UNIVERSITY OF ABERDEEN

HONOURS DISSERTATION

CHRISTOPHER JAMES THOMSON
2012

THIS THESIS WAS SUBMITTED AS PART OF THE REQUIREMENT FOR THE MENG. DEGREE IN ENGINEERING.

School of Engineering 2012 Univ. of Aberdeen

ABSTRACT

The objective of this thesis is to determine a general technique for converting continuous potentials to equilivant discrete potentials. Discrete potentials have many desirable properties.

CONTENTS

Final word count:

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

Acronyms/Terminology

FCC Face Centered Cubic, 22

ACKNOWLEDGEMENTS

I would like to dedicate this work to..

CHAPTER

ONE

INTRODUCTION

Process simulation packages have become an integral part of chemical engineering design. Central to these simulation software packages is the ability to calculate thermodynamic and transport properties of fluids quickly and accurately. Many modern processes rely on molecular scale effects... (Absorption, membrane technology (reverse osmosis), catalysis)....

To better understand these large scale systems, we need to improve our understanding of the smaller scales. Experiments are hard, can't hold a ruler up to a molecule, everything is too fast, too small to see. (X-ray crystallography). theory is hard, lots of molecules, can't solve even three molecules motion analytically (need cite). simulations are great, don't try to solve analytically. Since 50's (alder wainwright), computers are faster, modern sims are amazing.

At the heart of these simulations are models for the atoms and molecules involved. There are two classes of models, discrete and contiuous.

In chapter **??**, the something is discussed

CHAPTER

TWO

MOLECULAR MODELS

In this chapter, the models used for molecular dynamics are discussed. The two major types of potentials: continuous and discrete potentials, both are important to molecular dynamics.

2.1 Classical Mechanics

The underlying assumption behind many molecular dynamics simulations is that the particles move according to the laws of classical mechanics. Strictly, due to their size and speed, atom and molecules should be treated using quantum mechanics. However molecular dynamics makes a couple of assumptions that allow these quantum mechanical effects to be ignored. The first is the Born-Oppenheimer Approximation which allows the motion of electrons and the nucleus to be treated separately. Since the nucleus is much larger than the electrons and hence less affected by quantum mechanics, it is treated as a classical particle. The electrons on the other hand are represented using a potential. The second assumption is that any quantum mechanical effects should cancel out. Molecular dynamics is rarely interested in the motion of a single particle, it is more concerned with the statistical average of every particle.

The effects fo quantum mechanics are usuallysmall unless very light atoms (such as hydrogen or helium) are being simulated or the particles are vibrating at very high rates [3]

The fundamental identity of newtonian mechnanics is Newton's Second Law of Motion (3.1.1). This equation allows the prediction of a particle's trajectory provided that an initial position and velocity is known; and the forces acting on that particle can be calculated for any position or velocity.

$$\vec{F} = m\vec{a} \tag{2.1.1}$$

If a force depends only on the position of a particle it is known as a conservative

force. Almost all forces considered in molecular dynamics are of this type because atoms or molecules do not lose energy due to friction or any other dissipative process.

Conservative forces can be further subdivided into forces that depend either on absolute position such as gravity or forces can depend on position relative to another particle (intermolecular forces). Gravity is usually neglected in MD simulations as the mass of atoms and molecules is very small.

While the forces caused by groups of particles should also be considered, this would severly complicate the simulation, and hence is often ignored.

In MD simulations the intermolecular forces are usually described using potentials.

2.2 Continuous Potentials

LJ

more complicated stuff..

2.3 Discontinuous Potentials

Hard sphere,

square well -> compare to LJ

SPEAMD, PRIME

Lead out with the two methods used to simulate these potentials are quite different....

CHAPTER

THREE

MOLECULAR DYNAMICS

3.1 Introduction

The underlying assumption behind many molecular dynamics simulations is that the particles move according to the laws of newtonian mechanics. The effects fo quantum mechanics are usuallysmall unless very light atoms (such as hydrogen or helium) are being simulated or the particles are vibrating at very high rates [3]

The fundamental identity of newtonian mechnanics is Newton's Second Law of Motion (3.1.1). This equation allows the prediction of a particle's trajectory provided that an initial position and velocity is known; and the forces acting on that particle can be calculated for any position or velocity.

$$\vec{F} = m\vec{a} \tag{3.1.1}$$

If a force depends only on the position of a particle it is known as a conservative force. Almost all forces considered in molecular dynamics are of this type because atoms or molecules do not lose energy due to friction or any other dissipative process.

Conservative forces can be further subdivided into forces that depend either on absolute position such as gravity or forces can depend on position relative to another particle (intermolecular forces). Gravity is usually neglected in MD simulations as the mass of atoms and molecules is very small.

While the forces caused by groups of particles should also be considered, this would severly complicate the simulation, and hence is often ignored.

In MD simulations the intermolecular forces are usually described using potentials.

3.1.1 Potentials

A potential is the function of potential energy with position, where position is usually expressed as a set of orthogonal vectors such as x, y and z in three dimensions. Con-

servative forces can be calculated from their potential by equation 3.1.2. The gradient of the potential, denoted by ∇ is the partial differential of the potential in each orthogonal direction.

$$\vec{F} = -\nabla U \tag{3.1.2}$$

A very popular potential used in molecular dynamics simulations is the Lennard-Jones potential [10] shown in figure 3.1 and equation 3.1.3 as it is simple yet gives comparable results to experimental values.

$$U(r) = 4\varepsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right] \tag{3.1.3}$$

In equation 3.1.3, ε is the depth of the energy well, while σ is the root of the Lennard-Jones potential which corresponds to the change from attractive to repulsive forces (see figure 3.2), this is taken to be the diameters of the particles during the collision.

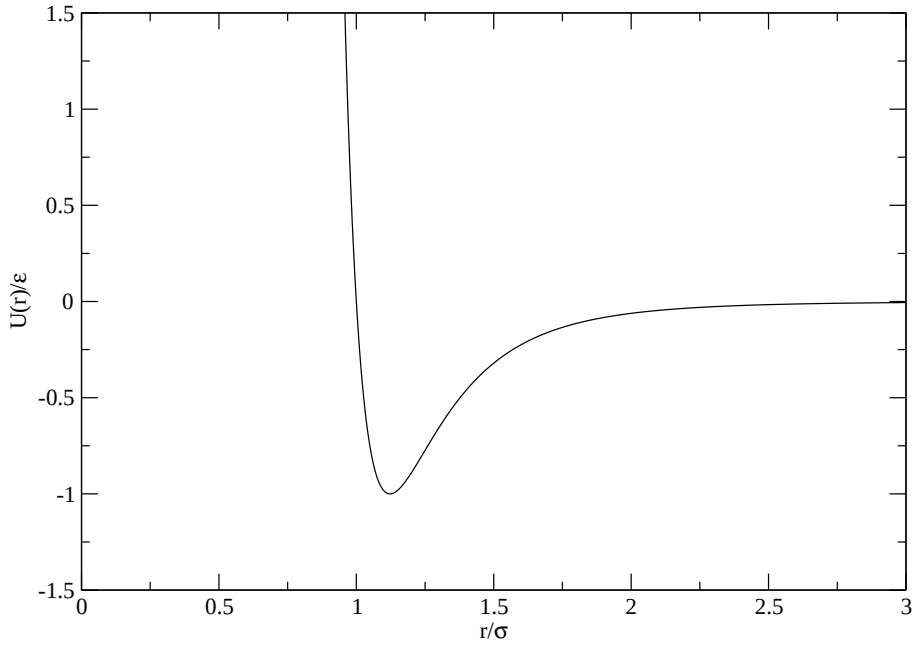

Figure 3.1: Plot of the Lennard-Jones potential

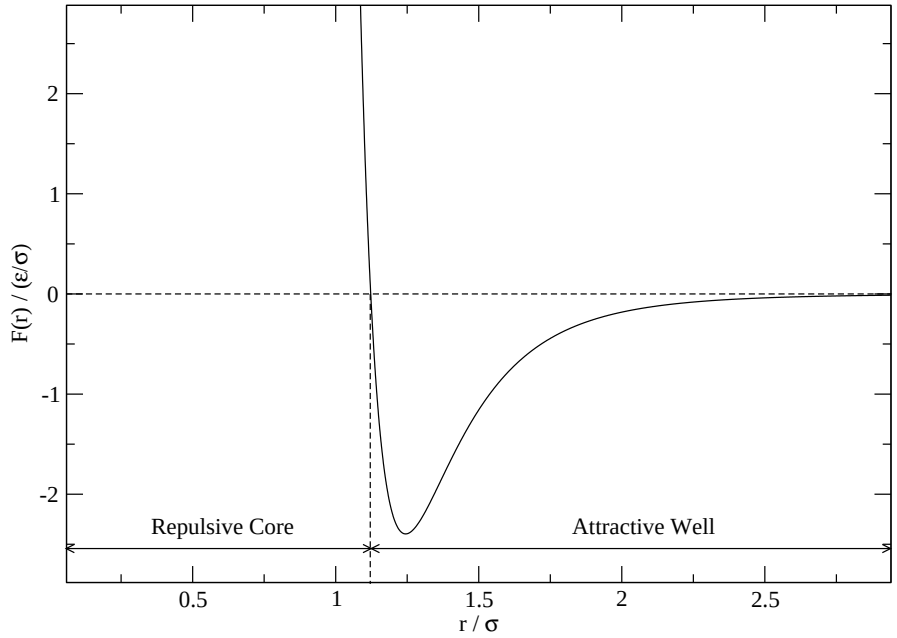

Figure 3.2: Plot of the Lennard-Jones force between a pair of particles

3.2 Force-Driven Simulators

3.2.1 Introduction

Force-driven (or time driven) simulators are the most popular method of simulating particles due to their relative simplicity and ability to handle soft potentials. Simulators of this kind were pioneered by Rahman [13] who predicted physical properties of liquid argon with reasonable accuracy.

The distinguishing feature between force-driven and event-driven simulators is the way in which they move through time. During force-based simulations particles' positions and velocities are calculated every unit of time, Δt using the forces acting on the particles. These newly calculated values are then used to predict the next set of particle positions. This is then repeated over the desired simulation time.

3.2.2 Integrators

Once the forces acting on a particle is known, that particle's acceleration can be calculated using Newton's Second Law of Motion (equation 3.2.1).

$$\vec{F} = m\frac{\partial^2 \vec{r}}{\partial t^2} \tag{3.2.1}$$

However, since acceleration is the second time derivative of position (velocity is the first time deriviative), calculating the particle's future position results in solving a differential equation of order 2 or higher since force, andhence acceleration likely changes with time. In order to accomplish this numerical integrators are used.

The majority of numerical integrators are based on Taylor Series (equation 3.2.2).

$$\vec{r}(t+\Delta t) = r(t) + \frac{\partial \vec{r}(t)}{\partial t}(\Delta t) + \frac{1}{2}\frac{\partial^2 \vec{r}(t)}{\partial t^2}\Delta t^2 + \frac{1}{3!}\frac{\partial^3 \vec{r}(t)}{\partial t^3}\Delta t^3 + \frac{1}{4!}\frac{\partial^4 \vec{r}(t)}{\partial t^4}\Delta t^4 + \ldots \quad (3.2.2)$$

The simplest integrator is Euler's Method which is just the Taylor Series truncated after the acceleration term (equation 3.2.3).

$$\vec{r}(t+\Delta t) = r(t) + \vec{v}(\Delta t) + \frac{1}{2}\vec{a}\Delta t^2 + \mathcal{O}(\Delta t^3) \quad (3.2.3)$$

However this method suffers from large errors and is unstable [6] and is therefore rarely used. The Verlet Integrator [15] improves upon Euler's method by combining the forward timestep with a reverse timestep (3.2.4a). This method is actually fourth order as the third (and first) derivative is cancelled out during its derivation. The Verlet integrator does not include an equation to calculate the future velocity so the central difference used by Verlet is often used (3.2.4b).

$$\vec{r}(t+\Delta t) = 2\vec{r}(t) - \vec{r}(t-\Delta t) + \vec{a}(t)\Delta t^2 + \mathcal{O}(\Delta t^4) \quad (3.2.4a)$$

$$\vec{v}(t+\Delta t) = \frac{\vec{r}(t+\Delta t) - \vec{r}(t-\Delta t)}{2\Delta t} \quad (3.2.4b)$$

Integrators suffer from couple key failings that cause a systematic gain of energy known as "energy drift". Firstly, integrators are based on infinite Taylor series which cannot be fully implemented, therefore they have to be truncated after a certain number of terms, this introduces an error. Secondly, integrators struggle to predict values of forces that have discontinuities in them, such as hard spheres which only have forces on contact but not before, or discontinuities introduced by truncating potentials to improve simulator speed. There are a couple of types of integrators that try and reduce these problems.

Sympletic integrators have the useful property in that they, on average, conserve energy [7]. The most common symplectic integrator used in MD is the Velocity Verlet Integrator [14] shown in (3.2.5).

$$\vec{r}(t+\Delta t) = \vec{r}(t) + \vec{v}(t)\Delta t + \frac{1}{2}\vec{a}(t)\Delta t^2 + \mathcal{O}(\Delta t^4) \quad (3.2.5a)$$

$$\vec{v}(t+\Delta t) = \vec{v}(t) + \frac{\vec{a}(t) + \vec{a}(t+\Delta t)}{2}\Delta t \quad (3.2.5b)$$

The popularity of the Velocity Verlet is due to its computational simplicity andits accuracy and stability at relatively long timesteps. It can even be expanded[8] to maintain its accuracy and stability at very long timesteps at a small extra computational cost. However the Velocity Verlet cannot be used in systems that do not conserve energy, ie systems with dissipative forces.

Another popular method of improving the traditional integrator is predictor-corrector integrators. These use a truncated Taylor series to calculated a predicted value for the future position and higher order time derivatives. The force is then calculated at this predicited position, then the difference between the predicted acceleration and the corrected acceleration calculated from the force is used to correct the position and time derivatives.

The most popular predictor-corrector integrator is that of Gear [4],using his 5th order algorithm. The predicted value for the i^{th} time derivative is shown in (3.2.6), and defining $\Delta\vec{a} = \vec{a}^{\,C} - \vec{a}^{\,P}$, the corrected time derivatives can be calculated using (3.2.7) with coefficients from (3.2.8).

$$\frac{\partial^i}{\partial t^i}\vec{r}^{\,P}(t+\Delta t) = \sum_{k=i}^{n} \frac{1}{k!}\frac{\partial^i}{\partial t^i}\vec{r}(t)\Delta t^k \tag{3.2.6}$$

$$\frac{\partial^i}{\partial t^i}\vec{r}^{\,C}(t+\Delta t) = \frac{\partial^i}{\partial t^i}\vec{r}^{\,P}(t+\Delta t) + \frac{c_i}{\Delta t^i}\left(\frac{\Delta t^2}{2}\Delta\vec{a}\right) \tag{3.2.7}$$

$$c_0 = \frac{3}{16},\ \ c_1 = \frac{251}{360},\ \ c_2 = 1,\ \ c_3 = \frac{11}{18},\ \ c_4 = \frac{1}{6},\ \ c_5 = \frac{1}{60} \tag{3.2.8}$$

The Gear's algorithm, while more accurate at short timesteps than Verlet's integrator [6], suffers at long timesteps and is computationally more expensive.

3.3 Event-Driven Simulators

Start off by pointing out that discontinuous potentials have discontinuities! Whats the force on these points? Infinity! Can't integrate over these discontinuities, but what about between them? We can solve F=ma, (solve it). We can then analytically integrate newton's equation of motion. We can then try to detect when a discontinuity occurs and treat them as they happen.

How do you solve what happens over a discontinuity? Conservation of momentum and energy (see appendix).

3.3.1 Introduction

Though force-driven simulators are more popular the first MD simulation was done using an event-driven simulator by Alder and Wainwright [1]. The difference between force and event driven simulations lies in how the simulators move through time. While force driven simulators move forward in uniform blocks, event driven simulators jump between successive collisions. These collisions are taken to be instantaneous and only one can occur at any particular time.

3.3.2 Collision Time Prediction

An event-driven simulator first must calculate the collision times between every pair of particles (provided the particles do collide), in order to select the earliest collision. For hard sphere simulations there are two conditions that must be satisfied in order for a collision to occur. Firstly, the particles must be moving towards each other and secondly, the particles must pass close enough to each other to collide. These conditions can be expressed mathematically in equations (3.3.1a) and (3.3.1b) respectively [6].

$$\mathbf{v}_{ij} \cdot \mathbf{r}_{ij} < 0 \tag{3.3.1a}$$

$$(\mathbf{v}_{ij} \cdot \mathbf{r}_{ij})^2 - v_{ij}^2(r_{ij}^2 - \sigma^2) \geq 0 \tag{3.3.1b}$$

The time to collision can then be calculated using the quadratic in equation (3.3.2). While there are two solutions, only the earliest collision (the negative root) needs to be considered. The second root gives the time when the particles leave after passing through each other which, for hard sphere, cannot happen.

$$\Delta t = \frac{(-\mathbf{v}_{ij} \cdot \mathbf{r}_{ij}) \pm \sqrt{(\mathbf{v}_{ij} \cdot \mathbf{r}_{ij})^2 - v_{ij}^2(r_{ij}^2 - \sigma^2)}}{v_{ij}^2} \tag{3.3.2}$$

Since $\mathbf{v}_{ij} \cdot \mathbf{r}_{ij}$ must be negative for the collision, there is the possibility of catastrophic

cancellation [5], if $(\mathbf{v}_{ij} \cdot \mathbf{r}_{ij})^2 \gg v_{ij}^2(r_{ij}^2 - \sigma^2)$. Therefore it is advisable to use the positive root from the alternate form of the quadratic equation given in equation (3.3.3) [12].

$$\Delta t = \frac{r_{ij}^2 - \sigma^2}{(-\mathbf{v}_{ij} \cdot \mathbf{r}_{ij}) \mp \sqrt{(\mathbf{v}_{ij} \cdot \mathbf{r}_{ij})^2 - v_{ij}^2(r_{ij}^2 - \sigma^2)}} \tag{3.3.3}$$

When considering stepped potentials many of the same principles apply, except there are now two possible "collisions". The first, when two particles enter a step is treated identically to hard spheres. The other event, when the particles leave the step, is calculated using the second, later root of the quadratic. In order to prevent loss of numerical precision, the leaving time should be calculated using equation (3.3.4).

$$\Delta t = \begin{cases} \frac{r_{ij}^2 - \sigma^2}{(-\mathbf{v}_{ij} \cdot \mathbf{r}_{ij}) - \sqrt{(\mathbf{v}_{ij} \cdot \mathbf{r}_{ij})^2 - v_{ij}^2(r_{ij}^2 - \sigma^2)}}, & \text{if } \mathbf{v}_{ij} \cdot \mathbf{r}_{ij} > 0 \\ \\ \frac{(-\mathbf{v}_{ij} \cdot \mathbf{r}_{ij}) + \sqrt{(\mathbf{v}_{ij} \cdot \mathbf{r}_{ij})^2 - v_{ij}^2(r_{ij}^2 - \sigma^2)}}{v_{ij}^2}, & \text{if } \mathbf{v}_{ij} \cdot \mathbf{r}_{ij} < 0 \end{cases} \tag{3.3.4}$$

3.3.3 Collision Dynamics

Once the time of the next collision is known, the particles can be moved to their new locations. Generally there is no external force applied to the particles in event-driven molecular dynamics, therefore the particles move in straight lines. Hence the particles' new positions can be calculated using equation (3.3.5).

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\Delta t \tag{3.3.5}$$

The post-collision velocities of the colliding particles must now be calculated. The simplest collision between two particles is an elastic bounce, where the velocities and just exchanged along the separation vector between the two particles. The change in velocity during the collision for particles i and j is shown in equation (3.3.6).

$$\Delta \mathbf{v}_i = -(\mathbf{v}_{ij} \cdot \hat{\mathbf{r}}_{ij})\hat{\mathbf{r}}_{ij} \tag{3.3.6a}$$

$$\Delta \mathbf{v}_j = (\mathbf{v}_{ij} \cdot \hat{\mathbf{r}}_{ij})\hat{\mathbf{r}}_{ij} \tag{3.3.6b}$$

For stepped potential system, the collision dynamics are more complex. When two particles collide they must pay an energy "cost" to proceed through the step. This energy cost ΔU is the difference in the energy of the current step and the step the particles are going into, and is shown in equation (3.3.7).

$$\Delta U = U_{\text{next step}} - U_{\text{current step}} \tag{3.3.7}$$

If the kinect energy of the particles is insufficient, the pair bounce off the step and the post-collision velocities are calculated using equation (3.3.6). However, if the particles can pay this cost i.e. the inequality (3.3.8) is true, then the particles can enter the step.

$$\frac{1}{4}m(\mathbf{v}_{ij}\cdot\hat{\mathbf{r}}_{ij})^2 > \Delta U \tag{3.3.8}$$

The change in the velocities of particles i and j after going through a step are shown in equation (3.3.9) where A is given in equation (3.3.10), the derivation of these equations is given in Appendix ???. If the particles are entering a step, the positive root of A is used, whereas if the particles are leaving a step it is the negative root that should be used.

$$\Delta\mathbf{v}_i = \frac{A}{m}\hat{\mathbf{r}}_{ij} \tag{3.3.9a}$$

$$\Delta\mathbf{v}_j = -\frac{A}{m}\hat{\mathbf{r}}_{ij} \tag{3.3.9b}$$

$$A = -\frac{m}{2}\left((\mathbf{v}_{ij}\cdot\hat{\mathbf{r}}_{ij}) \pm \sqrt{(\mathbf{v}_{ij}\cdot\hat{\mathbf{r}}_{ij})^2 - \frac{4}{m}\Delta U}\right) \tag{3.3.10}$$

3.4 Measurement of System Properties

3.4.1 Introduction

Molecular dynamics is a useful tool to predict macroscopic properties of particles systems. Many of the identities and methods to measure these properties are derived in statistical mechanics. However, even when the system is at equilibrium, these properties fluctuate around a mean (see figure 3.3) therefore it is common to take time averages of these values. These time averages are denoted with angle brackets $\langle\ \rangle$, and the time average of a property, A, is shown in (3.4.1) [6].

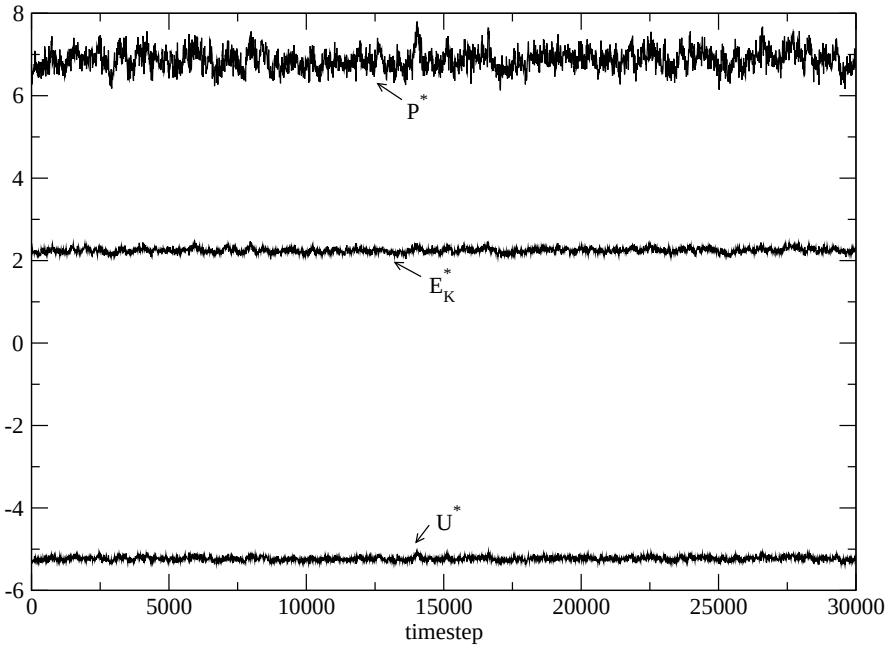

Figure 3.3: Plot showing flucuation of pressure ($P^* = P\sigma^3/\varepsilon$), kinetic energy per particle ($E_K^* = E_K/N\varepsilon$) and potential energy per particle ($U^* = U/N\varepsilon$). Results are from a force-driven simulation involving 864 particles, at a density $\rho^* = 0.9$ and temperature $\langle T \rangle = 1.497$. Values were collected every 10 timesteps where each timestep was $\Delta t = 0.005$.

$$\langle A \rangle = \lim_{t \to \infty} \frac{1}{t} \int_{t_0}^{t_0+t} A(\tau) d\tau \tag{3.4.1}$$

This time average can be calculated precisely in event-driven simulations for several properties such as pressure, kinetic energy and potential energy. These only change at collisions and therefore are constant for the time between the collision and can be calculated by (3.4.2).

$$\langle A \rangle = \frac{1}{t} \sum_{t_0}^{t_o+t} A(\tau) \Delta\tau \tag{3.4.2}$$

In force-driven simulators all properties change continuously and hence time averages cannot be calculated precisely, however approximations can be made. If properties are measured every uniform period of time, the time average can be approximated by equation (3.4.3), where M is the number of measurements taken.

$$\langle A \rangle = \frac{1}{M} \sum_{1}^{M} A(\tau) \tag{3.4.3}$$

3.4.2 Units

In molecular dynamics simulations, properties are frequently measured in dimensionless forms [6]. These "reduced units" are usually denoted with an asterisk. In order to achieve this a number of fundamental dimensions are needed: a characteristic length σ, a characteristic energy ε, and the mass of one particle m. In the case of the Lennard-Jones potential, the characteristic length and energy are taken as: the distance of the root, and the depth of the attractive well respectively. A table of reduced forms are given in table 3.1.

Table 3.1: Table of reduced forms of various quanities used in this dissertation [6]

Quantity	Reduced forms
Density	$\rho^* = N\sigma^3/V$
Energy	$E^* = E/\varepsilon$
Force	$F^* = F\sigma/\varepsilon$
Length	$r^* = r/\sigma$
Pressure	$P^* = P\sigma^3/\varepsilon$
Temperature	$T^* = kT/\varepsilon$
Time	$t^* = t/(\sigma\sqrt{m/\varepsilon})$
Velocity	$v^* = v\sqrt{m/\varepsilon}$

3.4.3 Energy

Perhaps one of the most important properties to measure in MD simulations is the total internal energy of the system. For isolated systems, i.e. systems where mass or energy cannot enter or leave the system, this internal energy is the sum of kinetic and potential energy (equation (3.4.4)).

$$E = E_K + U \tag{3.4.4}$$

The total kinetic energy in the system is the sum of the kinetic energy of each particle, as shown in equation (3.4.5).

$$E_K = \sum_{i}^{N} m v_i^2 \tag{3.4.5}$$

The potential energy of the system is the sum of the potential energy between every pair of particles (for a pairwise potential), and is shown in equation (3.4.6).

$$U = \sum_{i\ <\ j} \sum U(r_{ij}) \tag{3.4.6}$$

Event-driven simulators strictly conserve energy, therefore the kinetic and potential energy can be measured at the beginning of the simulation and then updated whenever either changes e.g. when a collision occurs.

3.4.4 Temperature

The velocity distrubution of particles is given by the Maxwell distribution [6], shown in equation (3.4.7).

$$f(v_x)dv_x = \sqrt{\frac{m}{2\pi kT}} e^{-\frac{mv_x^2}{2kT}} \tag{3.4.7}$$

This is the form of a Gaussian distribution and it can be shown [9] that the mean square velocity in any direction is as shown in equation (3.4.8).

$$\bar{v_x^2} = \frac{kT}{m} \tag{3.4.8}$$

Making the assumption that the velocity distribution is the same in each direction, the temperature can be expressed as equation (3.4.9), by taking the average temperature in each direction.

$$T^* = kT = \frac{mv^2}{3} = \frac{2}{3} E_K \tag{3.4.9}$$

CHAPTER

FOUR

METHODOLOGY

4.1 Introduction

The two simulators described in this chapter were coded in C++ using the C++ Standard Library with the Boost [cite] library to handle random number generation.

4.2 Force-driven Simulator

Force driven simuators are currently the dominant MD paradigm therefore it was decided that the results from the stepped potentials would be compared to the equivalent results obtained from a force-driven simulator. In order to acquire these results a force-driven simulator was written.

The algorithm for the force driven simulator is as follows.

1. Initialisation
2. Calculate particles' future positions
3. Calculate the forces acting on the particles
4. Calculate the future velocities of particles
5. Run thermostat (if enabled)
6. Measure properties
7. Repeat steps 2-6 for the desired number of iterations

4.2.1 Initialisation

The particles are initialised in a Face Centered Cubic (FCC) structure. The use of the FCC lattice is common when simulating Lennard-Jones potentials as the first force-driven simulation [13] was carried out using liquid Argon which crystalises to a FCC lattice.

Particle velocities are assigned randomly from a Gaussian distribution with a mean, $\mu = 0$, and a standard deviation, $\sigma = \sqrt{T^*}$, where T^* is the desired reduced temperature. The velocities are then rescaled to ensure there net shift in linear momentum in any direction by applying (4.2.1) in each orthongonal direction.

$$v_i^{new} = v_i^{old} - \frac{1}{N}\sum_{i}^{N} v_i^{old} \tag{4.2.1}$$

4.2.2 Running Simulation

The

CHAPTER

FIVE

MEASURING THERMODYNAMIC PROPERTIES

5.1 Temperature and Internal energy

5.2 Pressure (continuous and discontinuous)

5.3 g(r)

5.4 g(r) to pressure and temperature

5.5 long range corrections

CHAPTER

SIX

FROM CONTINUOUS TO DISCONTINUOUS

State the case why we want to run discontinuous systems, the advantages (fast, extremely stable-> infinitely hard potentials, and disadvantages (underdeveloped set of potentials in the literature, complex algorithm). These disadvantages can be overcome by writing a good general EDMD program and finding a a general method to convert the hard work in soft potentials to stepped potentials.

CHAPTER

SEVEN

RESULTS

7.1 Benchmarking

7.1.1 Introduction

After a MD simulator has been created it is necessary to compare its results with those generated by others, to verify that the simulator works correctly.

7.1.2 Force based code verusus NIST and ESpReSSo

7.1.3 Event Driven

Event driven codes are significantly more complex than time-stepping codes, so we need more tests

HARD SPHERES v LEO

Stepped Potential of Chapela

7.2 Chapela's dumb stepping and very good stepping versus force based.

Show that dumb stepping doesn't work, show how good chapela's results actually are when he tries. Proves this is possible.

7.3 Stepping in probability versus action

Hard cores dominate the freezing behaviour (see Alder and Wainwrights famous paper) and therefore for high density pressure etc. but the stepping in equal probability doesnt do this.

Table 7.1: Comparison of results obtained by the event-driven simulator with literature values. t_{avg} is the average time between collisions, $\langle \hat{\mathbf{r}} \cdot \Delta \mathbf{v} \rangle_{coll}$ is the average momentum transfer per collision, and D is the coefficient of diffusion.

ρ	t_{avg}		$\langle \hat{\mathbf{r}} \cdot \Delta \mathbf{v} \rangle_{coll}$		D	
	Simulator	Lue	Simulator	Lue	Simulator	Lue
0.3	0.3052	0.3052	1.775	1.772	0.53	0.55
0.4	0.1944	0.1942	1.776	1.773	0.341	0.359
0.5	0.13024	0.13031	1.774	1.7724	0.247	0.247
0.6	0.08966	0.08968	1.771	1.7721	0.169	0.173
0.7	0.0625	0.0625	1.773	1.776	0.114	0.113
0.8	0.04365	0.0436	1.772	1.772	0.064	0.065
0.9	0.03029	0.03024	1.773	1.772	0.033	0.0327

7.4 Hard core position

Try setting the inner step to infinite energy

Use barker henderson, an old attempt to make hard sphere match everything else. (too far out).

talk about probability of finding a particle in the core. Talk about sigma try out 3, 4, and 5.

7.5 Temperature comparisons

Show that chapelas solution gets worse faster than ours.

7.5.1 Event-Driven Simulator

The event-driven simulator was first tested running a hard sphere simulation before testing the more complex stepped potentials. A single 'step' with a energy requirement sufficiently large such that no particle could enter it. The simulation was run once at a range of densities using 864 particles at a reduced temperature of $T^* = 1$ for 5 million collisions, the results were compared with those of Lue [11] in table 7.1. The agreement between results is good and lies within statistical uncertainty. The largest discrepancies are in the values for the coefficient of diffusion at low densities which is probably due to Lue's values were obtained after 10 million collisions.

The simulator was then benchmarked using a step potential. The results were compared with Chapela et al [2] using their 'Case 6' steps. The simulation was run for 1.5 million collisions using 864 particles. Each simulation was run ten times and the mean values and standard deviations are given in table 7.2

Table 7.2: Comparison of results obtained by the event-driven simulator with literature values using stepped potentials. Numbers in parenthesis indicate the uncertainty in the final digit.

ρ	$\langle T \rangle$		$\langle U \rangle$		$\langle P \rangle$	
	Simulator	Chapela et al	Simulator	Chapela et al	Simulator	Chapela etal
0.85	0.719(3)	0.72	-6.04(7)	-5.80	-0.5(4)	0.54
0.85	1.339(8)	1.34	-5.130(9)	-5.14	4.08(4)	4.08
0.85	2.35(1)	2.35	-4.24(2)	-4.20	8.78(9)	8.86
0.85	3.37(2)	3.37	-3.48(2)	-3.49	12.90(9)	13.00
0.85	4.59(1)	4.60	-2.67(1)	-2.68	17.31(8)	13.43
0.75	0.811(2)	0.81	-5.095(3)	-5.08	-0.20(2)	-0.24
0.75	1.309(9)	1.31	-4.67(1)	-4.63	1.81(5)	1.84
0.75	2.49(1)	2.49	-3.88(1)	-3.82	5.80(4)	5.95
0.75	3.59(2)	3.59	-3.26(1)	-3.22	9.03(7)	9.20
0.65	1.309(8)	1.31	-4.081(8)	-4.06	0.80(3)	0.81
0.65	2.61(1)	2.61	-3.42(1)	-3.41	3.86(5)	3.89
0.65	3.79(1)	3.79	-2.926(9)	-2.94	6.34(7)	6.33

7.6 References

[1] B. J. Alder and T. E. Wainwright. "Studies in Molecular Dynamics. I. General Method". In: *J. Chem. Phys.* 31 (1959), pp. 459–466. DOI: 10.1063/1.173037.

[2] G. Chapela, L. E. Scriven, and H. T. Davis. "Molecular dynamics for discontinuous potential. IV. Lennard-Jonesium". In: *J. Chem. Phys.* 91 (1989), p. 4307. DOI: 10.1063/1.456811.

[3] D. Frenkel and B. Smit. *Understanding Molecular Simulations*. 2nd Edition. Academic Press, 2002. DOI: 10.1016/B978-012267351-1/50006-7.

[4] C. W. Gear. *Numerical Initial Value Problems in Ordinary Differential Equations*. Ed. by George Forsythe. Prentice-Hall, 1971.

[5] D. Goldberg. "What every computer scientist should know about floating-point arithmetic". In: *ACM Comput. Surv.* 23.1 (1991), pp. 5–48. DOI: 10.1145/103162.103163.

[6] J M Haile. *Molecular Dynamics Simulation: Elementary Methods*. Wiley-Interscience, 1997.

[7] Ernst Hairer, Christian Lubich, and Gerhard Wanner. "Geometric numerical integration illustrated by the Stormer-Verlet method". In: *Acta Numerica* 12 (2003), pp. 399–450. DOI: 10.1017/S0962492902000144.

[8] Z.M. Khakimov. “New integrator for molecular dynamics simulations”. In: *Comput. Phys. Commun.* 147 (2002), pp. 733–736. DOI: 10.1016/S0010-4655(02)00387-9.

[9] L.D. Landau and E.M. Lifshitz. *Statistical Mechanics*. 2nd. Vol. 5. Course of Theoretical Physics. Oxford: Pergamon, 1968.

[10] J.E. Lennard-Jones. “The Determination of Molecular Fields I & II”. In: *Proc. R. Soc. Lon. Ser-A* 106A (1924), pp. 441–477. DOI: 10.1098/rspa.1924.0081.

[11] L. Lue. “Collision statistics, thermodynamics, and transport coeffiecients of hard hyperspheres in three, four, and five dimensions”. In: *J. Chem. Phys.* 122.4 (2005), p. 044513. DOI: 10.1063/1.1834498.

[12] T. Pöschel and T. Schwager. *Computational Granular Dynamics: Models and Algorithms*. Springer, 2005.

[13] A Rahman. “Correlations in the Motion of Atoms in Liquid Argon”. In: *Phys. Rev.* 136 (1964), A405–A411. DOI: 10.1103/PhysRev.136.A405.

[14] William C. Swope et al. “A computer simulation method for the calculation of equilibrium contants for the formation of physical clusters of molecultes: Application to small water clusters”. In: *J. Chem. Phys.* 76 (1982), pp. 637–649. DOI: 10.1063/1.442716.

[15] Loup Verlet. “Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules”. In: *Phys. Rev.* 159 (1967), pp. 98–103. DOI: 10.1103/PhysRev.159.98.

APPENDIX

A

DERIVATION OF COLLISION DYNAMICS FOR STEPPED POTENTIALS

Considering a collision between particles i and j, each with mass, m with a step energy difference of ΔU, the conservation of momentum is shown in equation (A.0.1). Here the prime indicates post-collision values.

$$m\mathbf{v}_i + m\mathbf{v}_j = m\mathbf{v}_i' + m\mathbf{v}_j' \tag{A.0.1}$$

The momentum change of each particle must occur along the separation vector between the two particles, which can be expressed by equation (A.0.2), where A is an arbitary coefficient.

$$m\mathbf{v}_i - m\mathbf{v}_i' = -(m\mathbf{v}_j - m\mathbf{v}_j') = -A\hat{\mathbf{r}}_{ij} \tag{A.0.2}$$

Energy must also be conserved in the system so equation (A.0.3) must also apply. This can be rewritten to equations (A.0.4) and (A.0.5)

$$\frac{1}{2}mv_i^2 + \frac{1}{2}mv_j^2 = \frac{1}{2}mv_i'^2 + \frac{1}{2}mv_j'^2 + \Delta U \tag{A.0.3}$$

$$v_i^2 - v_i'^2 + v_j^2 - v_j'^2 - \frac{2}{m}\Delta U = 0 \tag{A.0.4}$$

$$(\mathbf{v}_i - \mathbf{v}_i') \cdot (\mathbf{v}_i + \mathbf{v}_i') + (\mathbf{v}_j - \mathbf{v}_j') \cdot (\mathbf{v}_j + \mathbf{v}_j') - \frac{2}{m}\Delta U = 0 \tag{A.0.5}$$

Equation (A.0.2) can now be substituted into (A.0.5) to give equation (A.0.6).

$$\frac{A}{m}\hat{\mathbf{r}}_{ij}(\mathbf{v}_j - \mathbf{v}_i + \mathbf{v}_j' - \mathbf{v}_i') - \frac{2}{m}\Delta U = 0 \tag{A.0.6}$$

Equation (A.0.2) and the definition of the separation velocity vector ($\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$) can be substituted into equation (A.0.6) to give (A.0.7).

$$-\frac{A^2}{m} - A\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij} - \Delta U = 0 \tag{A.0.7}$$

This is a quadratic equation in terms of A therefore it's roots must be given by equation (A.0.8).

$$A = -\frac{m}{2}\left((\mathbf{v}_{ij} \cdot \hat{\mathbf{r}}_{ij}) \pm \sqrt{(\mathbf{v}_{ij} \cdot \hat{\mathbf{r}}_{ij})^2 - \frac{4}{m}\Delta U}\right) \tag{A.0.8}$$

From equation (A.0.2), the change in velocity of each particle is given in equations (A.0.9)

$$\Delta\mathbf{v}_i = \frac{A}{m}\hat{\mathbf{r}}_{ij} \tag{A.0.9a}$$

$$\Delta\mathbf{v}_j = -\frac{A}{m}\hat{\mathbf{r}}_{ij} \tag{A.0.9b}$$