (These methods are not described closer in this work, for a brief description see [8].)

Q is designed for simulating protein molecules solvated in water and uses the spherical boundary conditions. The interesting part of the protein, the active site, is covered by a droplet of explicit water molecules. The droplet often coincides with the dynamics sphere. The part of the protein outside the sphere is held fixed.

5.1 Structure of the program

Q is actually not one program but four;

Qdyn The main dynamics program with the implementation of the above described algorithms.

Qprep Prepares the input to Qdyn with coordinates from files in PDB-format and force field parameters.

Qfep Program for analysis of the result from Qdyn. Carries out FEP and EVB calculations.

Qcalc For calculation of average structure, coordinate deviation etc.

The package is in its whole written in FORTRAN90 and is available for almost any platform. All programs are made up of modules, a computer structure in FORTRAN containing a number of variables, parameters and functions or subroutines operating on these. To some extent the module structure resembles that of a class in C++, but it is not possible to declare instances of a module. Instead the module is included in a new piece of code to avoid doubling of code.

The modules in the Q package are to some extent free-standing from the four programs. There are modules that are included in more than one program but also those that are program specific.

This work deals with the two programs Qdyn and Qprep. Qfep is used in some of the benchmarks but not altered and Qcalc is not used at all.

A thorough description of the details of the code has no place in this report. Those interested are referred to the Q website [9] where variable lists and short descriptions of functions and subroutines are available for the modules used in this work.

5.2 Units

Molecular dynamics is performed in the world of molecules and atoms. To get easily readable numbers without too many digits and exponents sometimes other units than the SI-units are used. Table I shows a list of the units used in Q and their conversion to SI-units.

Quantity	SI-unit	Q-unit	Conversion
energy	J/mol	kcal/mol	1 kcal/mol= $4.1868 \cdot 10^3$ J/mol
distance	m	Å	1 Å= $10^{-10}m$
temperature	K	K	–
charge	C	e	1 e= $1.6021773 \cdot 10^{-19}$C
time	s	$\text{Å}(\frac{\text{mol·u}}{\text{kcal}})^{\frac{1}{2}}$	1 $\text{Å}(\frac{\text{mol·u}}{\text{kcal}})^{\frac{1}{2}} = 4.88718612 \cdot 10^{-14}$
mass	kg	u	1 u= $1.66054 \cdot 10^{-27}$ kg
pressure	Pa	$\frac{\text{kcal}}{\text{mol Å}^3}$	1 $\frac{\text{kcal}}{\text{mol Å}^3} = 6.95235 \cdot 10^9$Pa

TABLE I: The internal units used in Q

For example the molar gas constant R is 8.31451 J/mol K but in the above 'molecular units' R=0.001986 kcal/mol K.

6 Results and discussion

In this section three kinds of tests of the periodic boundary implementation in Q are presented. The first is the rather simple calculation of the hydration energy of sodium. The second is the somewhat more complex and time consuming modelling of a proton transfer reaction. The last is the determination of the density of water performed with the new constant pressure algorithm. Results are compared with previous published computational results and experimental values.

6.1 Solvation energy of sodium

The solvation energy of sodium is computed by calculating the difference in free energy between an 'uncharged' and a 'charged' sodium atom in water solution.

The difference in free energy, ΔG, between two states, A and B, is related to the difference in potential energy, U_A and U_B, as

$$\Delta G = -k_B T \ln\langle e^{-(U_B - U_A)/k_B T}\rangle_A \ . \tag{15}$$

Due to the sampling of phase space Eq. (15) does not hold if the difference in potential energy is too large, typically $|U_B - U_A| \gg k_B T$. Under such circumstances intermediate states are created by linear combinations of state A and B. The potential energy of each intermediate state is described by

$$U(\lambda) = (1-\lambda)U_A - \lambda U_B \ , \tag{16}$$

where $\lambda \in [0,1]$. When changing from state A to B, λ is changed from 0 to 1. The overall free energy, ΔG, is obtained by summing the differences from all intermediate states

$$\Delta \mathrm{G} = \sum_{i=0}^{n-1} -\mathrm{k_B T} \ln \langle e^{-(\mathrm{U}_{i+1} - \mathrm{U}_i)/\mathrm{k_B T}} \rangle_i \ , \tag{17}$$

where n is the total number of states. The result is independent of the path chosen between A and B, as well as it is possible to chose a path over non-physical states.

This method is called free energy perturbation (FEP) and is used here to calculate the hydration free energy of Na^+.

Two separate systems with different sizes of the periodic box were set up. One with a side length of 21 Å, 227 water molecules, and one with 34.15 Å, 1329 water molecules. In both cases a cut-off of 10 Å was used both concerning interactions between water molecules and ion-water interactions.

The water model Single Point Charge, *SPC* [10], was used. In this model the hydrogens do not contribute to the Lennard–Jones potential, that is, the A_{ij} and B_{ij} in the last summation in Eq. (4) are zero if any of the interacting particles are a water hydrogen.

When calculating hydration energies one must take into account a correction due to the (infinite) surrounding medium. This correction can be made with the Born formula [11] (in molecular units)

$$\Delta G_{Born} = -332 \frac{q^2}{2 r_{Born}} \left(1 - \frac{1}{\epsilon} \right) \ , \tag{18}$$

where q is the charge of the ion and ϵ is the dielectric constant of the surrounding medium, in this case 80 for water. The variable r_{Born} is the radius of the cavity in the medium where the ion is, in this case the same as the ion-water cut-off. (Imagine that the ion only 'feels' the molecules inside the cut-off, outside it only 'feels' a dielectric medium.) In this example, the Born correction is equal to 16.3925 kcal/mol.

In Table II, the results with and without the Born correction are displayed. The forward and backward column indicates the solvation energy computed while charging (forward) and discharging (backward) the ion. They should give the same results. For reference, a value [12] obtained with spherical boundary conditions is shown too.

The experimental value for the solvation of a sodium ion is -98.2 kcal/mol and at first it seems like the calculations have failed since they haven't reproduced this value. There is also a difference in results between the two boxes. Furthermore, the result from the spherical system is in much better agreement with experimental data. Why is that?

The water-water cut-off radius

When an ion is solvated, the hydration free energy, ΔG, is negative, that is, the solvated state is more energetically favourable than the unsolvated state. When

		box side 21 Å	box side 34.15 Å
without Born correction	Forward	-89.749	-92.908
	Backwards	-89.883	-92.801
	Average	-89.816	-92.8545
with Born correction		-106.2085	-109.247
Spherical boundary		-99.7±0.3	

TABLE II: The hydration free energy in kcal/mol of a sodium ion with and without Born correction computed in two different box sizes. The value obtained with spherical boundary is from [12].

computing the interactions between water molecules with the aid of a cut-off, a single water molecule does not 'feel' all other water molecules interacting with the ion because they are out of range, see Fig. 3. This leads to the situation where each water molecule solvates the ion too much, that is, too frequently points towards the ion. This in turn gives too big a contribution to the solvation energy and ΔG becomes too negative.

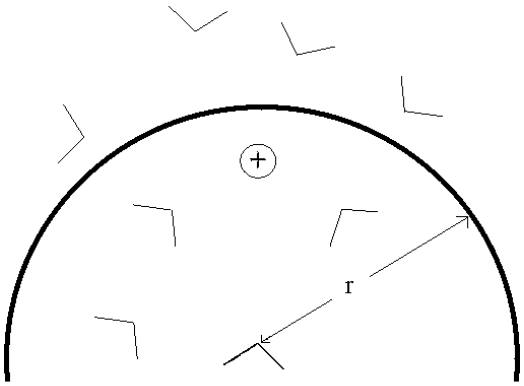


FIGURE 3: The water-water cut-off effect. The water molecule in question does not feel the interactions of all other water molecules solvating the ion.

The value obtained with spherical boundary from Table II were computed without any water-water cut-off, all molecules in the sphere interacted with each other. When a cut-off of 10 Å was introduced, ΔG fell to -108.0±0.1 kcal/mol. On the other hand, when the water-water cut-off criterion in a periodic system was altered so that all water molecules within the ion-water cut-off interacted, ΔG rose by about 5 kcal/mol [12].

Also the ion-water cut-off has an impact on the hydration energy. The water molecules outside the cut-off do not interact with the ion. However, they do interact, and adjust to, the structured water molecules within range of the ion, which leads to a decrease of ΔG.

The PBC and Born correction problem

The Born correction takes into account the interactions between the medium inside and outside of the Born sphere (the cut-off ball with the ion in the middle) For spherical systems this is a rather trivial task because outside the computational sphere there is really nothing or possibly a continuum. But in PBC the system structure is not spherical but cubic (or other angular shape, see section 3, page 4) allowing explicit water molecules outside the Born sphere.

In this case the interaction between inside and outside of the Born sphere are counted twice, once in the Born correction and once in the pairwise energy calculation of the interactions between water molecules. Thus solvation energy becomes too large and ΔG more negative.

The PBC system size effect

Water molecules adjust their orientation both to the positive ion in the middle of the box and to other water molecules. If the periodic box is small, that is, the ions in different boxes are close together, there might be some disturbances. The water molecules close to the border are in steric conflict, adjust to the ion or to the water molecules in the neighbouring box? (see Fig. 4). This makes the water molecules point less frequently towards the ion, thus leading to a higher ΔG. In spherical systems this never happens because there is only one ion present.

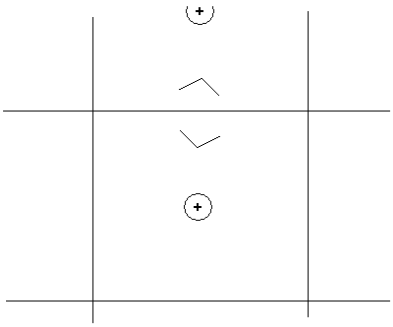


FIGURE 4: The size effect on solvation energy.

As mentioned above, water molecules outside the ion-water cut-off give a negative contribution to ΔG. In the smaller box there are less water molecules outside the cut-off which also makes ΔG rise.

The three above effects gives the results shown in Table II. The two first decrease the energy and affect both of the trial systems. In the case of the smaller system the deviation is to some extent compensated for by the small box size.

These results show that, to some extent, for this kind problem, the calculation of the hydration energy of an ion, the spherical boundary conditions are

somewhat more suitable. The simulations made here are thus not intended for determination of some physical constants, but to show that the program works correctly and gives the expected results.

6.2 The free energy of proton transfer

The first step in the reaction mechanism of low-molecular weight protein tyrosine phosphatase (low M_r PTP) was modelled. Protein tyrosine phosphorylation plays a major role in cell signaling and PTPs catalyses the dephosphorylation of phosphotyrosines. Low M_r PTPs are cytosolic enzymes of 18 kD with 140-180 residues. Their natural substrates are unknown but they dephosphorylates a wide range of phosphorylated proteins and oligopeptides *in vitro*.

The first step in the dephosphorylation mechanism is the transfer of a proton from Cys12 in the active site to the phosphate group of the substrate, see Fig. 5.

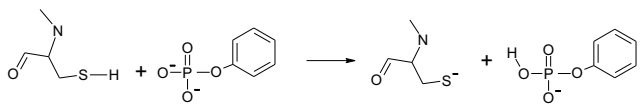


FIGURE 5: The first step of dephosphorylation in low M_r PTP.

The reaction was modelled with the empirical valence bond method (EVB), a combination of molecular mechanics and quantum mechanics. The breaking and making of bonds are modelled quantum mechanics and the rest of the system with a classical force field.

To obtain a correct EVB modell, it is calibrated against a corresponding reference reaction in aqueous solution for which experimental results are available. Thus the simulations are done twice.

The simulations were done as FEP-simulations between two states, Φ_1 with the hydrogen bound to the sulfur and Φ_2 with the hydrogen bound to one of the phosphate oxygens. The result is a free energy profile shown in Fig. 6. The activation free energy, the free energy barrier, is 4.0 kcal/mol. The reaction free energy, the difference in free energy between Φ_1 and Φ_2, is 2.3 kcal/mol.

The same reaction has be modelled in Q with spherical boundary conditions [13]. The activation free energy was reported $\sim$4 kcal/mol and the the reaction energy 1.5$\pm$1 kcal/mol.

The major difference between the simulations in spherical and periodic boundary conditions is runtime. To completely solvate the protein a box with side length 55 Å and around 4500 water molecules were needed. Of course this affects runtime. The FEP-simulation was done with a total of 51 different λ:s each requiring approximately 1 hour of runtime (on Compac Alpha XP1000), summing up to little more than 2 days for a simulation. This was done twice, once with and once without the enzyme.

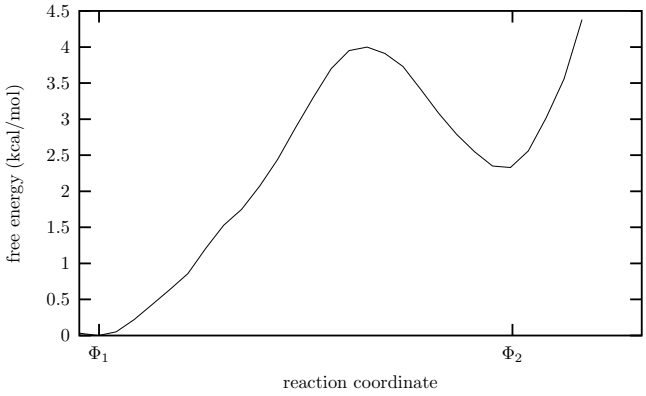


FIGURE 6: The free energy profile of the proton transfer in low M_r PTP.

With spherical conditions on the other hand, each λ require only 8 minutes, a total of 7 hours. Thus, EVB is normally done with spherical conditions, in fact this is the first time a EVB-simulation has been performed under periodic conditions to our knowledge.

6.3 Density of water

The density of water was determined with the new constant pressure algorithm. Two kinds of water models were used, SPC, described above, and TIP3P [14]. In TIP3P, all atoms in the water molecule are modelled explicitly. All atoms contribute to the Lennard–Jones potential.

Simulations were done on three system sizes, 125, 216 and 820 water molecules with cut-offs of 7.5, 9.0 and 12.0 Å. The temperature was set to 300 K and the pressure to 1 bar. Simulations were performed with $5 \cdot 10^5$ time steps of 2 fs giving a total of 1.0 ns. Averages were calculated from the last 0.9 ns considering the first 0.1 ns as equilibration. Starting configurations were taken from systems equilibrated in the canonical ensemble (constant N, V and T). Volume changes were done every 20:th time step giving 25000 trial volume configurations. The maximal volume displacement, ΔV_{max}, varies with box size and is shown in Table III together with the acceptance ratio and the results.

As the observant reader might note, the cut-off for the smallest systems (N=125) and the medium system (N=216) with TIP3P are smaller than one might expect. This is due to the fact that at some point during the simulation the box shrunk so much that the criterion $r_c < \frac{1}{2}l$ did not hold any more and the simulation had to be done again with a smaller cut-off.

From the simulations the average volume, V, of the system was deduced. This value was then converted to density by

$$\rho = 29.9163 \cdot N/V \tag{19}$$

where ρ is the density in g/cm^3 and N is the number of water molecules.

Model	N	r_c (Å)	ρ (g/cm^3)	U_{pot} (kcal/mol)	ΔV_{max}	ratio
SPC	125	7.4	0.959 (0.0268)	-9.522 (0.142)	160	0.434
	216	9.0	0.970 (0.0214)	-9.498 (0.104)	230	0.460
	820	9.0	1.042 (0.0144)	-10.141 (0.0538)	400	0.396
		12.0	1.050 (0.0136)	-10.197 (0.0535)	400	0.389
TIP3P	125	7.4	0.983 (0.0328)	-9.128 (0.134)	160	0.467
	216	8.9	0.996 (0.0257)	-9.275 (0.100)	230	0.451
	820	9.0	1.058 (0.0182)	-9.811 (0.0585)	400	0.431
		12.0	1.066 (0.0172)	-9.892 (0.0537)	400	0.413

TABLE III: The density of water, ρ, and the intermolecular potential energy, E_{pot}, obtained with the combined MC/MD constant pressure algorithm. N is the number of water moleculed in each box, r_c is the cut-off used, ratio is the acceptance ratio with the given maximum volume displacement, ΔV_{max}. Values in parenthesis are the standard deviation as given by [15] M13.

The experimental value of the density of water is 0.997 g/cm^3 [15], but the computed density varies with water model and a comparison with other computed data is thus important.

A classical paper is Jorgensen et al. [14] from 1983. In this paper the density is computed with over all Monte Carlo (MC both on positions of water molecules and on the volume of the system) on a box with 125 water molecules using a 7.5 Å cut-off. Their values are 0.971 and 0.982 g/cm^3 for the SPC and TIP3P model, respectively.

A more recent paper is [2], where the density, among other properties, is computed for a number of different water models. The method used is molecular dynamics with pressure scaling combined with the reaction field method to deal with long-range electrostatic interactions. The results for SPC and TIP3P are summarized in Table IV.

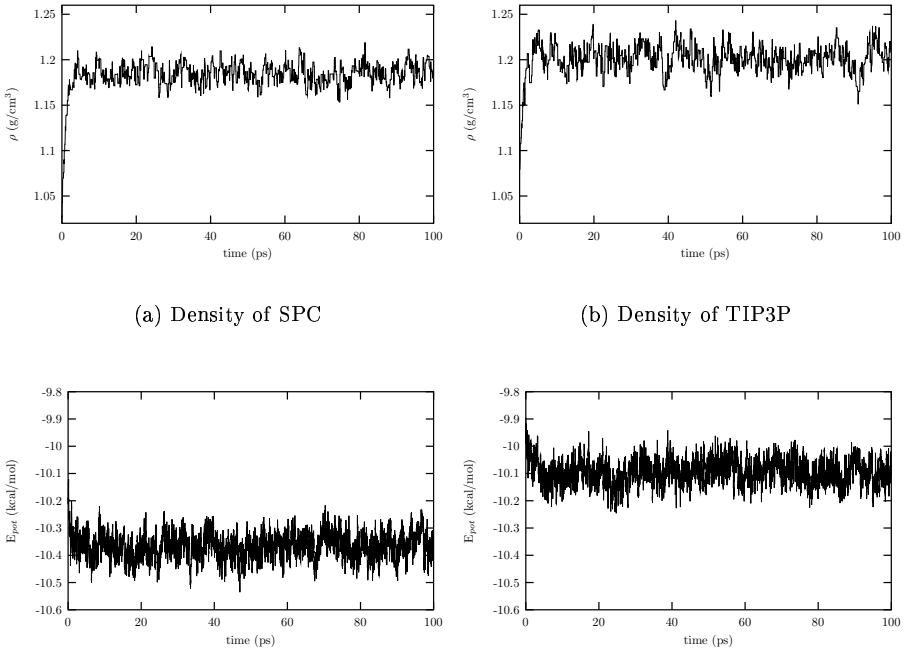
Model	N	r_c (Å)	ρ (g/cm^3)	U_{pot} (kcal/mol)
SPC	216	9.0	0.967(10)	-9.79 (0.079)
	820	9.0	0.968 (6)	-9.79 (0.040)
		12.0	0.971 (5)	-9.86 (0.038)
TIP3P	216	9.0	0.972(11)	-9.38 (0.076)
	820	9.0	0.971 (6)	-9.28 (0.041)
		12.0	0.978 (5)	-9.48 (0.041)

TABLE IV: The density and intermolecular potential energy of water of ref [2]. Energies are converted from kJ/mol by multiplying with a factor of 0.2338 .

In Table III and IV the intermolecular energies, E_{pot}, are shown for different simulations. The experimental value is -9.958 kcal/mol [2] and [14] presents values of -10.18 and -9.86 kcal/mol for SPC and TIP3P.

Also relaxation experiments were performed. A system equilibrated at 1 bar in the isothermal-isobaric ensemble was restarted with target pressure 5000 bar. This is done to monitor how fast the system is shifted towards the new density and after only approximately 100 volume configurations (equaling 4 ps) the new

density is stabilized, Fig 7. Chow and Ferguson [6] do the same for neon with a change in pressure of around 300 bar and report equilibrium within 5 ps. It should be noted however that they generate new volume configurations every single time step and since they use a 1 fs time step this means around 5000 volume configurations.



(a) Density of SPC

(b) Density of TIP3P

(c) Energy of SPC

(d) Energy of TIP3P

FIGURE 7: Relaxation of density and potential energy after change of pressure from 1 to 5000 bar. The graph shows the first 100 ps of a 1.0 ns simulation.

Previously published values of computed densities of water at this high pressure have been hard to find. In [16] Mahoney and Jorgensen present results for a different water model, TIP5P. This model represents water with five points; four charged ones, the hydrogens and the two lone pairs, and one uncharged, the oxygen. The oxygen is the only point contributing to the Lennard–Jones potential, see Eq. (4). The results were obtained with overall MC.

According to [16], the experimental density of water at 5000 atm (=5065 bar) is 1.15 g/cm^3. Their computed results are shown in Table V together with the results of our own for SPC and TIP3P. Please note that [16] made simulations at 5000 atm instead of 5000 bar.

At a first glance the all over results for the new constant pressure algorithm are promising. The computed values of density and potential energy are in reasonable accordance with experimental and previously published data. Moreover, the algorithm gives quick adjustment to new conditions for the systems, however, there are some troublesome features.

Water model	ρ (g/cm^3)	E$_{pot}$ (kcal/mol)
SPC (5000 bar)	1.187 (0.0115)	-10.371 (0.0451)
TIP3P (5000 bar)	1.206 (0.0141)	-10.106 (0.0455)
TIP5P (5065 bar)	1.1828±0.0011	-10.078±0.001

TABLE V: The density and intermolecular potential energy of water at high pressure. The SPC and TIP3P results are obtained with the new constant pressure algorithm. TIP5P results are taken from [16].

First of all, both density and potential energy show a very strong size dependency. The larger the system (and cut-off), the higher the density and the lower the potential energy. The cut-off problem of the smallest box and the medium box with TIP3P is also worth noticing. In general, it seems like the algorithm allows too large fluctuations of volume. For comparison, Jorgensen et al [14] change the volume of their box with 125 water molecules ± 50Å^3 to get an acceptance ratio of $\sim$0.40.

This problems could be due to the algorithm. It is a new algorithm with very little known, and maybe the combination of MC and MD could lead to some, so far unknown, side effects. It could also be a matter of long range corrections. No such corrections were used, but Chow and Ferguson [6] stress that this is important for the sampling.

A more possible explanation, however, is the presence of a bug. The implementation of this algorithm has from time to time required a lot of insecticides, and I give no guarantee that everything is dead. If the problem is due to a bug, however, the discrepancies in results from what is wanted are so small that the bug will be utterly hard to find.

7 Conclusions

The overall periodic boundary conditions implementation works fine as the results from the sodium solvation energy show. This means that the program Q can now do virtually everything with PBC as well as with spherical boundary.

The constant pressure algorithm certainly needs more testing. As mentioned before it is a new method and it needs to be investigated more thoroughly.

8 Acknowledgments

A big thank you to my supervisor Johan Åqvist, and his coworkers in the Åqvist group. I'd also like to thank my examiner David van der Spoel for taking the time to answer my questions and Tom Taylor for proofreading the manuscript.

References

[1] Leach, A.R. 2001. *Molecular Modelling. Principles and Applications.* Second edition. Pearson Education Limited. Essex.

[2] van der Spoel, D., van Maaren, P.J. and Berendsen, H.J.C. 1998. A systematic study of water models for molecular simulation: Derivation of water model optimized for use with a reaction field. *J. Chem. Phys.* 108(24), 10220-10230.

[3] King, G. and Warshel, A. 1989. A surface contained all-atom solvent model for effective simulations of polar solutions. *J. Chem. Phys.* 91, 3647-3661.

[4] Essex, J.W. and Jorgensen, W.L. 1995. An empirical boundary potential for water droplet simulations. *J. Comput. Chem* 16(8), 951-972.

[5] Allen, M.P. and Tildesley, D.J. 1987. *Computer Simulation of Liquids.* Oxford Science Publications.

[6] Chow, K-H. and Ferguson, D.M. 1995. Isothermal-isobaric molecular dynamics simulations with Monte Carlo volume sampling. *Computer Physics Communications.* 91, 283-289.

[7] Marelius, J., Kolmodin, K., Feierberg, I. and Åqvist J. 1998. Q: A molecular dynamics program for free energy calculations and empirical valence bond simulations in biomolecular systems. *Journal of Molecular Graphics and Modelling.* 16, 213-255.

[8] Kolmodin, K. *Computer Simulation of Protein Tyrosine Phosphatase Reaction Mechanism and Dihydrofolate Reductase Inhibition* Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science and Technology 667. Acta Universitatis Upsaliensis. Uppsala.

[9] Uppsala University. Åqvist, J. et al. *Q - our molecular dynamics program.* http://xray.bmc.uu.se/~aqwww/q/default.html 2002-01-18

[10] Berendsen, H.J.C., Postma, J.P.M., van Gunsteren, W.F. and Hermans, J. 1981. Interaction models for water in relation to protein hydration. *Intermolecular Forces.* ed. Pullman, B. Reidel, Dordrecht. p. 331-342.

[11] Åqvist, J. 1990. Ion-water interaction potentials eerived from free energy perturbation simulations. *J. Phys. Chem.* 94(21), 8021-8024.

[12] Åqvist, J. 1994. Comment on "Transferability of Ion Models". *J. Phys. Chem.* 98(33), 8253-8255.

[13] Kolmodin, K. and Åqvist, J. 1999. Computational modelling of catalysis and binding in low-molecular-weight protein tyrosine phosphatase. *Int. J. Quant. Chem.* 73(2), 147-159.

[14] Jorgensen, W.L., Chandrasekhar, J., Madure J.D., Impey, R.W. and Klein, M.L. 1983. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* 79(2), 926-935.

[15] Nordling, C. and Österman, J. 1987. *Physics handbook; elementary constants and units, tables, formulae and diagr. and mathemat. formulas.* Studentlitteratur. Lund, Sweden.

[16] Mahoney M.W. and Jorgensen W.L. 2000. A fice-site model for liquid water and the reproduction of the density anomaly by rigid, nonpolarizable potential functions. *J. Chem. Phys.* 112(20), 8910-8922.