1495

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

This is a short summary of my work...

# CONTENTS

**Final word count:**

# LIST OF FIGURES

# LIST OF TABLES

# NOMENCLATURE

**Acronyms/Terminology**

| | |
|---|---|
| FCC | Face Centered Cubic, 15 |

# 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 is the ability to predict the thermodynamic and transport properties of fluids.

CHAPTER

**TWO**

# MOLECULAR DYNAMICS

## 2.1 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 [1]

The fundamental identity of Newtonian Mechnanics is Newton's Second Law of Motion (2.1.1). 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.

Only pairwise intermolecular forces are considered ie the total force acting on a particle is the sum of the forces caused by every other particle in the system(2.1.2).

$$\vec{F_i} = \sum_{j \neq i} F_i \tag{2.1.2}$$

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.

### 2.1.1 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. Conservative forces can be calculated from their potential by equation 2.1.3. The gradient of the potential, denoted by $\nabla$ is the partial differential of the potential in each orthogonal direction.

$$\vec{F} = -\nabla U \tag{2.1.3}$$

A very popular potential used in molecular dynamics simulations is the Lennard-Jones potential [2] (equation 2.1.4) as it is simple yet gives comparable results to experimental values.

$$U(r) = 4\epsilon \left[ \left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right] \tag{2.1.4}$$

In equation 2.1.4, $\epsilon$ is the depth of the energy well, while $\sigma$ is the root of the Lennard-Jones potential which corresponds to the change from attractive to repulsive forces (see Lennard Jones Figures), this is taken to be the diameters of the particles during the collision.

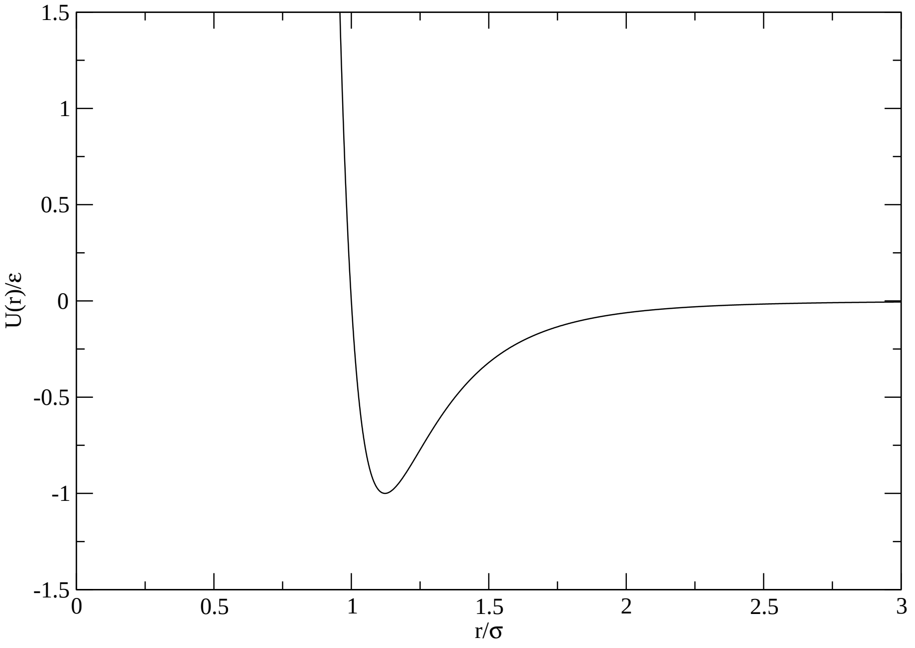

Figure 2.1: Plot of the Lennard-Jones potential

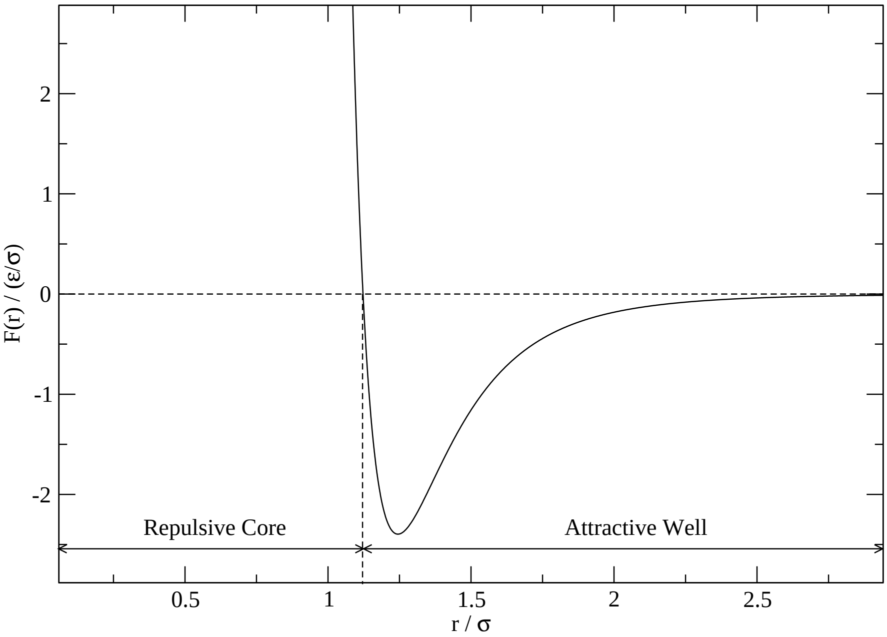

Figure 2.2: Plot of the Lennard-Jones force between a pair of particles

## 2.2 Force-Driven Simulators

### 2.2.1 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 [3] 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, $\Delta t$ 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.

### 2.2.2 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 2.2.1).

$$\vec{F} = m\frac{\partial^2 \vec{r}}{\partial t^2} \tag{2.2.1}$$

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 2.2.2).

$$\vec{r}(t+\Delta t) = r(t)+\frac{\partial \vec{r}(t)}{\partial t}(\Delta t)+\frac{1}{2}\frac{\partial^2 \vec{r}(t)}{\partial t^2}\Delta t^2+\frac{1}{3!}\frac{\partial^3 \vec{r}(t)}{\partial t^3}\Delta t^3+\frac{1}{4!}\frac{\partial^4 \vec{r}(t)}{\partial t^4}\Delta t^4+... \quad (2.2.2)$$

The simplest integrator is Euler's Method which is just the Taylor Series truncated after the acceleration term (equation 2.2.3).

$$\vec{r}(t+\Delta t) = r(t) + \vec{v}(\Delta t) + \frac{1}{2}\vec{a}\Delta t^2 + \mathcal{O}(\Delta t^3) \quad (2.2.3)$$

However this method suffers from large errors and is unstable [4] and is therefore rarely used. The Verlet Integrator [5] improves upon Euler's method by combining the forward timestep with a reverse timestep (2.2.4a). 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 (2.2.4b).

$$\vec{r}(t+\Delta t) = 2\vec{r}(t) - \vec{r}(t-\Delta t) + \vec{a}(t)\Delta t^2 + \mathcal{O}(\Delta t^4) \quad (2.2.4a)$$

$$\vec{v}(t+\Delta t) = \frac{\vec{r}(t+\Delta t) - \vec{r}(t-\Delta t)}{2\Delta t} \quad (2.2.4b)$$

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 [6]. The most common symplectic integrator used in MD is the Velocity Verlet Integrator [7] shown in (2.2.5).

$$\vec{r}(t+\Delta t) = \vec{r}(t) + \vec{v}(t)\Delta t + \frac{1}{2}\vec{a}(t)\Delta t^2 + \mathcal{O}(\Delta t^4) \quad (2.2.5a)$$

$$\vec{v}(t+\Delta t) = \vec{v}(t) + \frac{\vec{a}(t) + \vec{a}(t+\Delta t)}{2}\Delta t \quad (2.2.5b)$$

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 [9],using his 5th order algorithm. The predicted value for the $i^{th}$ time derivative is shown in (2.2.6), and defining $\Delta\vec{a} = \vec{a}^C - \vec{a}^P$, the corrected time derivatives can be calculated using (2.2.7) with coefficients from (2.2.8).

$$\frac{\partial^i}{\partial t^i}\vec{r}^P(t+\Delta t) = \sum_{k=i}^{n} \frac{1}{k!}\frac{\partial^i}{\partial t^i}\vec{r}(t)\Delta t^k \tag{2.2.6}$$

$$\frac{\partial^i}{\partial t^i}\vec{r}^C(t+\Delta t) = \frac{\partial^i}{\partial t^i}\vec{r}^P(t+\Delta t) + \frac{c_i}{\Delta t^i}\left(\frac{\Delta t^2}{2}\Delta\vec{a}\right) \tag{2.2.7}$$

$$c_0 = \frac{3}{16},\ \ c_1 = \frac{251}{360},\ \ c_2 = 1,\ \ c_3 = \frac{11}{18},\ \ c_4 = \frac{1}{6},\ \ c_5 = \frac{1}{60} \tag{2.2.8}$$

The Gear's algorithm, while more accurate at short timesteps than Verlet's integrator [4], suffers at long timesteps and is computationally more expensive.

## 2.3 Event-Driven Simulators

### 2.3.1 Introduction

Though force-driven simulators are more popular the first MD simulation was done using an event-driven simulator by Alder and Wainwright [10]. Event-driven simulators move through time by jumping between successive collisions. These collisions last an infinitesmall amount of time, and deliver an impulse between the particles.

CHAPTER
# THREE

# METHODOLOGY

## 3.1 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.

## 3.2 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

### 3.2.1 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 [3] 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 (3.2.1) in each orthongonal direction.

$$v_i^{new} = v_i^{old} - \frac{1}{N}\sum_{i}^{N} v_i^{old} \tag{3.2.1}$$

### 3.2.2 Running Simulation

The

CHAPTER

# FOUR

# RESULTS

## 4.1 Benchmarking

### 4.1.1 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.

### 4.1.2 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 4.1. 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 [12] 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 4.2

Table 4.1: 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.

| $\rho$ | $t_{avg}$ | | $\langle \hat{\mathbf{r}} \cdot \Delta\mathbf{v} \rangle_{coll}$ | | 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 |

Table 4.2: 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.

| $\rho$ | $\langle T \rangle$ | | $\langle U \rangle$ | | $\langle P \rangle$ | |
|---|---|---|---|---|---|---|
| | 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 |

## 4.2 References

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

[2] 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.

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

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

[5] 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.

[6] 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.

[7] 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.

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

[10] 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.

[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] 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.