OXFORD



Computer Simulation of Liquids

SECOND EDITION

Michael P. Allen & Dominic J. Tildesley

COMPUTER SIMULATION OF LIQUIDS

Computer Simulation of Liquids

Second Edition

Michael P. Allen

Department of Physics, University of Warwick, UK H. H. Wills Physics Laboratory, University of Bristol, UK

Dominic J. Tildesley

Centre Européen de Calcul Atomique et Moléculaire (CECAM), EPFL, Switzerland



OXFORD

UNIVERSITY PRESS

Great Clarendon Street, Oxford, OX2 6DP, United Kingdom

Oxford University Press is a department of the University of Oxford. It furthers the University's objective of excellence in research, scholarship, and education by publishing worldwide. Oxford is a registered trade mark of Oxford University Press in the UK and in certain other countries

© M. P. Allen and D. J. Tildesley 2017

The moral rights of the authors have been asserted

First Edition published in 1987 Second Edition published in 2017

Impression: 1

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior permission in writing of Oxford University Press, or as expressly permitted by law, by licence or under terms agreed with the appropriate reprographics rights organization. Enquiries concerning reproduction outside the scope of the above should be sent to the Rights Department, Oxford University Press, at the address above

> You must not circulate this work in any other form and you must impose this same condition on any acquirer

Published in the United States of America by Oxford University Press 198 Madison Avenue, New York, NY 10016, United States of America

> British Library Cataloguing in Publication Data Data available

Library of Congress Control Number: 2017936745

ISBN 978-0-19-880319-5 (hbk.) ISBN 978-0-19-880320-1 (pbk.)

Printed and bound by CPI Group (UK) Ltd, Croydon, CR0 4YY

Links to third party websites are provided by Oxford in good faith and for information only. Oxford disclaims any responsibility for the materials contained in any third party website referenced in this work. To Diane, Eleanor, Pauline, and Charles

Preface

In the years following the publication of the first edition, we have frequently discussed producing an updated version, and indeed have been nagged on many occasions by colleagues to do so. Despite its increasingly dated content, with quaint references to microfiche, magnetic tapes, and Fortran-77 language examples, the first edition has continued to sell well for three decades. This is quite surprising, bearing in mind the tremendous development of the field and the computer technologies on which it is based. To an extent, the material in our book has been complemented by the publication of other books and online resources which help to understand the underlying principles. Also, it is much easier than it used to be to find technical details in the primary literature, in papers, appendices, and supplementary information. New and improved techniques appear all the time, and the problem is almost that there is too much information, and too much rediscovery of existing methods. The widespread use of simulation packages has provided enormous leverage in this research field. There is much to gain by carefully reading the manual for your chosen package, and we strongly recommend it!

Nonetheless, it remains true that 'getting started' can be a significant barrier, and there is always the need to understand properly what is going on 'under the hood', so as not to use a packaged technique beyond its range of validity. Many colleagues have reaffirmed to us that there is still a need for a general guide book, concentrating on the strengths of the first edition: providing practical advice and examples rather than too much theory. So, we agreed that an updated version of our book would be of value. We intended to produce this many years ago, and it is a sad fact that the demands of academia and industry left too little time to make good on these aspirations. We wish to acknowledge the patience of our editor at Oxford University Press, Sönke Adlung, who has stuck with us over this long period.

Although the field has grown enormously, we resisted the temptation to change the title of the book. It was always focused on the liquid state, and this encompasses what are now known as complex fluids, such as liquid crystals, polymers, some colloidal suspensions, gels, soft matter in general, some biological systems such as fluid membranes, and glasses. The techniques will also be of interest outside the aforementioned fields, and there is no well-defined dividing line, but we try not to stray too far outside our expertise. Rather than give a long list in the title, we hope that 'Computer Simulation of Liquids', interpreted with some latitude, is still sufficiently descriptive.

The content of the book, although structured in the same way as the first edition, has changed to reflect the above expansion in the field, as well as technical advances. The first few chapters cover basic material. Molecular dynamics in various ensembles is now regarded as basic, rather than advanced, and we devote whole chapters to the handling of long-range forces and simulating on parallel computers, both of which are now mainstream topics. There are a few more chapters covering advanced simulation methods, especially those for studying rare events, mesoscale simulations (including coarse graining), and the study of inhomogeneous systems. Instead of concentrating some scientific examples in a single chapter, we have scattered them through the text, to illustrate still further what can be done with the techniques we describe. These examples very much reflect our personal preferences, and we have tried to resist the temptation to turn our book into a collection of scientific or technical reviews, so many otherwise suitable 'highlights' have been omitted. To give a balanced overview of such a huge field would probably be impossible and would certainly have resulted in a very different, and much larger, book. We have dropped material, when methods have been superceded (such as predictor–corrector algorithms), or when they were really of limited or specialized interest (such as the use of integral equations to extend correlation functions to longer distance).

The examples of program code which accompanied the first edition were first provided on microfiche, and later online, courtesy of Cornell University and CCP5. We continue to use such code examples to illustrate ideas in the text, and provide them online. We give the individual filenames, the first few lines of each example, and some guidance on usage, in the book. The full set of codes is available online at

http://www.oup.co.uk/companion/allen_tildesley

Although we stick to Fortran 2008 in the main, some online files are also provided in Python, to widen the accessibility. Some relevant programming considerations may be found in Appendix A.

We wish to reiterate our thanks to those who supported us at the start of our careers (see below) and we have many more people to thank now. J. Anwar, P. Carbone, J. H. Harding, P. A. Madden, S. C. Parker, M. Parrinello, D. Quigley, P. M. Rodger, M. B. Sweatman, A. Troisi, and M. R. Wilson all provided advice and/or encouragement during the early stages of writing. S. Bonella, P. J. Daivis, S. Khalid, P. Malfreyt, B. D. Todd, and R. Vuilleumier advised us on specific topics. G. Ciccotti, A. Humpert, and G. Jackson read and commented on a complete first draft. Any mistakes or misconceptions, naturally, remain our own responsibilities. Our colleagues over the years, at Bristol, Warwick, Southampton, Imperial College London, Unilever plc, and CECAM Lausanne, have also provided a stimulating working environment and a challenging intellectual atmosphere. MPA also wishes to acknowledge helpful study leave periods spent in Germany, at the Universities of Mainz and Bielefeld, and in Australia, at the Universities of Swinburne, Monash, and Deakin. DJT acknowledges an important and stimulating collaboration with the chemistry department at the Université Blaise Pascal, Clermont-Ferrand, France.

Our families have remained an ongoing source of support and inspiration. DJT thanks Eleanor for her unwavering encouragement, while MPA particularly wishes to thank Pauline and Charles, whose holidays frequently had to coincide with conferences and summer schools over the years! *Bristol* MPA

Lausanne August 2016 MPA DJT

From the Preface to the First Edition

This is a 'how-to-do-it' book for people who want to use computers to simulate the behaviour of atomic and molecular liquids. We hope that it will be useful to first-year graduate students, research workers in industry and academia, and to teachers and lecturers who want to use the computer to illustrate the way liquids behave.

Getting started is the main barrier to writing a simulation program. Few people begin their research into liquids by sitting down and composing a program from scratch. Yet these programs are not inherently complicated: there are just a few pitfalls to be avoided. In the past, many simulation programs have been handed down from one research group to another and from one generation of students to the next. Indeed, with a trained eye, it is possible to trace many programs back to one of the handful of groups working in the field 20 years ago. Technical details such as methods for improving the speed of the progam or for avoiding common mistakes are often buried in the appendices of publications or passed on by word of mouth. In the first six chapters of this book, we have tried to gather together these details and to present a clear account of the techniques, namely Monte Carlo and molecular dynamics. The hope is that a graduate student could use these chapters to write his own program.

Both of us were fortunate in that we had expert guidance when starting work in the field, and we would like to take this opportunity to thank P. Schofield (Harwell) and W. B. Streett (Cornell), who set us on the right road some years ago. This book was largely written and created at the Physical Chemistry Laboratory, Oxford, where both of us have spent a large part of our research careers. We owe a great debt of gratitude to the head of department, J. S. Rowlinson, who has provided us with continuous encouragement and support in this venture, as well as a meticulous criticism of early versions of the manuscript. We would also like to thank our friends and colleagues in the physics department at Bristol and the chemistry department at Southampton for their help and encouragement, and we are indebted to many colleagues, who in discussions at conferences and workshops, particularly those organized by CCP5 and CECAM, have helped to form our ideas. We cannot mention all by name but should say that conversations with D. Frenkel and P. A. Madden have been especially helpful. We would also like to thank M. Gillan and J. P. Ryckaert, who made useful comments on certain chapters, and I. R. McDonald who read and commented on the completed manuscript.

Books are not written without a lot of family support. One of us (DJT) wants to thank the Oaks and the Sibleys of Bicester for their hospitality during many weekends over the last three years. Our wives, Diane and Pauline, have suffered in silence during our frequent disappearances, and given us their unflagging support during the whole project. We owe them a great deal. Bristol MPA

Southampton May 1986 MPA DJT

Contents

Intro	oduction	1
1.1	A short history of computer simulation	1
1.2	Computer simulation: motivation and applications	4
1.3	Model systems and interaction potentials	5
1.4	Constructing an intermolecular potential from first principles	25
1.5	Force fields	29
1.6	Studying small systems	35
Stati	stical mechanics	46
2.1	Sampling from ensembles	46
2.2	Common statistical ensembles	52
2.3	Transforming between ensembles	58
2.4	Simple thermodynamic averages	60
2.5	Fluctuations	66
2.6	Structural quantities	69
2.7	Time correlation functions and transport coefficients	73
2.8	Long-range corrections	79
2.9	Quantum corrections	81
2.10	Constraints	83
2.11	Landau free energy	85
2.12	Inhomogeneous systems	86
2.13	Fluid membranes	90
2.14	Liquid crystals	92
Mol	ecular dynamics	95
3.1	Equations of motion for atomic systems	95
3.2	Finite-difference methods	97
3.3	Molecular dynamics of rigid non-spherical bodies	106
3.4	Constraint dynamics	113
3.5	Multiple-timestep algorithms	120
3.6	Checks on accuracy	121
3.7	Molecular dynamics of hard particles	125
3.8	Constant-temperature molecular dynamics	130
3.9	Constant-pressure molecular dynamics	140
3.10	Grand canonical molecular dynamics	144
3.11	Molecular dynamics of polarizable systems	145
Mon	te Carlo methods	147
4.1	Introduction	147
4.2	Monte Carlo integration	147
	Intro 1.1 1.2 1.3 1.4 1.5 1.6 Stati 2.1 2.3 2.4 2.5 2.6 2.7 2.8 2.9 2.10 2.11 2.12 2.13 2.14 Mole 3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 3.9 3.10 3.11 Mon 4.1 4.2	Introduction1.1A short history of computer simulation1.2Computer simulation: motivation and applications1.3Model systems and interaction potentials1.4Constructing an intermolecular potential from first principles1.5Force fields1.6Studying small systemsStatistical mechanics2.1Sampling from ensembles2.2Common statistical ensembles2.3Transforming between ensembles2.4Simple thermodynamic averages2.5Fluctuations2.6Structural quantities2.7Time correlation functions and transport coefficients2.8Long-range corrections2.9Quantum corrections2.11Landau free energy2.12Inhomogeneous systems2.13Fluid membranes2.14Liquid crystalsMolecular dynamics3.1Equations of motion for atomic systems3.2Finite-difference methods3.3Molecular dynamics of rigid non-spherical bodies3.4Constraint dynamics3.5Multiple-timestep algorithms3.6Checks on accuracy3.7Molecular dynamics of polarizable systems3.8Constant-temperature molecular dynamics3.9Constant-temperature molecular dynamics3.10Grand canonical molecular dynamics3.11Molecular dynamics of polarizable systems3.12Constant-temperature molecular dynamics3.13Gr

xii	Contents

	4.3	Importance sampling	151
	4.4	The Metropolis method	155
	4.5	Isothermal–isobaric Monte Carlo	160
	4.6	Grand canonical Monte Carlo	164
	4.7	Semi-grand Monte Carlo	168
	4.8	Molecular liquids	169
	4.9	Parallel tempering	177
	4.10	Other ensembles	183
5	Som	e tricks of the trade	185
	5.1	Introduction	185
	5.2	The heart of the matter	185
	5.3	Neighbour lists	193
	5.4	Non-bonded interactions and multiple timesteps	200
	5.5	When the dust has settled	201
	5.6	Starting up	204
	5.7	Organization of the simulation	210
	5.8	Checks on self-consistency	214
6	Long	g-range forces	216
	6.1	Introduction	216
	6.2	The Ewald sum	217
	6.3	The particle-particle particle-mesh method	224
	6.4	Spherical truncation	231
	6.5	Reaction field	235
	6.6	Fast multipole methods	239
	6.7	The multilevel summation method	243
	6.8	Maxwell equation molecular dynamics	247
	6.9	Long-range potentials in slab geometry	250
	6.10	Which scheme to use?	254
7	Para	llel simulation	258
	7.1	Introduction	258
	7.2	Parallel loops	260
	7.3	Parallel replica exchange	262
	7.4	Parallel domain decomposition	265
	7.5	Parallel constraints	269
8	How	y to analyse the results	271
	8.1	Introduction	271
	8.2	Liquid structure	272
	8.3	Time correlation functions	274
	8.4	Estimating errors	281
	8.5	Correcting the results	289
9	Adv	anced Monte Carlo methods	297
	9.1	Introduction	297
	9.2	Estimation of the free energy	298

	9.3	Smarter Monte Carlo	317
	9.4	Simulation of phase equilibria	333
	9.5	Reactive Monte Carlo	338
10	Rare	event simulation	342
	10.1	Introduction	342
	10.2	Transition state approximation	343
	10.3	Bennett-Chandler approach	345
	10.4	Identifying reaction coordinates and paths	346
	10.5	Transition path sampling	347
	10.6	Forward flux and transition interface sampling	350
	10.7	Conclusions	354
11	Non	equilibrium molecular dynamics	355
	11.1	Introduction	355
	11.2	Spatially oscillating perturbations	357
	11.3	Spatially homogeneous perturbations	361
	11.4	Inhomogeneous systems	370
	11.5	Flow in confined geometry	371
	11.6	Nonequilibrium free-energy measurements	376
	11.7	Practical points	379
	11.8	Conclusions	381
12	Mese	oscale methods	382
	12.1	Introduction	382
	12.2	Langevin and Brownian dynamics	383
	12.3	Brownian dynamics, molecular dynamics, and Monte Carlo	387
	12.4	Dissipative particle dynamics	390
	12.5	Multiparticle collision dynamics	392
	12.6	The lattice-Boltzmann method	394
	12.7	Developing coarse-grained potentials	397
13	Qua	ntum simulations	406
	13.1	Introduction	406
	13.2	Ab-initio molecular dynamics	408
	13.3	Combining quantum and classical force-field simulations	420
	13.4	Path-integral simulations	426
	13.5	Quantum random walk simulations	437
	13.6	Over our horizon	442
14	Inho	mogeneous fluids	446
	14.1	The planar gas–liquid interface	446
	14.2	The gas–liquid interface of a molecular fluid	462
	14.3	The liquid–liquid interface	464
	14.4	The solid–liquid interface	464
	14.5	The liquid drop	469
	14.6	Fluid membranes	475
	14.7	Liquid crystals	479

xiv	Contents
AI V	Contents

Appendi	x A Computers and computer simulation	481
A.1	Computer hardware	481
A.2 Programming languages		
A.3	Fortran programming considerations	483
Appendi	x B Reduced units	487
B.1	Reduced units	487
Appendi	x C Calculation of forces and torques	491
C.1	Introduction	491
C.2	The polymer chain	491
C.3	The molecular fluid with multipoles	494
C.4	The triple-dipole potential	496
C.5	Charged particles using Ewald sum	497
C.6	The Gay–Berne potential	498
C.7	Numerically testing forces and torques	500
Appendi	x D Fourier transforms and series	501
D.1	The Fourier transform	501
D.2	Spatial Fourier transforms and series	502
D.3	The discrete Fourier transform	504
D.4	Numerical Fourier transforms	505
Appendi	x E Random numbers	509
E.1	Random number generators	509
E.2	Uniformly distributed random numbers	510
E.3	Generating non-uniform distributions	511
E.4	Random vectors on the surface of a sphere	514
E.5	Choosing randomly and uniformly from complicated regions	515
E.6	Generating a random permutation	516
Appendi	x F Configurational temperature	517
F.1	Expression for configurational temperature	517
F.2	Implementation details	517
List of A	cronyms	521
List of G	reek Symbols	527
List of R	oman Symbols	529
List of E	xamples	533
List of C	odes	534
Bibliogra	aphy	536
Index		622

-Introduction

1.1 A short history of computer simulation

What is a liquid? As you read this book, you may be mixing up, drinking down, sailing on, or swimming in, a liquid. Liquids flow, although they may be very viscous. They may be transparent or they may scatter light strongly. Liquids may be found in bulk, or in the form of tiny droplets. They may be vaporized or frozen. Life as we know it probably evolved in the liquid phase, and our bodies are kept alive by chemical reactions occurring in liquids. There are many fascinating details of liquid-like behaviour, covering thermodynamics, structure, and motion. Why do liquids behave like this?

The study of the liquid state of matter has a long and rich history, from both the theoretical and experimental standpoints. From early observations of Brownian motion to recent neutron-scattering experiments, experimentalists have worked to improve the understanding of the structure and particle dynamics that characterize liquids. At the same time, theoreticians have tried to construct simple models which explain how liquids behave. In this book, we concentrate exclusively on atomic and molecular models of liquids, and their analysis by computer simulation. For excellent accounts of the current status of liquid science, the reader should consult the standard references (Barker and Henderson, 1976; Rowlinson and Widom, 1982; Barrat and Hansen, 2003; Hansen and McDonald, 2013).

Early models of liquids (Morrell and Hildebrand, 1936) involved the physical manipulation and analysis of the packing of a large number of gelatine balls, representing the molecules; this resulted in a surprisingly good three-dimensional picture of the structure of a liquid, or perhaps a random glass, and later applications of the technique have been described (Bernal and King, 1968). Assemblies of metal ball bearings, kept in motion by mechanical vibration (Pieranski et al., 1978), have been used as models of granular materials and show some analogies with molecular systems (Olafsen and Urbach, 2005). Clearly, the use of large numbers of macroscopic physical objects to represent molecules can be very time-consuming; there are obvious limitations on the types of interactions between them, and the effects of gravity are difficult to eliminate. However, modern research on colloidal suspensions, where the typical particle size lies in the range 1 nm–1000 nm, with the ability to manipulate individual particles and study large-scale collective behaviour, has greatly revitalized the field (Pusey and van Megen, 1986; Ebert et al., 2009; Lekkerkerker and Tuinier, 2011; Bechinger et al., 2013).

[©] M. P. Allen and D. J. Tildesley 2017. Published in 2017 by Oxford University Press.

2 Introduction

The natural extension of this approach is to use a mathematical, rather than a physical, model, and to perform the analysis by computer. It is now over 60 years since the first computer simulation of a liquid was carried out at the Los Alamos National Laboratories in the United States (Metropolis et al., 1953). The Los Alamos computer, called MANIAC, was at that time one of the most powerful available; it is a measure of the continuing rapid advance in computer technology that handheld devices of comparable power are now available to all at modest cost.

Rapid development of computer hardware means that computing power continues to increase at an astonishing rate. Using modern parallel computer architectures, we can expect to enjoy exaflop computing by 2020 (an exaflop is 10¹⁸ floating-point operations per second). This is matched by the enormous increases in data storage available to researchers and the general public. Computer simulations, of the type we describe in this book, are possible on most machines from laptops to continental supercomputers, and we provide an overview of some opportunities with respect to architecture and computing languages, as they relate to the field, in Appendix A.

The very earliest work (Metropolis et al., 1953) laid the foundations of modern Monte Carlo simulation (so-called because of the role that random numbers play in the method). The precise technique employed in this study is still widely used, and is referred to simply as 'Metropolis Monte Carlo'. The original models were highly idealized representations of molecules, such as hard spheres and disks, but, within a few years, Monte Carlo (MC) simulations were carried out on the Lennard-Jones interaction potential (Wood and Parker, 1957) (see Section 1.3). This made it possible to compare data obtained from experiments on, for example, liquid argon, with the computer-generated thermodynamic data derived from a model.

A different technique is required to obtain the dynamic properties of many-particle systems. Molecular dynamics (MD) is the term used to describe the solution of the classical equations of motion (Newton's equations) for a set of molecules. This was first accomplished, for a system of hard spheres, by Alder and Wainwright (1957; 1959). In this case, the particles move at constant velocity between perfectly elastic collisions, and it is possible to solve the dynamic problem without making any approximations, within the limits imposed by machine accuracy. It was several years before a successful attempt was made to solve the equations of motion for a set of Lennard-Jones particles (Rahman, 1964). Here, an approximate, step-by-step procedure is needed, since the forces change continuously as the particles move. Since that time, the properties of the Lennard-Jones model have been thoroughly investigated (Verlet, 1967; 1968; Johnson et al., 1993).

After this initial groundwork on atomic systems, computer simulation developed rapidly. An early attempt to model a diatomic molecular liquid (Harp and Berne, 1968; Berne and Harp, 1970) using molecular dynamics was quickly followed by two ambitious attempts to model liquid water, first by MC (Barker and Watts, 1969), and then by MD (Rahman and Stillinger, 1971). Water remains one of the most interesting and difficult liquids to study by simulation (Morse and Rice, 1982; McCoustra et al., 2009; Lynden-Bell, 2010; Lin et al., 2012). From early studies of small rigid molecules (Barojas et al., 1973) and flexible hydrocarbons (Ryckaert and Bellemans, 1975), simulations have developed to model more complicated systems such as polymers (Binder, 1995), proteins, lipids, nucleic acids, and carbohydrates (Monticelli and Salonen, 2013). Simulations containing half a



Fig. 1.1 The approximate number of articles concerning the computer simulation of condensed phases published in each complete decade. The search was carried out using the Web of Science[®] by searching on Monte Carlo, molecular dynamics, Brownian dynamics, lattice Boltzmann, dynamical density functional theory, Car–Parrinello, QM/MM in both the TITLE and TOPIC search fields.

million atoms have been conducted for 50 million timesteps to study the surface tension of a small liquid droplet (van Giessen and Blokhuis, 2009) and the massive parallel molecular dynamics code, ls1 mardyn, has been used to simulate a trillion Lennard-Jones atoms (Niethammer et al., 2014). It is now possible to follow the folding of a solvated protein using simulations in the microsecond-to-millisecond range (ca. 10^9-10^{12} timesteps) on a special purpose computer (Piana et al., 2014).

The growth of the field of computer simulation over the last 60 years, as evidenced by the number of publications in refereed journals, has been dramatic. In Fig. 1.1, we have attempted to calculate the number of papers published in this field during each complete decade. While bibliometric exercises of this kind will fail to capture some important papers and will often include some unwanted papers in related disciplines, the overall trend in the number of articles is clear.

This is, in part, due to the continuing and substantial increase in computing power, which follows the celebrated Moore's law curve over this period (see Appendix A). It is also due to the application of these methods to a wide range of previously intractable problems in the materials and life sciences. However, it is also, in no small part, due to the ingenuity of its practitioners in extending the early methods to areas such as: the calculation of free energies and phase diagrams (Chapter 9); the simulation of rare events (Chapter 10); the development of nonequilibrium methods for calculating transport coefficients (Chapter 11); the development of coarse-grained methods to extend the length and timescales that can be simulated (Chapter 12); and in the extension to include

4 Introduction

quantum mechanical effects (Chapter 13). This level of activity points to the proposition that computer simulation now sits alongside experiment and theory as a third and equally important tool in modern science. We start by asking: what is a computer simulation? How does it work? What can it tell us?

1.2 Computer simulation: motivation and applications

Some problems in statistical mechanics are exactly soluble. By this, we mean that a complete specification of the microscopic properties of a system (such as the Hamiltonian of an idealized model like the perfect gas or the Einstein crystal) leads directly, and perhaps easily, to a set of useful results or macroscopic properties (such as an equation of state like $PV = Nk_BT$). There are only a handful of non-trivial, exactly soluble problems in statistical mechanics (Baxter, 1982); the two-dimensional Ising model is a famous example.

Some problems in statistical mechanics, while not being exactly soluble, succumb readily to an analysis based on a straightforward approximation scheme. Computers may have an incidental, calculational, part to play in such work; for example, in the evaluation of cluster integrals in the virial expansion for dilute, imperfect gases (Rosenbluth and Rosenbluth, 1954; Wheatley, 2013). The problem is that, like the virial expansion, many 'straightforward' approximation schemes simply do not work when applied to liquids. For some liquid properties, it may not even be clear how to begin constructing an approximate theory in a reasonable way. The more difficult and interesting the problem, the more desirable it becomes to have exact results available, both to test existing approximate methods and to point the way towards new approaches. It is also important to be able to do this without necessarily introducing the additional question of how closely a particular model (which may be very idealized) mimics a real liquid, although this may also be a matter of interest. Computer simulations have a valuable role to play in providing essentially exact results for problems in statistical mechanics which would otherwise only be soluble by approximate methods, or might be quite intractable. In this sense, computer simulation is a test of theories and, historically, simulations have indeed discriminated between well-founded approaches, such as integral equation theories (Hansen and McDonald, 2013), and ideas that are plausible but, in the event, less successful, such as the old cell theories of liquids (Lennard-Jones and Devonshire, 1939a,b). The results of computer simulations may also be compared with those of real experiments. In the first place, this is a test of the underlying model used in a computer simulation. Eventually, if the model is a good one, the simulator hopes to offer insights to the experimentalist, and assist in the interpretation of new results. This dual role of simulation, as a bridge between models and theoretical predictions on the one hand, and between models and experimental results on the other, is illustrated in Fig. 1.2. Because of this connection role, and the way in which simulations are conducted and analysed, these techniques are often termed 'computer experiments'.

Computer simulation provides a direct route from the microscopic details of a system (the masses of the atoms, the interactions between them, molecular geometry, etc.) to macroscopic properties of experimental interest (the equation of state, transport coefficients, structural order parameters, and so on). As well as being of academic interest, this type of information is technologically useful. It may be difficult or impossible to carry out experiments under extremes of temperature and pressure, while a computer simulation



Fig. 1.2 The connection between experiment, theory, and computer simulation.

of the material in, say, a shock wave, a high-temperature plasma, a nuclear reactor, or a planetary core, would be perfectly feasible. Quite subtle details of molecular motion and structure, for example in heterogeneous catalysis, fast ion conduction, or enzyme action, are difficult to probe experimentally but can be extracted readily from a computer simulation. Finally, while the speed of molecular events is itself an experimental difficulty it represents no hindrance to the simulator. A wide range of physical phenomena, from the molecular scale to the galactic (Hockney and Eastwood, 1988), may be studied using some form of computer simulation.

In most of this book, we will be concerned with the details of carrying out simulations (the central box in Fig. 1.2). In the rest of this chapter, however, we deal with the general question of how to put information in (i.e. how to define a model of a liquid) while in Chapter 2 we examine how to get information out (using statistical mechanics).

1.3 Model systems and interaction potentials

1.3.1 Introduction

In most of this book, the microscopic state of a system may be specified in terms of the positions and momenta of a constituent set of particles: the atoms and molecules. Within the Born–Oppenheimer (BO) approximation (see also Chapter 13), it is possible to express the Hamiltonian of a system as a function of the nuclear variables, the (rapid) motion of the electrons having been averaged out. Making the additional approximation that a classical description is adequate, we may write the Hamiltonian \mathcal{H} of a system of N molecules as a sum of kinetic- and potential-energy functions of the set of coordinates \mathbf{q}_i

and momenta \mathbf{p}_i of each molecule *i*. Adopting a condensed notation

$$\mathbf{q} = (\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_N) \tag{1.1a}$$

$$\mathbf{p} = (\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_N) \tag{1.1b}$$

we have

$$\mathcal{H}(\mathbf{q}, \mathbf{p}) = \mathcal{K}(\mathbf{p}) + \mathcal{V}(\mathbf{q}). \tag{1.2}$$

Usually, the Hamiltonian will be equal to the total internal energy E of the system. The generalized coordinates \mathbf{q}_i may simply be the set of Cartesian coordinates \mathbf{r}_i of each atom (or nucleus) in the system, but, as we shall see, it is sometimes useful to treat molecules as rigid bodies, in which case \mathbf{q} will consist of the Cartesian coordinates of each molecular centre of mass together with a set of variables Ω_i that specify molecular orientation. In any case, \mathbf{p} stands for the appropriate set of conjugate momenta. For a simple atomic system, the kinetic energy \mathcal{K} takes the form

$$\mathcal{K} = \sum_{i=1}^{N} \sum_{\alpha} p_{i\alpha}^2 / 2m_i \tag{1.3}$$

where m_i is the molecular mass, and the index α runs over the different (x, y, z) components of the momentum of atom *i*. The potential energy \mathcal{V} contains the interesting information regarding intermolecular interactions: assuming that \mathcal{V} is fairly sensibly behaved, it will be possible to construct, from \mathcal{H} , an equation of motion (in Hamiltonian, Lagrangian, or Newtonian form) which governs the entire time-evolution of the system and all its mechanical properties (Goldstein, 1980). Solution of this equation will generally involve calculating, from \mathcal{V} , the forces f_i and torques τ_i acting on the molecules (see Chapter 3). The Hamiltonian also dictates the equilibrium distribution function for molecular positions and momenta (see Chapter 2). Thus, generally, it is \mathcal{H} (or \mathcal{V}) which is the basic input to a computer simulation program. The approach used almost universally in computer simulation is to separate the potential energy into terms involving pairs, triplets, etc. of molecules. In the following sections we shall consider this in detail.

Recently, there has been a spectacular growth in the number of simulation studies which avoid the use of effective potentials by considering the electrons explicitly using density functional theory (Martin, 2008). In an early approach, the electron density was represented by an extension of the electron gas theory (LeSar and Gordon, 1982; 1983; LeSar, 1984). In most of the current work, the electronic degrees of freedom are explicitly included in the description. The electrons, influenced by the external field of the nuclei, are allowed to evolve during the course of the simulation by an auxiliary set of dynamical equations (Car and Parrinello, 1985). This method, known as *ab initio* molecular dynamics (Marx and Hutter, 2012), is now sufficiently well developed that it may become the method of choice for simulations in materials and the life sciences as the speed of computers increases. We will consider this approach in more detail in Chapter 13.



Fig. 1.3 Argon pair potentials. We illustrate (solid line) a recent pair potential for argon calculated by *ab initio* methods (see Patkowski and Szalewicz, 2010). Also shown is the Lennard-Jones 12–6 potential (dashed line) used in computer simulations of liquid argon.

1.3.2 Atomic systems

Consider first the case of a system containing N atoms. The potential energy may be divided into terms depending on the coordinates of individual atoms, pairs, triplets, etc.:

$$\mathcal{V} = \sum_{i} v_1(\mathbf{r}_i) + \sum_{i} \sum_{j>i} v_2(\mathbf{r}_i, \mathbf{r}_j) + \sum_{i} \sum_{j>i} \sum_{k>j} v_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) + \dots$$
(1.4)

The $\sum_i \sum_{j>i}$ notation indicates a summation over all distinct pairs *i* and *j* without counting any pair twice (i.e. as *ij* and *ji*); the same care must be taken for triplets. The first term in eqn (1.4), $v_1(\mathbf{r}_i)$, represents the effect of an external field (including, e.g. the container walls) on the system. The remaining terms represent particle interactions. The second term, v_2 , the pair potential, is the most important. The pair potential depends only on the magnitude of the pair separation $r_{ij} = |\mathbf{r}_{ij}| = |\mathbf{r}_i - \mathbf{r}_j|$, so it may be written $v_2(r_{ij})$. Figure 1.3 shows one of the more recent estimates for the pair potential between two argon atoms, as a function of separation (Patkowski and Szalewicz, 2010). This potential was determined by fitting to very accurate *ab initio* calculations for the argon dimer. The potential provides a position for the minimum and a well-depth that are very close to the experimental values. It can be used to calculate the spectrum of the isolated argon dimer and it produces a rotational constant and dissociation energy that are in excellent agreement with experiment (Patkowski et al., 2005). In fact, the computed potential is accurate enough to cast some doubt on the recommended, experimental, values of the second virial coefficient of argon at high temperatures (Dymond and Smith, 1980).

The potential shows the typical features of intermolecular interactions. There is an attractive tail at large separations, essentially due to correlation between the electron clouds surrounding the atoms ('van der Waals' or 'London' dispersion). In addition, for

8 Introduction

charged species, Coulombic terms would be present. There is a negative well, responsible for cohesion in condensed phases. Finally, there is a steeply rising repulsive wall at short distances, due to non-bonded overlap between the electron clouds.

The v_3 term in eqn (1.4), involving triplets of molecules, is undoubtedly significant at liquid densities. Estimates of the magnitudes of the leading, triple-dipole, three-body contribution (Axilrod and Teller, 1943) have been made for inert gases in their solid-state face centred cubic (FCC) lattices (Doran and Zucker, 1971; Barker and Henderson, 1976). It is found that up to 10 % of the lattice energy of argon (and more in the case of more polarizable species) may be due to these non-additive terms in the potential; we may expect the same order of magnitude to hold in the liquid phase. Four-body (and higher) terms in eqn (1.4) are expected to be small in comparison with v_2 and v_3 .

Despite the size of three-body terms in the potential, they are only rarely included in computer simulations (Barker et al., 1971; Attard, 1992; Marcelli and Sadus, 2012). This is because, as we shall see shortly, the calculation of any quantity involving a sum over triplets of molecules will be very time-consuming on a computer. In most cases, the pairwise approximation gives a remarkably good description of liquid properties because the average three-body effects can be partially included by defining an 'effective' pair potential. To do this, we rewrite eqn (1.4) in the form

$$\mathcal{V} \approx \sum_{i} v_1(\mathbf{r}_i) + \sum_{i} \sum_{j>i} v_2^{\text{eff}}(r_{ij}).$$
(1.5)

The pair potentials appearing in computer simulations are generally to be regarded as effective pair potentials of this kind, representing all the many-body effects; for simplicity, we will just use the notation $v(r_{ij})$, or v(r). A consequence of this approximation is that the effective pair potential needed to reproduce experimental data may turn out to depend on the density, temperature, etc., while the true two-body potential $v_2(r_{ij})$, of course, does not.

Now we turn to the simpler, more idealized, pair potentials commonly used in computer simulations. These reflect the salient features of real interactions in a general, often empirical, way. Illustrated, with the accurate argon pair potential, in Fig. 1.3 is a simple Lennard-Jones 12–6 potential

$$v^{\mathrm{LJ}}(r) = 4\epsilon \left[(\sigma/r)^{12} - (\sigma/r)^6 \right]$$
(1.6)

which provides a reasonable description of the properties of argon, via computer simulation, if the parameters ϵ and σ are chosen appropriately. The potential has a long-range attractive tail of the form $-1/r^6$, a negative well of depth ϵ , and a steeply rising repulsive wall at distances less than $r \sim \sigma$. The well-depth is often quoted in units of temperature as $\epsilon/k_{\rm B}$, where $k_{\rm B}$ is Boltzmann's constant; values of $\epsilon/k_{\rm B} = 120$ K and $\sigma = 0.34$ nm provide reasonable agreement with the experimental properties of liquid argon. Once again, we must emphasize that these are not the values which would apply to an isolated pair of argon atoms, as is clear from Fig. 1.3.

For the purposes of investigating general properties of liquids, and for comparison with theory, highly idealized pair potentials may be of value. In Fig. 1.4, we illustrate three



Model systems and interaction potentials 9

Fig. 1.4 Idealized pair potentials. (a) The hard-sphere potential; (b) the square-well potential; (c) The soft-sphere potential with repulsion parameter v = 1; (d) The soft-sphere potential with repulsion parameter v = 1?. Vertical and horizontal scales are arbitrary.

forms which, although unrealistic, are very simple and convenient to use in computer simulation and in liquid-state theory. These are: the hard-sphere potential

$$v^{\rm HS}(r) = \begin{cases} \infty & \text{if } r < \sigma \\ 0 & \text{if } \sigma \le r; \end{cases}$$
(1.7)

the square-well potential

$$v^{\text{SW}}(r) = \begin{cases} \infty & \text{if } r < \sigma_1 \\ -\epsilon, & \text{if } \sigma_1 \le r < \sigma_2 \\ 0, & \text{if } \sigma_2 \le r; \end{cases}$$
(1.8)

and the soft-sphere potential

$$v^{\rm SS}(r) = \epsilon (\sigma/r)^{\nu} = ar^{-\nu}, \qquad (1.9)$$

where v is a parameter, often chosen to be an integer. The soft-sphere potential becomes progressively 'harder' as v is increased. Soft-sphere potentials contain no attractive part. It is often useful to divide more realistic potentials into separate attractive and repulsive



Fig. 1.5 The separation of the Lennard-Jones potential v^{LJ} into attractive and repulsive components, v^{ALJ} and v^{RLJ} , respectively. The vertical dashed line shows the position of r_{min} .

components, and the separation proposed by Weeks et al. (1971) involves splitting the potential at the minimum. For the Lennard-Jones potential, the repulsive and attractive parts are, as illustrated in Fig. 1.5,

$$v^{\text{RLJ}}(r) = \begin{cases} v^{\text{LJ}}(r) + \epsilon & \text{if } r < r_{\min} \\ 0 & \text{if } r_{\min} \le r \end{cases}$$
(1.10a)

$$v^{\text{ALJ}}(r) = \begin{cases} -\epsilon & \text{if } r < r_{\min} \\ v^{\text{LJ}}(r) & \text{if } r_{\min} \le r, \end{cases}$$
(1.10b)

where $r_{\min} = 2^{1/6} \sigma \approx 1.12 \sigma$. In perturbation theory (Weeks et al., 1971), a hypothetical fluid of molecules interacting via the repulsive potential v^{RLJ} is treated as a reference system and the attractive part v^{ALJ} is the perturbation. It should be noted that the potential v^{RLJ} is significantly harder than the inverse twelfth power soft-sphere potential, which is also sometimes thought of as the 'repulsive' part of $v^{\text{LJ}}(r)$.

For ions, of course, these potentials are not sufficient to represent the long-range interactions. A simple approach is to supplement one of these pair potentials with the Coulomb charge-charge interaction

$$v^{qq}(r_{ij}) = \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \tag{1.11}$$

where q_i , q_j are the charges on ions *i* and *j* and ϵ_0 is the permittivity of free space (not to be confused with ϵ in eqns (1.6)–(1.10)). For ionic systems, induction interactions are important: the ionic charge induces a dipole on a neighbouring ion. This term is not pairwise additive and hence is difficult to include in a simulation. The shell model is a crude attempt to account for this polarizability (Dixon and Sangster, 1976; Lindan, 1995).

Each ion is represented as a core surrounded by a shell. Part of the ionic charge is located on the shell and the rest in the core. This division is always arranged so that the shell charge is negative (it represents the electronic cloud). The interactions between ions are just sums of the Coulombic shell–shell, core–core, and shell–core contributions. The shell and core of a given ion are coupled by a harmonic spring potential. The shells are taken to have zero mass. During a simulation, their positions are adjusted iteratively to zero the net force acting on each shell: this process makes the simulations expensive. We shall return to the simulation of polarizable systems in Section 1.3.3.

When a potential depends upon just a few parameters, such as ϵ and σ , it may be possible to choose an appropriate set of units in which these parameters take values of unity. This results in a simpler description of the properties of the model, and there may also be technical advantages within a simulation program. For Coulomb systems, the factor $4\pi\epsilon_0$ in eqn (1.11) is often omitted, and this corresponds to choosing a non-standard unit of charge. We discuss such reduced units in Appendix B. Reduced densities, temperatures, etc. are often denoted by an asterisk, that is, ρ^* , T^* etc.

1.3.3 Molecular systems

In principle, there is no reason to abandon the atomic approach when dealing with molecular systems: chemical bonds are simply interatomic potential-energy terms (Chandler, 1982). Ideally, we would like to treat all aspects of chemical bonding, including the reactions which form and break bonds, in a proper quantum mechanical fashion. This difficult task has not yet been accomplished but there are two common simplifying approaches. We might treat the bonds as classical harmonic springs (or Morse oscillators) or we could treat the molecule as a rigid or semi-rigid unit, with fixed bond lengths and, sometimes, fixed bond angles and torsion angles.

Bond vibrations are of very high frequency (and hence difficult to handle, certainly in a classical simulation). It quite possible that a high-frequency vibration will not be in thermal equilibrium with the fluid that surrounds it. These vibrations are also of low amplitude (and are therefore unimportant for many liquid properties). For these reasons, we prefer the approach of constraining the bond lengths to their equilibrium values. Thus, a diatomic molecule with a strongly binding interatomic potential-energy surface might be replaced by a dumb-bell with a rigid interatomic bond.

The interaction between the nuclei and electronic charge clouds of a pair of molecules *i* and *j* is clearly a complicated function of relative positions \mathbf{r}_i , \mathbf{r}_j and orientations Ω_i , Ω_j (Gray and Gubbins, 1984). One way of modelling a molecule is to concentrate on the positions and sizes of the constituent atoms (Eyring, 1932). The much simplified 'atomatom' or 'site-site' approximation for diatomic molecules is illustrated in Fig. 1.6. The total interaction is a sum of pairwise contributions from distinct sites *a* in molecule *i*, at position \mathbf{r}_{ia} , and *b* in molecule *j*, at position \mathbf{r}_{jb} :

$$v(\mathbf{r}_{ij}, \,\Omega_i, \,\Omega_j) = \sum_a \sum_b v_{ab}(r_{ab}). \tag{1.12}$$

Here *a*, *b* take the values 1, 2, v_{ab} is the pair potential acting between sites *a* and *b*, and r_{ab} is shorthand for the inter-site separation $r_{ab} = |\mathbf{r}_{ab}| = |\mathbf{r}_{ia} - \mathbf{r}_{jb}|$. The interaction sites are usually centred, more or less, on the positions of the nuclei in the real molecule, so as to



Fig. 1.6 The atom-atom model of a diatomic molecule. The total interaction is a sum of terms involving the distances $|\mathbf{r}_{ia} - \mathbf{r}_{jb}|$, indicated by dashed lines.

represent the basic effects of molecular 'shape'. A very simple extension of the hard-sphere model is to consider a diatomic composed of two hard spheres fused together (Streett and Tildesley, 1976), but more realistic models involve continuous potentials. Thus, nitrogen, fluorine, chlorine, etc. have been depicted as two 'Lennard-Jones atoms' separated by a fixed bond length (Barojas et al., 1973; Cheung and Powles, 1975; Singer et al., 1977).

The description of the molecular charge distribution may be improved somewhat by incorporating point multipole moments at the centre of charge (Streett and Tildesley, 1977). These multipoles may be equal to the known (isolated molecule) values, or may be 'effective' values chosen simply to yield a better description of the liquid structure and thermodynamic properties. A useful collection of the values of multipole moments is given in Gray and Gubbins (1984). Price et al. (1984) have developed an efficient way of calculating the multipolar energy, forces and torques between molecules of arbitrary symmetry up to terms of $O(r_{ij}^{-5})$. However, it is now generally accepted that such a multipole expansion of the electrostatic potential based around the centre of mass of a molecule is not rapidly convergent.

A pragmatic alternative approach, for ionic and polar systems, is to use a set of fictitious 'partial charges' distributed 'in a physically reasonable way' around the molecule so as to reproduce the known multipole moments (Murthy et al., 1983). For example, the electrostatic part of the interaction between nitrogen molecules may be modelled using five partial charges placed along the axis, while for methane, a tetrahedral arrangement of partial charges is appropriate. These are illustrated in Fig. 1.7. For the case of N₂, taking the molecular axis to lie along *z*, the quadrupole moment *Q* is given by (Gray and Gubbins, 1984)

$$Q = \sum_{a=1}^{5} q_a z_a^2 \tag{1.13}$$



Fig. 1.7 Partial charge models: (a) A five-charge model for N₂. There is one charge at the bond centre, two at the positions of the nuclei, and two more displaced beyond the nuclei. Typical values (with $e = 1.602 \times 10^{-19}$ C): q = +5.2366 e, q' = -4.0469 e, giving $Q = -4.67 \times 10^{-40}$ C m² (Murthy et al., 1983). (b) A five-charge model for CH₄. There is one charge at the centre and four others at the positions of the hydrogen nuclei. Typical values are CH bond length 0.1094 nm, q = 0.143 e giving $O = 5.77 \times 10^{-50}$ C m³ (Righini et al., 1981).

with similar expressions for the higher multipoles (all the odd ones vanish for N_2). The first non-vanishing moment for methane is the octopole O

$$O = \frac{5}{2} \sum_{a=1}^{5} q_a x_a y_a z_a \tag{1.14}$$

in a coordinate system aligned with the cube shown in Fig. 1.7. The aim of all these approaches is to approximate the complete charge distribution in the molecule. In a calculation of the potential energy, the interaction between partial charges on different molecules would be summed in the same way as the other site–site interactions.

The use of higher-order multipoles has enjoyed a renaissance in recent years. This is because we can obtain an accurate representation of the electrostatic potential by placing multipoles at various sites within the molecule. These sites could be at the atom positions, or at the centres of bonds or within lone pairs, and it is normally sufficient to place a charge, dipole and quadrupole at any particular site. This approach, known as a distributed multipole analysis (Stone, 1981; 2013, Chapter 7), is illustrated for N_2 and CO in Fig. 1.8. In the case of N₂ the multipoles are placed at the centre of the bond and on the two nitrogen atoms, with their z-axis along the bond. Each site has a charge and a quadrupole and, in addition, the two atoms have equal and opposite dipoles. These are calculated using an accurate density functional theory B3LYP (Martin, 2008). In atomic units (see Appendix B), the overall quadrupole of the molecule calculated from this distribution is $-1.170 ea_0^2$ corresponding to the experimental estimate of $(-1.09 \pm 0.07) ea_0^2$. A similar calculation for CO produces charges, dipoles and quadrupoles on all three sites (the C and O atoms and the centre of the bond). The overall dipole and quadrupole moments from this distribution are $0.036 ea_0$ and $-1.515 ea_0^2$ respectively, compared with the experimental estimates of $0.043 ea_0$ and $-1.4 ea_0^2$. The electrostatic energy between two molecules is now the sum of the multipole interactions between the atoms or sites in different molecules. The energy of interaction between two sets of distributed multipoles $\{q_a, \mu_a, Q_a\}$ and $\{q_b, \mu_b, Q_b\}$,

	1.0337 1.0337 ← → ← →			1.066 1.066 ← → ← →		
	• N	-0	 N	C	O	• 0
q	0.427	-0.854	0.427	0.556	-0.832	0.276
μ_z	0.947	0.0	-0.947	1.159	-0.030	-0.796
Q_{zz}	0.775	0.283	0.775	0.377	0.274	1.068

Fig. 1.8 The distributed multipoles required to represent the electrostatic potential of a N₂ and CO molecule calculated using a cc-p-VQZ basis set. Multipoles are placed at the positions of the atoms (black circles) and at the midpoint of the bond (white circles). The distances are in atomic units, $a_0 = 0.529$ Å; charges, q, are in units of $e = 1.602 \times 10^{-19}$ C; dipoles, μ , are in units of $ea_0 = 8.478 \times 10^{-30}$ C m; and quadrupoles, Q, are in units of $ea_0^2 = 4.487 \times 10^{-40}$ C m² (see Appendix B). Data from Stone (2013).

on atoms *a* and *b* at \mathbf{r}_a and \mathbf{r}_b , is given by

$$v_{ab}^{elec} = Tq_a q_b + T_{\alpha} (\mu_{a\alpha} q_b - q_a \mu_{b\alpha}) + T_{\alpha\beta} (\frac{1}{3} q_a Q_{b\alpha\beta} - \mu_{a\alpha} \mu_{b\beta} + \frac{1}{3} Q_{a\alpha\beta} q_b) + \frac{1}{3} T_{\alpha\beta\gamma} (\mu_{a\alpha} Q_{b\beta\gamma} - Q_{a\alpha\beta} \mu_{b\gamma}) + \frac{1}{9} T_{\alpha\beta\gamma\delta} Q_{a\alpha\beta} Q_{b\gamma\delta}$$
(1.15)

where we take the sum over repeated Cartesian indices α , β etc. The interaction or '*T*' tensors are given by

$$T_{\alpha,\beta\ldots\gamma} = (-1)^n \nabla_\alpha \nabla_\beta \ldots \nabla_\gamma \frac{1}{r_{ab}},$$
(1.16)

where n is the order of the tensor. Thus

$$T = \frac{1}{r_{ab}}, \quad T_{\alpha} = \frac{(r_{ab})_{\alpha}}{r_{ab}^3}, \quad T_{\alpha\beta} = \frac{3(r_{ab})_{\alpha}(r_{ab})_{\beta} - r_{ab}^2 \delta_{\alpha\beta}}{r_{ab}^5}, \quad (1.17)$$

and so on. Note that the *T* tensors are defined for $\mathbf{r}_{ab} = \mathbf{r}_a - \mathbf{r}_b$. This is a useful formulation of the electrostatic energy for a computer simulation where the *T* tensors are readily expressed in terms of the Cartesian coordinates of the atoms. In addition, it is also straightforward to evaluate the derivative of the potential to obtain the force (the field) or the field gradient. The electrostatic potential, ϕ , at a distance *r* from a charge *q* is $\phi(r) = q/r$ and the corresponding electric field is $\mathcal{E} = -\nabla \phi(r)$. The field is simply the force per unit charge. For example the field (\mathcal{E}) and field gradient (\mathcal{E}') arising from a charge q_b at *b* are

$$\mathcal{E}_{\alpha} = -\nabla_{\alpha} q_b T = q_b T_{\alpha}$$

$$\mathcal{E}'_{\alpha\beta} = -\nabla_{\alpha} \nabla_{\beta} q_b T = -q_b T_{\alpha\beta}.$$
 (1.18)

The quadrupole tensor used in eqn (1.15) is defined to be traceless,

$$Q_{\alpha\beta} = \sum_{a} q_a \left(\frac{3}{2} (r_a)_{\alpha} (r_b)_{\beta} - \frac{1}{2} r_a^2 \delta_{\alpha\beta} \right).$$
(1.19)

Code 1.1 Calculation of T tensors

This file is provided online. For a pair of linear molecules, electrostatic energies and forces are calculated using both the angles between the various vectors, and the *T* tensors.

```
! t_tensor.f90
! Electrostatic interactions: T-tensors compared with angles
PROGRAM t_tensor
```

The components of the dipole and quadrupole will initially be defined in an atom-fixed axis frame centred on an atom (or site) and at any given point in a simulation it will be necessary to transform these properties to the space-fixed axis system for use in eqn (1.15) (Dykstra, 1988; Ponder et al., 2010). This can be simply achieved with a rotation matrix which we discuss in Section 3.3.1. An example of the calculation of T tensors is given in Code 1.1.

Electronic polarization refers to the distortion of the electronic charge cloud by the electrostatic field from the other molecules. In a molecular fluid it can be an important contribution to the energy. It is inherently a many-body potential and unlike many of the interactions already discussed in this chapter, it cannot be broken down to a sum over pair interactions. For this reason, it is expensive to calculate and was often omitted from earlier simulations. In these cases, some compensation was obtained by enhancing the permanent electrostatic interactions in the model. For example, in early simulations of water, the overall permanent dipole of the molecule was set to ca. 2.2 D rather than the gas-phase value 1.85 D (where $1 D = 0.299 79 \times 10^{-30} C m$) in order to fit to the condensed phase properties in the absence of polarization (Watanabe and Klein, 1989). Nevertheless, polarization can be included explicitly in a model and there are three common approaches: the induced point multipole model; the fluctuating charge model; and the Drude oscillator model (Antila and Salonen, 2013; Rick and Stuart, 2003).

The induced multipole approach (Applequist et al., 1972) is based on a knowledge of the atomic dipole polarizability, $\alpha^a_{\alpha\beta}$, on a particular atom *a*. Consider a molecule containing a set of charges, q_a , on each atom. The induced dipole at *a* contains two terms

$$\Delta \mu_{\gamma}^{a} = \alpha_{\alpha\gamma}^{a} \Big(\mathcal{E}_{\alpha}^{a} + \sum_{b \neq a} T_{\alpha\beta} \Delta \mu_{\beta}^{b} \Big), \tag{1.20}$$

where we sum over repeated indices α , β . The first term in the field, \mathcal{E} , comes from the permanent charges at the other atoms and the second term comes from the dipoles that have been induced at these atoms. We ignore contributions from the field gradient at atom *a* by setting the higher-order polarizabilities to zero. Eqn (1.20) can be formally solved for the induced dipoles

$$\Delta \mu^a_{\alpha} = \sum_{b\beta} A_{\alpha\beta} \mathcal{E}^b_{\beta} \tag{1.21}$$

Example 1.1 Water, water everywhere

The earliest simulations of molecular liquids focused on water (Barker and Watts, 1969; Rahman and Stillinger, 1971) and, since then, there have been over 80 000 published simulations of the liquid. Considerable effort and ingenuity have gone into developing models of the intermolecular potential between water molecules. There are three types of classical potential models in use: rigid, flexible, and polarizable. The simplest rigid models use a single Lennard-Jones site to represent the oxygen atom and three partial charges: at the centre of the oxygen and the position of the hydrogen atoms. There are no specific dispersion interactions involving the H atoms and the charges are set to model the effective condensed-phase dipole moment of water, 2.2 D-2.35 D. Examples include the the SPC and the SPC/E models (Berendsen et al., 1981; 1987) used in the GROMOS force field, and the TIP3P model (Jorgensen et al., 1983) implemented in AMBER and CHARMM. The precise geometry and the size of the charges are different in each of these models. They predict the experimental liquid densities at a fixed pressure but tend to overestimate the diffusivity. The addition of a fourth negative charge along the bisector of the H–O–H bond creates the TIP4P model (Jorgensen et al., 1983) and its generalization TIP4P/2005 (Abascal and Vega, 2005). These models are capable of producing many of the qualitative features of the complicated water phase diagram. The TIP5P potential model (Mahoney and Jorgensen, 2000) supplements the three charges on the atoms with two negative charges at the position of the lone pairs. This model correctly predicts the density maximum near 4 °C at 1 bar, and the liquid structure obtained from diffraction experiments. Flexibility can be included in models such as SPC/E using the intramolecular potential of Toukan and Rahman (1985), in which anharmonic oscillators are used to represent the O-H and H-H stretches. These flexible models predict many of the features of

A recent study by Shvab and Sadus (2013) indicates that rigid models underestimate the water structure and H-bond network at temperatures higher than 400 K and that none of the models so far discussed can predict the heat capacities or thermal expansion coefficients of the liquid. To improve on this position it is necessary to include polarization in the potential. Li et al. (2007a) show that the Matsuoka-Clementi-Yoshimine potential fitted from quantum calculations can be adapted to include three-body dispersion interactions for O atoms and fluctuating charges to create the more accurate MCYna model (Shvab and Sadus, 2013). These enhancements produce good agreement with experimental data over the entire liquid range of temperatures. Jones et al. (2013) have taken a different approach by embedding a quantum Drude oscillator (QDO) and using adiabatic path-integral molecular dynamics to simulate 4000 water molecules. Sokhan et al. (2015) show that this approach can produce accurate densities, surface tensions, and structure over a range of temperatures. Models of water in terms of pseudo-potentials to describe the nuclei and core electrons, and a model of the exchange correlation function to describe the non-classical electron repulsion between the valence electrons will be described in Chapter 13.

the vibrational spectrum of the liquid (Praprotnik et al., 2005).

where the relay matrix $\mathbf{A} = \mathbf{B}^{-1}$ and

$$B_{\alpha\beta} = \begin{cases} (\alpha^a)^{-1} & \text{if } a = b \\ -T_{\alpha\beta} & \text{if } a \neq b. \end{cases}$$
(1.22)

Here **A** and **B** have dimensions of the number of sites involved in the polarization; this can be a large matrix, so practically eqn (1.20) is solved in a simulation by iterating the induced dipoles until convergence is achieved (Warshel and Levitt, 1976). This method can also be used with the distributed multipole analysis where the field at a polarizable atom might contain terms from the charge, dipole and quadrupole at a neighbouring atom, while the induction still occurs through the dipole polarizability (Ponder et al., 2010).

There is a well-known problem with these point polarizability models in which the elements of **A** diverge at short separations: the so-called polarization catastrophe. This is caused by the normal breakdown in the multipole expansion at these distances. It can be mitigated by smearing the charges on a particular site (Thole, 1981). The effect of this modification is to change the interaction tensor to

$$\tilde{T}_{\alpha\beta} = \frac{3f_t(r_{ab})_\alpha(r_{ab})_\beta - f_e r_{ab}^2 \delta_{\alpha\beta}}{4\pi\epsilon_0 r_{ab}^5}$$
(1.23)

where f_e and f_t are two, simple, damping functions. A useful discussion of the various possible choices for these damping functions is given by Stone (2013). The modified tensor $\tilde{T}_{\alpha\beta}$ can now be used in eqn (1.20) to calculate the induced moments. Once the induced dipole at atom *a* has been consistently determined then the induction energy associated with that atom is

$$v_a^{\text{ind}} = -\frac{1}{2} \mathcal{E}_{\alpha}^a \Delta \mu_{\alpha}^a. \tag{1.24}$$

The second method of including polarization in a model is the fluctuating charge model, sometimes referred to as the electronegativity equalization model. The partial charges are allowed to fluctuate as dynamical quantities. We can illustrate this approach by considering a model for water (Sprik, 1991). In addition to the three permanent charges normally used to represent the electrostatic moments, four additional fluctuating charges are disposed in a tetrahedron around the central oxygen atom (see Fig. 1.9). The magnitudes of the charges $q_i(t)$ fluctuate in time, but they preserve overall charge neutrality

$$\sum_{i=1}^{4} q_i(t) = 0 \tag{1.25}$$

and they produce an induced dipole

$$\Delta \boldsymbol{\mu} = \sum_{i=1}^{4} q_i(t) \mathbf{r}_i \tag{1.26}$$

where \mathbf{r}_i are the vectors describing the positions of the tetrahedral charges with respect to the O atom. If $|\mathbf{r}_i| \ll r_{\text{OH}}$ then the higher moments of the fluctuating charge distribution can be neglected. The potential energy from the four charges is the sum of the electrostatic



Fig. 1.9 A polarizable model for water (Sprik, 1991). The oxygen nucleus, O, is at the centre of the small tetrahedron. The three permanent charges +q, +q and -2q, at the filled black spheres, are arranged to model the permanent electrostatic potential of water. The four fluctuating charges, $q_i(t)$, located at the white spheres, respond to the surrounding field and can be used to model the polarization of the molecule.

energy $(-\Delta \boldsymbol{\mu} \cdot \boldsymbol{\mathcal{E}})$ and a self energy term $(\Delta \mu^2/2\alpha_{\rm O})$ where $\alpha_{\rm O}$ is the dipole polarizability associated with the oxygen atom. The fluctuating charges in this model can be determined by minimizing the potential energy in a given configuration subject to the constraint of charge neutrality, eqn (1.25),

$$\frac{\partial}{\partial q_i} \left(\frac{\left| \sum_{i=1}^4 q_i \mathbf{r}_i \right|^2}{2\alpha_0} - \sum_{i=1}^4 q_i \mathbf{r}_i \cdot \mathcal{E} \right) = 0.$$
(1.27)

This approach can be extended to more complicated molecules by adding the appropriate number of fluctuating charges; and simplified to study spherical ions by including just two fluctuating charges within the spherical core. In these models, the fluctuating charges, q_i , are a crude representation of the electronic charge density and these can be usefully replaced by more realistic Gaussian charge distributions of width σ

$$\rho_i(\mathbf{r}) = q_i \left(\frac{1}{2\pi\sigma^2}\right)^{3/2} \exp\left(-\frac{\left|\mathbf{r} - \mathbf{r}_i\right|^2}{2\sigma^2}\right).$$
(1.28)

This improves the description of the polarization, particularly at short intermolecular separations (Sprik and Klein, 1988). We note that these models can be readily included in a molecular dynamics simulation by setting up separate equations of motions for the fluctuating charges (Sprik and Klein, 1988; Rick et al., 1994), and we shall consider this approach in Section 3.11.

The third approach is the Drude oscillator model or shell model. A polarizable site is represented as a heavy core particle of charge q_d and a massless or light shell particle of charge $-q_d$. These two particles are connected by a harmonic spring with a spring constant *k*. The minimum in the spring potential is obtained when the core and shell are coincident. The small charge q_d is in addition to the permanent charge at a particular site. The shell and core can separate to produce an induced dipole moment

$$\Delta \boldsymbol{\mu} = -q_{\rm d} \Delta \mathbf{r} \tag{1.29}$$

where $\Delta \mathbf{r}$ is the vector from the core to the shell. The repulsion–dispersion interactions associated with a particular site are normally centred on the shell part of the site.

In the adiabatic implementation the shell is massless and at each step of a simulation the positions of the shells are adjusted iteratively to achieve the minimum energy configuration. In the dynamic model the shells are given a low mass (0.5 u) and an extended Lagrangian approach is used to solve the dynamics for short timesteps. In this case the shell particles are coupled to a heat bath at a low temperature (see Section 3.11). For these models, the atomic polarizability is isotropic and given by $\alpha^a = q_d^2/k$. Procedures are available for parameterizing the shell models to produce the correct molecular polarizabilities and electrostatic moments (Anisimov et al., 2005).

The model for water, shown in Fig. 1.9, begs the question as to whether we need to use a separate intermolecular potential to represent the hydrogen bond between two molecules. The hydrogen bond, between an H atom in one molecule and a strongly electronegative atom in another, is part permanent electrostatic interaction, part induced interaction, and some charge transfer. The evidence as reviewed by Stone (2013) indicates that the attractive electrostatic interaction is the most important term in determining the structure of the hydrogen-bonded dimer but that induced interactions will make an important contribution in condensed phases. It should be possible to avoid a separate hydrogen-bond potential by including an accurate representation of the electrostatic interactions (by using, for example, the distributed multipole approach) and by including polarization.

For larger molecules it may not be reasonable to 'fix' all the internal degrees of freedom. In particular, torsional motion about bonds, which gives rise to conformational interconversion in, for example, alkanes, cannot in general be neglected (since these motions involve energy changes comparable with normal thermal energies). An early simulation of n-butane, CH₃CH₂CH₂CH₃ (Ryckaert and Bellemans, 1975; Maréchal and Ryckaert, 1983), provides a good example of the way in which these features are incorporated in a simple model. Butane can be represented as a four-centre molecule, with fixed bond lengths and bond-bending angles, derived from known experimental (structural) data (see Fig. 1.10). A very common simplifying feature is built into this model: whole groups of atoms, such as CH₃ and CH₂, are condensed into spherically symmetric effective 'united atoms'. In fact, for butane, the interactions between such groups may be represented quite well by the ubiquitous Lennard-Jones potential, with empirically chosen parameters. In a simulation, the $C_1 - C_2$, $C_2 - C_3$ and $C_3 - C_4$ bond lengths are held fixed by a method of constraints, which will be described in detail in Chapter 3. The angles θ and θ' may be fixed by additionally constraining the $C_1 - C_3$ and $C_2 - C_4$ distances; that is, by introducing 'phantom bonds'. If this is done, just one internal degree of freedom, namely the rotation about the C_2-C_3 bond, measured by the angle ϕ , is left unconstrained; for each molecule, an extra term in the potential energy, $v^{\text{torsion}}(\phi)$, appears in the Hamiltonian. This potential would have a minimum at a value of ϕ corresponding to the *trans* conformer of butane, and secondary minima at the gauche conformations. It is easy to see how this approach



Fig. 1.10 (a) Geometry of a model of butane defining bending angles θ , θ' and the torsional angle ϕ (Ryckaert and Bellemans, 1975). (b) The torsional potential, in the AUA(2) model of Padilla and Toxvaerd (1991) as reviewed in Dysthe et al. (2000).

may be extended to much larger flexible molecules. The consequences of constraining bond lengths and angles will be treated in more detail in Chapters 2 and 4.

As the molecular model becomes more complicated, so too do the expressions for the potential energy, forces, and torques, due to molecular interactions. In Appendix C, we give some examples of these formulae, for rigid and flexible molecules, interacting via site–site pairwise potentials, including multipolar terms. We also show how to derive the forces from a simple three-body potential.

1.3.4 Coarse-grained potential models

Coarse graining a potential involves avoiding the full atomic representation of the molecules to find a description of the interaction at a longer or coarser length scale. We have already seen one simple example of this in the use of a united-atom potential for the methylene and methyl groups in butane. Coarse graining will reduce the number of explicit pairs that are needed for the calculation of the energy and force for a particular system and will reduce the computer time or, alternatively, allow us to study a much larger system. Normally an increase in the characteristic length scale in the model goes hand in hand with an increase in the timestep that we can use in a dynamical simulation of the problem. Coarse graining will allow us to use a longer timestep and to cover more 'real' time in our simulation.

One flavour of coarse-grained model has been widely used to study liquid crystalline systems, exhibiting some long-range orientational order. For example, for the nematogen quinquaphenyl, a large rigid molecule that forms a nematic phase, a substantial number of sites would be required to model the repulsive core. A crude model, which represented each of the five benzene rings as a single Lennard-Jones site, would necessitate 25 site–site interactions between each pair of molecules; sites based on each carbon atom would be more realistic but require 900 site–site interactions per pair. An alternative coarse-grained representation of intermolecular potential, introduced by Corner (1948), involves a single

site-site interaction between a pair of molecules, characterized by energy and length parameters that depend on the relative orientation of the molecules.

A version of this family of molecular potentials that has been used in computer simulation studies is the Gay–Berne potential (Gay and Berne, 1981). This is an extension of the Gaussian overlap model generalized to a Lennard-Jones form (Berne and Pechukas, 1972). The basic potential acting between two linear molecules is

$$v^{\text{GB}}(\mathbf{r}_{ij}, \hat{\mathbf{e}}_i, \hat{\mathbf{e}}_j) = 4\epsilon(\hat{\mathbf{r}}, \hat{\mathbf{e}}_i, \hat{\mathbf{e}}_j) \Big[(\sigma_{\text{s}}/\rho_{ij})^{12} - (\sigma_{\text{s}}/\rho_{ij})^6 \Big],$$
(1.30a)

where
$$\rho_{ij} = r_{ij} - \sigma(\hat{\mathbf{r}}, \hat{\mathbf{e}}_i, \hat{\mathbf{e}}_j) + \sigma_{\mathrm{s}}.$$
 (1.30b)

Here, r_{ij} is the distance between the centres of *i* and *j*, and $\hat{\mathbf{r}} = \mathbf{r}_{ij}/r_{ij}$ is the unit vector along \mathbf{r}_{ij} , while $\hat{\mathbf{e}}_i$ and $\hat{\mathbf{e}}_j$ are unit vectors along the axis of the molecules. The molecule can be considered (approximately) as an ellipsoid characterized by two diameters σ_s and σ_e , the separations at which the side-by-side potential, and the end-to-end potential, respectively, become zero. Thus

$$\sigma(\hat{\mathbf{r}}, \hat{\mathbf{e}}_i, \hat{\mathbf{e}}_j) = \sigma_s \left[1 - \frac{\chi}{2} \left(\frac{(\hat{\mathbf{e}}_i \cdot \hat{\mathbf{r}} + \hat{\mathbf{e}}_j \cdot \hat{\mathbf{r}})^2}{1 + \chi(\hat{\mathbf{e}}_i \cdot \hat{\mathbf{e}}_j)} + \frac{(\hat{\mathbf{e}}_i \cdot \hat{\mathbf{r}} - \hat{\mathbf{e}}_j \cdot \hat{\mathbf{r}})^2}{1 - \chi(\hat{\mathbf{e}}_i \cdot \hat{\mathbf{e}}_j)} \right) \right]^{-1/2}$$
(1.31a)

where
$$\chi = \frac{\kappa^2 - 1}{\kappa^2 + 1}$$
, and $\kappa = \sigma_e / \sigma_s$. (1.31b)

 κ is the elongation and χ is the shape anisotropy parameter ($\kappa = 1$, $\chi = 0$ for spherical particles, $\kappa \to \infty$, $\chi \to 1$ for very long rods, and $\kappa \to 0$, $\chi \to -1$ for very thin disks).

The energy term is the product of two functions

$$\epsilon(\hat{\mathbf{r}}, \hat{\mathbf{e}}_i, \hat{\mathbf{e}}_j) = \epsilon_0 \, \epsilon_1^{\nu}(\hat{\mathbf{e}}_i, \hat{\mathbf{e}}_j) \, \epsilon_2^{\mu}(\hat{\mathbf{r}}, \hat{\mathbf{e}}_i, \hat{\mathbf{e}}_j) \tag{1.32a}$$

where

$$\epsilon_1(\hat{\mathbf{e}}_i, \hat{\mathbf{e}}_j) = \left(1 - \chi^2 (\hat{\mathbf{e}}_i \cdot \hat{\mathbf{e}}_j)^2\right)^{-1/2}$$
(1.32b)

$$\epsilon_{2}(\hat{\mathbf{r}}, \hat{\mathbf{e}}_{i}, \hat{\mathbf{e}}_{j}) = 1 - \frac{\chi'}{2} \left(\frac{(\hat{\mathbf{e}}_{i} \cdot \hat{\mathbf{r}} + \hat{\mathbf{e}}_{j} \cdot \hat{\mathbf{r}})^{2}}{1 + \chi'(\hat{\mathbf{e}}_{i} \cdot \hat{\mathbf{e}}_{j})} + \frac{(\hat{\mathbf{e}}_{i} \cdot \hat{\mathbf{r}} - \hat{\mathbf{e}}_{j} \cdot \hat{\mathbf{r}})^{2}}{1 - \chi'(\hat{\mathbf{e}}_{i} \cdot \hat{\mathbf{e}}_{j})} \right)$$
(1.32c)

and the energy anisotropy parameter is

$$\chi' = \frac{\kappa'^{1/\mu} - 1}{\kappa'^{1/\mu} + 1}, \quad \text{where} \quad \kappa' = \epsilon_{\text{ss}}/\epsilon_{\text{ee}}. \tag{1.32d}$$

 ϵ_{ss} and ϵ_{ee} are the well depths of the potentials in the side-by-side and end-to-end configurations respectively. The potential is illustrated for these arrangements, as well as for T-shaped and crossed configurations, in Fig. 1.11. The original model, with exponents $\mu = 2$, $\nu = 1$, and parameters $\kappa = 3$, $\kappa' = 5$, was used to mimic four collinear Lennard-Jones sites (Gay and Berne, 1981). The potential and corresponding force and torque can be readily evaluated and the functional form is rich enough to create mesogens of different shapes and energy anisotropies that will form the full range of nematic, smectic, and discotic liquid crystalline phases (Luckhurst et al., 1990; Berardi et al., 1993; Allen, 2006a;



Fig. 1.11 The Gay-Berne potential, with parameters $\mu = 1$, $\nu = 3$, $\kappa = 3$, $\kappa' = 5$ (Berardi et al., 1993), as a function of centre–centre separation, for various molecular orientations.

Luckhurst, 2006). It is discussed further in Appendix C. Extensions of the potential, and its use in modelling liquid crystals, are discussed by Zannoni (2001).

The MARTINI approach is a coarse-grained potential developed for modelling lipid bilayers (Marrink et al., 2004; 2007) and proteins (Monticelli et al., 2008). In this model the bonded hydrogen atoms are included with their heavier partners, such as C, N, or O. These united atoms are then further combined using a 4:1 mapping to create larger beads (except in the case of rings where the mapping is normally 3:1). For these larger beads, there are four different bead types: charged (Q), polar (P), nonpolar (N), and apolar (C). Each of these types is further subdivided depending on the bead's hydrogen-bond forming propensities or its polarity. Overall there are 18 bead-types and each pair of beads interacts through a Lennard-Jones potential where the σ and ϵ parameters are specific to the atom types involved. Charged beads also interact through Coulombic potentials. The intramolecular interactions (bonds, angles, and torsions) are derived from atomistic simulations of crystal structures. This kind of moderate coarse graining has been successfully applied to simulations of the clustering behaviour of the membrane bound protein syntaxin-1A (van den Bogaart et al., 2011) and the simulation of the domain partitioning of membrane peptides (Schäfer et al., 2011).

It is possible to coarse grain potentials in a way that results in larger beads, that might contain 1–3 Kuhn chain-segments of a polymer or perhaps ten solvent molecules. We will consider this approach more fully in Chapter 12. However, at this point, we mention a very simple coarse-grained model of polymer chains due to Kremer and Grest (1990) and termed the finitely extensible nonlinear elastic (FENE) model. The bonds between beads



Fig. 1.12 The potential between bonded atoms in a coarse-grained polymer (solid line) together with its component parts (dashed lines): the attractive FENE potential, eqn (1.33) with $R_0 = 1.5\sigma$ and $k = 30\epsilon/\sigma^2$, and the repulsive Lennard-Jones potential, eqn (1.10a). Also shown (dotted line) is a harmonic potential, fitted to the curvature at the minimum. See Kremer and Grest (1990) for details.

within the chain are represented by the potential energy

$$v^{\text{FENE}}(r) = \begin{cases} -\frac{1}{2}kR_0^2 \ln(1 - (r/R_0)^2) & r < R_0\\ \infty & r \ge R_0. \end{cases}$$
(1.33)

This is combined with the potential $v^{\text{RLJ}}(r)$ of eqn (1.10a), representing the effects of excluded volume between every pair of beads (including those that are bonded together). The key feature of this potential is that it cannot be extended beyond $r = R_0$. This is important when studying entanglement effects: the simpler harmonic potential could, in principle, extend enough to let chains pass through one another, in some circumstances.

Finally, there has been considerable effort to develop a simple, single-site coarsegrained potential for water. One approach (Molinero and Moore, 2009; Moore and Molinero, 2011) has been to abandon the long-range electrostatics conventionally associated with hydrogen bonds, and use instead short-range directional interactions, of the kind previously used to model silicon (Stillinger and Weber, 1985). The resulting monatomic water (mW) model is very cheap to simulate but surprisingly successful in reproducing experimental structural and thermodynamic properties. Can one go further? It is difficult to imagine that a spherical, isotropic potential will be able to capture the strong association interactions in the fluid. Nevertheless, Lobanova et al. (2015) have used a Mie potential, a versatile form of the standard Lennard-Jones potential, where

$$v_{\text{Mie}}(r) = C\epsilon \left[\left(\frac{\sigma}{r} \right)^n - \left(\frac{\sigma}{r} \right)^m \right], \text{ with } C = \left(\frac{n}{n-m} \right) \left(\frac{n}{m} \right)^{m/(n-m)}.$$
 (1.34)

24 Introduction

A potential with n = 8 and m = 6 can be used with temperature-dependent energy and length parameters to represent the thermophysical properties of water over a broad range of conditions. However, a simpler form where ϵ and σ are independent of temperature can be used to represent water in the calculation of mixture phase diagrams such as CO_2/H_2O (Müller and Jackson, 2014). We briefly discuss this approach to coarse graining in Section 12.7.3. The examples just given are two amongst many attempts to model water in a coarse-grained way (for a review see Hadley and McCabe, 2012).

1.3.5 Calculating the potential

This is an appropriate point to introduce a piece of computer code, which illustrates the calculation of the potential energy in a system of Lennard-Jones atoms. Simulation programs are written in a range of languages: Fortran, C, and C++ are the most common, sometimes with a wrapper written in Python or Java. Here we shall use Fortran, which has a compact notation for arrays and array operations, and is simple enough to be read as a 'pseudo-code'. Appendix A contains some discussion of different programming approaches, and a summary of some of the issues affecting efficiency. We suppose that the coordinate vectors of our atoms are stored in an array r of rank two, with dimensions (3,n), where the first index covers the *x*, *y*, and *z* components, and the second varies from 1 to n (equal to N, the number of particles). The potential energy will be stored in a variable pot, which is zeroed initially, and is then accumulated in a double loop over all distinct pairs of atoms, taking care to count each pair only once. This is shown in Code 1.2. The Lennard-Jones parameters ϵ and σ are assumed to be stored in the variables epslj and sigma respectively. The colon ':' is short for an implied loop over the corresponding index, so the statement rij(:) = r(:,i) - r(:,j) stands for the vector assignment $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$. The SUM function simply adds the components of its (array) argument, which in this case gives $r_{ii}^2 = x_{ii}^2 + y_{ii}^2 + z_{ii}^2$. Code 1.2 takes no account of periodic boundary conditions (we return to this in Section 1.6.2). Some measures have been taken here to avoid unnecessary use of computer time. The value of σ^2 is computed once beforehand, and stored in the variable sigma_sq; the factor 4ϵ , which appears in every pair potential term, is multiplied in once, at the very end. The aim is to avoid many unnecessary operations within the crucial 'inner loop' over index j. The more general questions of time-saving tricks in this part of the program are addressed in Chapter 5. The extension of this type of double loop to deal with other forms of the pair potential, and to compute forces in addition to potential terms, is straightforward, and examples will be given in later chapters. For molecular systems, the same general principles apply, but additional loops over the different sites or atoms in a molecule may be needed. For example, consider the site-site diatomic model of eqn (1.12) and Fig. 1.6. Then the intermolecular interactions might be computed as in Code 1.3. Note that, apart from the dependence of the range of the j loop on the index i, the order of nesting of loops is a matter of choice. Here, we have placed a loop over molecular indices innermost; assuming that n is relatively large, and depending on the machine architecture, this may improve the efficiency of fetching the relevant coordinates from memory (in Fortran, the arrays are stored so that the first indices vary rapidly, and the last indices vary slowly, so there is usually an advantage in accessing contiguous blocks of memory, or cache, in sequence). Simulations of molecular systems may also

Code 1.2 Double loop for Lennard-Jones potential

This code snippet illustrates the calculation of the potential energy for a system of Lennard-Jones atoms, using a double loop over the atomic indices. The declarations at the start are given just to remind us of the types and sizes of variables and arrays (some notes on precision of variables appear in Appendix A).

```
INTEGER
                     :: n, i, j
REAL, DIMENSION(3,n) :: r
REAL, DIMENSION(3)
                   :: rij
REAL
                    :: epslj, sigma, sigma_sq
                    :: pot, rij_sq, sr2, sr6, sr12
REAL
sigma_sq = sigma ** 2
pot = 0.0
DO i = 1, n-1
  DO j = i+1, n
    rij(:) = r(:,i) - r(:,j)
    rij_sq = SUM (rij ** 2)
    sr2 = sigma_sq / rij_sq
         = sr2 ** 3
    sr6
    sr12 = sr6 ** 2
         = pot + sr12 - sr6
   pot
 END DO
END DO
pot = 4.0 * epslj * pot
```

involve the calculation of intramolecular energies, which, for site-site potentials, will necessitate a triple summation (over i, a, and b).

These examples are essentially summations over pairs of interaction sites in the system. Any calculation of three-body interactions will, of course, entail triple summations over distinct triplets of indices i, j, and k; these will be much more time consuming than the double summations described here. Even for pairwise-additive potentials, the energy or force calculation is the most expensive part of a computer simulation. We will return to this crucial section of the program in Chapter 5.

1.4 Constructing an intermolecular potential from first principles

1.4.1 Introduction

There are two approaches to constructing an intermolecular potential for use in a simulation. For small, simple molecules and their mixtures, it is possible to customize a model, with considerable freedom in choosing the functional form of the potentials and in adjusting the parameters for the problem at hand. For larger molecules such as polymers, proteins, or DNA, either in solution or at a surface, or for multi-component mixtures

Code 1.3 Site-site potential energy calculation

The coordinates \mathbf{r}_{ia} of site *a* in molecule *i* are stored in the elements r(:, i, a) of a rank-3 array; for a system of diatomic molecules na=2.

```
INTEGER :: n, i, j, a, b
REAL, DIMENSION(3,n,na) :: r
REAL, DIMENSION(3) :: rij
D0 a = 1, na
    D0 b = 1, na
    D0 i = 1, n - 1
        D0 j = i + 1, n
        rij(:) = r(:,i,a) - r(:,j,b)
        ... calculate the i-j interaction ...
        END D0
        END D0
    END D0
END D0
```

containing many different types of molecule, then it will be more usual to employ one of the standard force fields (consisting of fixed functional forms for the potentials combined with parameters corresponding to the many different atom types in the simulation). We will cover the first aspect of model building in this section and consider force fields in Section 1.5.

There are essentially two stages in setting up a model for a realistic simulation of a given system. The first is 'getting started' by constructing a first guess at a potential model. This will allow some preliminary simulations to be carried out. The second is to use the simulation results, in comparison with experiment, to refine the potential model in a systematic way, repeating the process several times if necessary. We consider the two phases in turn.

1.4.2 Building the model potential

To illustrate the process of building up an intermolecular potential from first principles, we consider a small molecule, such as N_2 , OCS, or CH_4 , which can be modelled using the interaction site potentials discussed in Section 1.3. The essential features of this model will be an anisotropic repulsive core, to represent the shape, an anisotropic dispersion interaction, and some partial charges or distributed multipoles to model the permanent electrostatic effects. This crude effective pair potential can then be refined by using it to calculate properties of the gas, liquid, and solid, and comparing with experiment. Each short-range site–site interaction can be modelled using a Lennard-Jones potential. Suitable energy and length parameters for interactions between pairs of identical atoms in different molecules are available from a number of simulation studies. Some of these are given in Table 1.1. The energy parameter ϵ increases with atomic number as the polarizability goes up; σ also increases down a group of the Periodic Table, but decreases

Atom	Source	$\epsilon/k_{\rm B}$ (K)	σ (nm)
Н	Murad and Gubbins (1978)	8.6	0.281
He	Maitland et al. (1981)	10.2	0.228
С	Tildesley and Madden (1981)	51.2	0.335
Ν	Cheung and Powles (1975)	37.3	0.331
0	English and Venables (1974)	61.6	0.295
F	Singer et al. (1977)	52.8	0.283
Ne	Maitland et al. (1981)	47.0	0.272
S	Tildesley and Madden (1981)	183.0	0.352
Cl	Singer et al. (1977)	173.5	0.335
Ar	Maitland et al. (1981)	119.8	0.341
Br	Singer et al. (1977)	257.2	0.354
Kr	Maitland et al. (1981)	164.0	0.383

Table 1.1 Atom-atom interaction parameters

from left to right across a period with the increasing nuclear charge. For elements which do not appear in Table 1.1, a guide to ϵ and σ might be provided by the polarizability and van der Waals radius respectively. These values are only intended as a reasonable first guess: they take no regard of chemical environment and are not designed to be transferable. For example, the carbon atom parameters in CS₂ given in the table are quite different from the values appropriate to a carbon atom in graphite (Crowell, 1958).

Interactions between unlike atoms in different molecules can be approximated using the venerable Lorentz–Berthelot combining rules. For example, in CS_2 the cross-terms are

$$\sigma_{\rm CS} = \frac{1}{2} \left(\sigma_{\rm CC} + \sigma_{\rm SS} \right), \quad \epsilon_{\rm CS} = \left(\epsilon_{\rm CC} \epsilon_{\rm SS} \right)^{1/2}. \tag{1.35}$$

These rules are approximate; the ϵ cross-term expression, especially, is not expected to be appropriate in the majority of cases (Delhommelle and Millié, 2001; Haslam et al., 2008).

In tackling larger molecules, it may be necessary to model several atoms as a unified site. We have seen this for butane in Section 1.3, and a similar approach has been used in a model of benzene (Evans and Watts, 1976). The specification of an interaction site model is made complete by defining the positions of the sites within the molecule. Normally, these are located at the positions of the nuclei, with the bond lengths obtained from a standard source (CRC, 1984).

Rapid progress has been made in fitting the parameters for many classical pair potentials using *ab initio* quantum mechanical calculations. For example, symmetry-adapted perturbation theory, based on a density-functional approach, can be used to calculate separable and transferable parameters for the dispersion and electrostatic interactions (McDaniel and Schmidt, 2013). Calculations on monomers are used to estimate asymptotic properties such as charge and polarizability, while dimer calculations are used to estimate the parameters depending on charge density overlaps. The resulting parameters can be used with simple functional forms in simulations and the technique has recently been applied to the parameterization and simulation of an ionic liquid (Son et al., 2016).

The site–site Lennard-Jones potentials include an anisotropic dispersion which has the correct r^{-6} radial dependence at long range. However, this is not the exact result for the anisotropic dispersion from second-order perturbation theory. The correct formula, in an appropriate functional form for use in a simulation, is given by Burgos et al. (1982). Its implementation requires an estimate of the polarizability and polarizability anisotropy of the molecule.

It is also possible to improve the accuracy of the overall repulsion–dispersion interaction by considering an anisotropic site–site potential in place of $v_{ab}(\mathbf{r}_{ab})$ in eqn (1.12). In other words, in a diatomic model of a chlorine molecule, the interatomic potential between chlorine atoms in different molecules would depend on r_{ab} and the angles between \mathbf{r}_{ab} and intramolecular bonds. This type of model has been used to rationalize the liquid and solid structures of liquid Cl₂, Br₂, and I₂ (Rodger et al., 1988a,b).

The most straightforward way of representing electrostatic interactions is through partial charges as discussed in Section 1.3. To minimize the calculation of site–site distances they can be made to coincide with the Lennard-Jones sites, but this is not always desirable or possible; the only physical constraint on partial charge positions is that they should not lie outside the repulsive core region, since the potential might then diverge if molecules came too close. The magnitudes of the charges can be chosen to duplicate the known gas-phase electrostatic moments (Gray and Gubbins, 1984, Appendix D). Alternatively, the moments may be taken as adjustable parameters. For example, in a simple three-site model of N₂ representing only the quadrupole–quadrupole interaction, the best agreement with condensed phase properties is obtained with charges giving a quadrupole 10 %–15 % lower than the gas-phase value (Murthy et al., 1980). However, a sensible strategy is to begin with the gas-phase values, and alter the repulsive core parameters ϵ and σ before changing the partial charges.

Partial charges can also be developed using theoretical calculations. Bayly et al. (1993) have developed the widely used restrained electrostatic potential (RESP) method. In this technique:

- (a) a molecule is placed in a 3D grid of points;
- (b) the electrostatic potential is calculated at each grid point, outside the repulsive core, using a quantum mechanical calculation;
- (c) a charge at each atom of the molecules is adjusted to reproduce the electrostatic potential at the grid points as accurately as possible.

Typically, accurate enough quantum mechanical estimates of the electrostatic field can be obtained using the 6-31G^{*} level of the Gaussian code (Frisch et al., 2009). In order to make this fitting procedure robust and to obtain charges that are transferable between different molecules, it is necessary to minimize the magnitude of the charges that will fit the field. This is achieved using a hyperbolic restraint function in the minimization that pulls the magnitude of the charges towards zero.

Distributed multipoles and polarizabilities, for molecules containing up to about 60 atoms, can be calculated from first principles using the CamCASP package developed by Stone and co-workers (Misquitta and Stone, 2013).

1.4.3 Adjusting the model potential

The first-guess potential can be used to calculate a number of properties in the gas, liquid, and solid phases; comparison of these results with experiment may be used to refine the potential, and the cycle can be repeated if necessary. The second virial coefficient is given by

$$B(T) = -\frac{2\pi}{\Omega^2} \int_0^\infty r_{ij}^2 \mathrm{d}r_{ij} \int \mathrm{d}\Omega_i \int \mathrm{d}\Omega_j \exp\left[-v(r_{ij}, \,\Omega_i, \,\Omega_j)/k_{\mathrm{B}}T\right] - 1 \tag{1.36}$$

where $\Omega = 4\pi$ for a linear molecule and $\Omega = 8\pi^2$ for a non-linear one. This multidimensional integral (four-dimensional for a linear molecule and six-dimensional for a non-linear one) is easily calculated using a non-product algorithm (Murad, 1978). Experimental values of B(T) have been compiled by Dymond and Smith (1980). Trial and error adjustment of the Lennard-Jones ϵ and σ parameters should be carried out, with any bond lengths and partial charges held fixed, so as to produce the closest match with the experimental B(T). This will produce an improved potential, but still one that is based on pair properties.

The next step is to carry out a series of computer simulations of the liquid state, as described in Chapters 3 and 4. The densities and temperatures of the simulations should be chosen to be close to the orthobaric curve of the real system, that is, the liquid–vapour coexistence line. The output from these simulations, particularly the total internal energy and the pressure, may be compared with the experimental values. The coexisting pressures are readily available (Rowlinson and Swinton, 1982), and the internal energy can be obtained approximately from the known latent heat of evaporation. The energy parameters ϵ are adjusted to give a good fit to the internal energies along the orthobaric curve, and the length parameters σ altered to fit the pressures. If no satisfactory fit is obtained at this stage, the partial charges may be adjusted. It is also possible to adjust potential parameters to reproduce structural properties of the liquid, such as the site–site pair distribution functions (see Section 2.6), which can be extracted from coherent neutron diffraction studies using isotopic substitution (Cole et al., 2006; Zeidler et al., 2012).

Although the solid state is not the province of this book it offers a sensitive test of any potential model. Using the experimentally observed crystal structure, and the refined potential model, the lattice energy at zero temperature can be compared with the experimental value (remembering to add a correction for quantum zero-point motion). In addition, the lattice parameters corresponding to the minimum energy for the model solid can be compared with the values obtained by diffraction, and also lattice dynamics calculations (Neto et al., 1978) used to obtain phonons, librational modes, and dispersion curves of the model solid. Finally, we can ask if the experimental crystal structure is indeed the minimum energy structure for our potential. These constitute severe tests of our model-building skills (Price, 2008).

1.5 Force fields

In approaching the simulation of a complicated system, there might be 30 different atom types to consider and several hundred different intra- and inter-molecular potentials to fit. One would probably not want to build the potential model from scratch. Fortunately, it is

possible to draw on the considerable body of work that has gone into the development of consistent force fields over the last 50 years (Bixon and Lifson, 1967; Lifson and Warshel, 1968; Ponder and Case, 2003).

A force field, in the context of a computer simulation, refers to the functional forms used to describe the intra- and inter-molecular potential energy of a collection of atoms, and the corresponding parameters that will determine the energy of a given configuration. These functions and parameters have been derived from experimental work on single molecules and from accurate quantum mechanical calculations. They are often refined by the use of computer simulations to compare calculated condensed phase properties with experiment. This is precisely the same approach described in Section 1.4.3, but on a bigger scale, so that the transferable parameters developed can be used with many different molecules. Some examples of widely used force fields are given in Table 1.2. This list is representative and not complete. The individual force fields in the table are constantly being updated and extended. For example, the OPLs force field has been refined to allow for the modelling of carbohydrates (Kony et al., 2002) and the OPLS and AMBER force fields have been used as the basis of a new field for ionic liquids (Lopes et al., 2004). Extensions and versions are often denoted by the ffXX specification following the force field name. A short search of the websites of the major force fields will establish the latest version and the most recent developments.

Force fields are often divided into three classes. Class I force fields normally have a functional form of the type

$$\mathcal{V} = \sum_{\text{bonds}} \frac{1}{2} k_r (r_{ij} - r_0)^2 + \sum_{\text{angles}} \frac{1}{2} k_\theta (\theta_{ijk} - \theta_0)^2 + \sum_{\text{torsions}} \sum_n k_{\phi,n} [\cos(n\phi_{ijk\ell} + \delta_n) + 1] + \sum_{\substack{\text{non-bonded} \\ \text{pairs}}} \left[\frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} + \frac{A_{ij}}{r_{ij}^1} - \frac{B_{ij}}{r_{ij}^6} \right].$$
(1.37)

The first term in eqn (1.37) is a sum over all bonds, with an equilibrium bond-length r_0 . There is one term for every pair *ij* of directly connected atoms. In some force fields the harmonic potential can be replaced by a more realistic functional form, such as the Morse potential, or the bonds can be fixed at their equilibrium values. The second term is a sum over all bond angles. There is one term for each set of three connected atoms *ijk* and it usually has a quadratic form. The third term is the sum over all torsions involving four connected atoms $ijk\ell$. In principle, this is an expansion in trigonometric functions with different values of n, the multiplicity (i.e. the number of minima in a rotation of 2π around the *j*-*k* bond); many force fields fix n = 3. This term can also include improper torsions, where the four atoms defining the angle are not all connected by covalent bonds; such terms serve primarily to enforce planarity around sp^2 centres and use a variety of functional forms (Tuzun et al., 1997). The fourth term is a sum over the non-bonded interactions (between molecules and within molecules). In particular, it describes the electrostatic and repulsion-dispersion interactions. It invariably excludes 1-2 and 1-3 pairs in the same molecule. Some force fields do include a non-bonded 1-4interaction but the parameters A'_{ii} , B'_{ii} describing this interaction can be different from the values for atoms separated by more than three bonds (a scaling factor of 0.4 is used in the param19 force field of CHARMM (Brooks et al., 1983)). In some force fields, the r_{ii}^{-12}

Force field	Class	Domain of Application	Source
OPLS	Ι	peptides, small organics	Jorgensen et al. (1996)
CHARMM22	Ι	proteins with explicit water	Mackerell et al. (1998)
CHARMM27	Ι	DNA, RNA, and lipids	Mackerell et al. (1998)
AMBER ff99	Ι	peptides, small organics, RESP charges	Wang et al. (2000)
GAFF	Ι	small organics, drug design	Wang et al. (2004)
GROMOS ffG45a3	Ι	lipids, micelles	Schuler et al. (2001)
COMPASS	II	small molecules, polymers	Sun (1998)
clayff	II	hydrated minerals	Cygan et al. (2004)
MM4	II	small organics, coordination compounds	Allinger et al. (1996)
UFF	II	full Periodic Table (including actinides)	Rappe et al. (1992)
AMBER ff02	III	polarizable atoms	Cieplak et al. (2001)
AMOEBA	III	polarizable multipoles, distributed multipoles	Ponder et al. (2010)
MARTINI	III	coarse-grained, proteins, lipids, polymers	Marrink et al. (2007)
ReaxFF	III	chemical reactions	van Duin et al. (2001)

Table 1.2 Force fields and their domains of application. This list is not complete and simply includes representativeexamples of some of the force fields commonly used in liquid-state simulations.

repulsion (associated with the Lennard-Jones potential) is replaced by an r_{ij}^{-9} repulsion which can produce better agreement with direct quantum calculations of the repulsion (Hagler et al., 1979; Halgren, 1992). The exponential form of the repulsion ($A \exp(-Br_{ij})$) was used in earlier versions of the AMBER force fields (MM2 and MM3) but has now been replaced by the r_{ij}^{-12} repulsion. The cross-interactions for the parameters in the repulsion– dispersion potential are often described using the Lorentz–Berthelot combining rules or an alternative such as the Slater–Kirkwood formula (Slater and Kirkwood, 1931). If these crossed interactions are important in the model they can be determined directly by fitting to experiment. In class I force fields, a simple Coulombic term is used to describe the interaction between the partial charges, which represent the electrostatic interactions between molecules.

Different parameters are required for different atoms in different environments, and all of the atom types in the model must be specified. For example, in the GROMOS force field ffG45a3 (Schuler et al., 2001), there are 12 types of C atoms, six Os, six Ns, four Cls, three Hs, two Ss, two Cus and one type for each of the remaining common atoms. The parameters $\{k_r, k_\theta, k_{\phi,n}, \delta_n, q_i, q_j, A_{ij}, B_{ij}\}$ are then specified for combinations of the atom types. For example, in a peptide chain, which contains C, N, and C_α atom types along the backbone (where C is a carbon additionally double-bonded to an oxygen and C_α is a carbon additionally connected to a hydrogen and a side chain) we would require k_r for the C–N stretch, a different k_r for the N–C $_\alpha$ stretch, k_θ for the C–N–C $_\alpha$ bend, $k_{\phi,n}$, for the C–N–C $_\alpha$ –C torsion, and additional parameters for the other bends and torsion in the backbone.

All-atom force fields provide parameters for every type of atom in a system, including hydrogen, while united-atom force fields treat the hydrogen and carbon atoms in each terminal methyl and each methylene bridge as a single interaction centre.

A class II force field normally adds cubic or anharmonic terms to the stretching potentials and defines explicit off-diagonal elements in the force constant matrix. Thus, the force field will contain terms of the form

$$v^{\text{str-str}}(r_{12}, r_{23}) = k_{12,23}(r_{12} - r_{12,0})(r_{23} - r_{23,0})$$
$$v^{\text{bend-str}}(\theta_{123}, r_{12}) = k_{123,12}(\theta_{123} - \theta_{123,0})(r_{12} - r_{12,0})$$
(1.38)

where r_{12} and r_{23} are two adjacent bonds in the molecule, which include the angle θ_{123} . These additional potentials represent the fact that bonds, angles and torsions are not independent in molecules. Most cross-terms involve two internal coordinates and Dinur and Hagler (1991) have used quantum mechanical calculations to show that the stretch–stretch, stretch–bend, bend–bend, stretch–torsion, and bend–bend–torsion are the important coupling terms. The cross-terms are essential to include in models when attempting to calculate accurate vibrational frequencies. Despite the additional complexity, Class II force fields, such as COMPASS and CFF, have been used to good effect in liquid-state simulations (Peng et al., 1997; Sun, 1998).

Class III force fields go beyond the basic prescription to include more accurate representations of the electrostatic interactions between molecules and the inclusion of polarizability (as discussed in Section 1.3.3). For example, the AMOEBA force field includes distributed multipoles and the atom polarizabilities with the Thole modification of the interaction tensor. This class would also include coarse-grained force fields such as MAR-TINI used to model lipids, proteins, and carbohydrates (see Section 1.3.4) and force fields specifically designed to model chemical reactions such as ReaxFF. ReaxFF includes a set of relationships between the bond distance and the bond order of a particular covalent bond. Once the bond order is determined, the associated bond energy can be calculated. This procedure results in proper dissociation of bonds to separated atoms at the appropriate distances.

After many decades of force field development, there are still considerable differences between the predictions from even the Class I force fields. In an excellent review of the field, Ponder and Case (2003) compare simulations of a solvated dipeptide using CHARMM27, AMBER94, and OPLS-aa force fields to map the free energy of the dipeptide as a function of the two torsional angles, ψ and ϕ . All three force fields exhibit $\psi - \phi$ maps that are different from one another and different from the results of an *ab initio* simulation of the same problem. In contrast, in considering the liquid-state properties for butane, methanol, and N-methylacetamide, Kaminski and Jorgensen (1996) demonstrated reasonable agreement between the AMBER94 and OPLS force field, both of which had been fitted to liquid-state properties. In this study the MMFF94 force field, that had been optimized for gas-phase geometries, needed to be adjusted to obtain the same level of agreement when applied to the liquids. One important point is that it is not possible to mix and match different force fields. They have been optimized as a whole and one should not attempt to use parts of one field with parts of another. This means that devising force fields to simulate very different materials interacting with each other is a particular challenge. As an illustration, the steps taken to model the adsorption of biomolecules on the surface of metallic gold, in water, are discussed in Example 1.2.

It is difficult to make blanket recommendations concerning the use of particular force fields. Individual researchers will need to understand the kind of problems for which the force field has been optimized to know if it can be applied to their particular problem. One sensible strategy would be to check the effect of using a few of the more common force fields on the problem to understand the sensitivity of the results to this choice.

An important advantage of the force-field approach is that that particular fields are often associated with large simulation programs. The acronyms CHARMM, AMBER, and GROMOS can also stand for large molecular dynamics codes which have been designed to work with the particular forms of a field and there are many examples of other codes such as LAMMPS (Plimpton, 1995) and DL_Poly (Todorov and Smith, 2011) that can take standard force fields with some adjustments. There is also a huge industry of analysis and data manipulation programmes that have grown with the major force fields and codes.

Of course, using these programmes as black-boxes is never a good idea and we plan in this book to dig into the principles behind such codes. Equally, if one can take advantage of the many years of careful development that have gone into producing these packages in an informed way, an enormous range of complicated and important applications can be tackled fairly quickly.

Example 1.2 Peptide-gold potentials

Peptides, short chains of amino acids, may be designed so as to specifically favour adsorption on certain material surfaces. This underpins a range of possible bionanotechnology applications (Care et al., 2015). Understanding this selectivity and specificity is a great challenge to molecular simulation: clearly the adsorption free energy depends on many factors, including changes in peptide flexibility, its solvation, and displacement of the water layer at the surface. Measurement of adsorption free energies requires advanced simulation techniques (see Chapters 4 and 9); modelling the potential energy of interaction between the surface and individual amino acids is itself challenging, involving the cross-interaction between two very different materials (Di Felice and Corni, 2011; Heinz and Ramezani-Dakhel, 2016). Here we focus on recent attempts to model peptide interactions with the surface(s) of metallic gold. A simple Lennard-Jones force field for a range of FCC metals, including gold, has been proposed (Heinz et al., 2008): ϵ_{AuAu} and σ_{AuAu} are chosen to reproduce various experimental bulk and surface properties, under ambient conditions. Water and peptide atom–Au parameters are obtained by standard combining rules. Feng et al. (2011) have used this potential to study the adsorption of individual amino acids on gold, while Cannon et al. (2015) have used it to highlight solvent effects in peptide adsorption. A different parameterization, similar in spirit, has been derived independently (Vila Verde et al., 2009; 2011). The whole method has been generalized to cover a range of other materials (Heinz et al., 2013). Compatibility with standard force fields, such as CHARMM, is an advantage of this approach; polarization of the metal, and chemisorption, however, are neglected.

A purely dispersive potential of this kind may have limitations when one considers structure: adsorption (of water molecules or peptide atoms) onto hollow sites on the surface is strongly favoured. On metallic surfaces, however, adsorption on top of surface atoms is often preferred, as indicated by first-principles simulations. In the GolP force field (Iori et al., 2009), dynamical polarization of gold atoms is represented by a rotating dipole, and virtual interaction sites are introduced to tackle the hollow-site adsorption problem. GolP is parameterized using extensive first-principles calculations and experimental data, with special consideration given to surface interactions with sp²-hybridized carbons. An extension, GolP-CHARMM, reparameterized for compatibility with CHARMM, also allows consideration of different gold surfaces (Wright et al., 2013b,a), opening up the study of facet selectivity (Wright et al., 2015). In GolP, the gold atoms are held fixed during the simulation.

Tang et al. (2013) have compared GolP results with experimental studies of peptide adsorption, and with the force field of Heinz et al. (2008). While both models perform reasonably well in describing the trend in amino acid adsorption energies, there are areas such as the prediction of water orientation in the surface layer where GolP-CHARMM agrees better with first-principles simulations (Nadler and Sanz, 2012). This approach may allow one to separate the enthalpic contributions to the binding free energy, and ascribe them to individual residues (Corni et al., 2013; Tang et al., 2013).

1.6 Studying small systems

1.6.1 Introduction

Simulations are usually performed on a small number of molecules, $10 \le N \le 10\,000$. The size of the system is limited by the available storage on the host computer, and, more crucially, by the speed of execution of the program. The time taken for a double loop used to evaluate the forces or potential energy is proportional to N^2 . Special techniques (see Chapter 5) may reduce this dependence to O(N), for very large systems, but the force/energy loop almost inevitably dictates the overall speed and, clearly, smaller systems will always be less expensive. If we are interested in the properties of a very small liquid drop, or a microcrystal, then the simulation will be straightforward. The cohesive forces between molecules may be sufficient to hold the system together unaided during the course of a simulation, otherwise our set of N molecules may be confined by a potential representing a container, which prevents them from drifting apart (see Chapter 13). These arrangements, however, are not satisfactory for the simulation of bulk liquids. A major obstacle to such a simulation is the large fraction of molecules which lie on the surface of any small sample; for 1000 molecules arranged in a $10 \times 10 \times 10$ cube, $8^3 = 512$ lie in the interior, leaving 488 (nearly half!) on the cube faces. Even for $N = 100^3 = 10^6$ molecules, 6% of them will lie on the surface. Whether or not the cube is surrounded by a containing wall, molecules on the surface will experience quite different forces from those in bulk.

1.6.2 Periodic boundary conditions.

The problem of surface effects can be overcome by implementing periodic boundary conditions (Born and von Karman, 1912). The cubic box is replicated throughout space to form an infinite lattice. In the course of the simulation, as a molecule moves in the original box, its periodic image in each of the neighbouring boxes moves in exactly the same way. Thus, as a molecule leaves the central box, one of its images will enter through the opposite face. There are no walls at the boundary of the central box, and no surface molecules. This box simply forms a convenient axis system for measuring the coordinates of the N molecules. A two-dimensional version of such a periodic system is shown in Fig. 1.13. The duplicate boxes are labeled A, B, C, etc., in an arbitrary fashion. As particle 1 moves through a boundary, its images 1_A , 1_B , etc. (where the subscript specifies in which box the image lies) move across their corresponding boundaries. The number density in the central box (and hence in the entire system) is conserved. It is not necessary to store the coordinates of all the images in a simulation (an infinite number!), just the molecules in the central box. When a molecule leaves the box by crossing a boundary, attention may be switched to the image just entering. It is sometimes useful to picture the basic simulation box (in our two-dimensional example) as being rolled up to form the surface of a three-dimensional torus or doughnut, when there is no need to consider an infinite number of replicas of the system, nor any image particles. This correctly represents the topology of the system, if not the geometry. A similar analogy exists for a three-dimensional periodic system, but this is more difficult to visualize!

It is important to ask if the properties of a small, infinitely periodic, system and the macroscopic system which it represents are the same. This will depend both on the range of the intermolecular potential and the phenomenon under investigation. For a fluid of



Fig. 1.13 A two-dimensional periodic system. Molecules can enter and leave each box across each of the four edges. In a three-dimensional example, molecules would be free to cross any of the six cube faces.

Lennard-Jones atoms it should be possible to perform a simulation in a cubic box of side $L \approx 6 \sigma$ without a particle being able to 'sense' the symmetry of the periodic lattice. If the potential is long range (i.e. $v(r) \sim r^{-v}$ where v is less than the dimensionality of the system) there will be a substantial interaction between a particle and its own images in neighbouring boxes, and consequently the symmetry of the cell structure is imposed on a fluid which is in reality isotropic. The methods used to cope with long-range potentials, for example in the simulation of charged ions $(v(r) \sim r^{-1})$ and dipolar molecules $(v(r) \sim r^3)$, are discussed in Chapter 5. We know that even in the case of short-range potentials the periodic boundary conditions can induce anisotropies in the fluid structure (Mandell, 1976; Impey et al., 1981). These effects are pronounced for small system sizes (N = 100) and for properties such as the g_2 light scattering factor (see Chapter 2), which has a substantial long-range contribution. Pratt and Haan (1981) have developed theoretical methods for investigating the effects of boundary conditions on equilibrium properties.

The use of periodic boundary conditions inhibits the occurrence of long-wavelength fluctuations. For a cube of side L, the periodicity will suppress any density waves with a wavelength greater than L. Thus, it would not be possible to simulate a liquid close



Fig. 1.14 Non-cubic, space-filling, simulation boxes. (a) The truncated octahedron and its containing cube; (b) the rhombic dodecahedron and its containing cube. The axes are those used in Code 1.4 and Code 1.5 of Section 1.6.4.

to the gas–liquid critical point, where the range of critical fluctuations is macroscopic. Furthermore, transitions which are known to be first order often exhibit the characteristics of higher-order transitions when modelled in a small box, because of the suppression of fluctuations. Examples are the nematic–isotropic transition in liquid crystals (Luckhurst and Simpson, 1982) and the solid–plastic-crystal transition for N₂ adsorbed on graphite (Mouritsen and Berlinsky, 1982). The same limitations apply to the simulation of long-wavelength phonons in model solids, where in addition, the cell periodicity picks out a discrete set of available wavevectors (i.e. $\mathbf{k} = (n_x, n_y, n_z)2\pi/L$, where n_x, n_y, n_z , are integers) in the first Brillouin zone (Klein and Weis, 1977). Periodic boundary conditions have also been shown to affect the rate at which a simulated liquid nucleates and forms a solid or glass when it is rapidly cooled (Honeycutt and Andersen, 1984).

Despite the preceding remarks, the common experience in simulation work is that periodic boundary conditions have little effect on the equilibrium thermodynamic properties and structures of fluids away from phase transitions and where the interactions are short-ranged. It is always sensible to check that this is true for each model studied. If the resources are available, it should be standard practice to increase the number of molecules (and the box size, so as to maintain constant density) and rerun the simulations. The cubic box has been used almost exclusively in computer simulation studies because of its geometrical simplicity. Of the four remaining semi-regular space-filling polyhedra, the rhombic dodecahedron (Wang and Krumhansl, 1972), and the truncated octahedron (Adams, 1979; 1980) have also been studied. These boxes are illustrated in Fig. 1.14. They are more nearly spherical than the cube, which may be useful for simulating liquids, whose structure is spatially isotropic. In addition, for a given number density, the distance between periodic images is larger than in the cube. This property is useful in calculating distribution functions and structure factors (see Chapters 2 and 8). As we shall see in Section 1.6.4, they are only slightly more complicated to implement in simulations than cubic boxes.



Fig. 1.15 Periodic boundary conditions used in the simulation of adsorption (see e.g. Severin and Tildesley, 1980). (a) A side view of the box. There is a reflecting boundary at height L_z . (b) A top view, showing the rhombic shape (i.e. the same geometry as the underlying graphite lattice). Periodic boundary conditions in this geometry are implemented in Code 1.6.

So far, we have tacitly assumed that there is no external potential, that is, no v_1 , term in eqns (1.4) and (1.5). If such a potential is present, then either it must have the same periodicity as the simulation box, or the periodic boundaries must be abandoned. In some cases, it is not appropriate to employ periodic boundary conditions in each of the three coordinate directions. In the simulation of CH_4 on graphite (Severin and Tildesley, 1980) the simulation box, shown in Fig. 1.15, is periodic in the plane of the surface. In the *z*-direction, the graphite surface forms the lower boundary of the box, and the bulk of the adsorbate is in the region just above the graphite. Any molecule in the gas above the surface is confined by reversing its velocity should it cross a plane at a height L_z above the surface. If L_z is sufficiently large, this reflecting boundary will not influence the behaviour of the adsorbed monolayer. In the plane of the surface, the shape of the periodic box is a rhombus of side *L*. This conforms to the symmetry of the underlying graphite. Similar boxes have been used in the simulation of the electrical double layer (Torrie and Valleau, 1979), of the liquid–vapour surface (Chapela et al., 1977), and of fluids in small pores (Subramanian and Davis, 1979).



Fig. 1.16 The minimum image convention in a two-dimensional system. The central 'box' contains five molecules. The dashed 'box' constructed with molecule 1 at its centre also contains five molecules. The dashed circle represents the cutoff.

1.6.3 Potential truncation

Now we must turn to the question of calculating properties of systems subject to periodic boundary conditions. The heart of the MC and MD programs involves the calculation of the potential energy of a particular configuration, and, in the case of MD, the forces acting on all molecules. Consider how we would calculate the force on molecule 1, or those contributions to the potential energy involving molecule 1, assuming pairwise additivity. We must include interactions between molecule 1 and every other molecule *i* in the simulation box. There are N - 1 terms in this sum. However, in principle, we must also include all interactions between molecule 1 and images i_A , i_B , etc. lying in the surrounding boxes. This is an infinite number of terms, and of course is impossible to calculate in practice. For a short-range potential-energy function, we may restrict this summation by making an approximation. Consider molecule 1 to rest at the centre of a region which has the same size and shape as the basic simulation box (see Fig. 1.16). Molecule 1 interacts with all the molecules whose centres lie within this region, that is, with the closest periodic images of the other N - 1 molecules. This is called the 'minimum image convention': for example, in Fig. 1.16, molecule 1 could interact with molecules 2, 3_D , 4_E , and 5_C . This technique, which is a natural consequence of the periodic boundary conditions, was first used in simulation by Metropolis et al. (1953).

In the minimum image convention, then, the calculation of the potential energy due to pairwise-additive interactions involves $\frac{1}{2}N(N-1)$ terms. This may still be a very substantial calculation for a system of (say) 1000 particles. A further approximation significantly improves this situation. The largest contribution to the potential and forces comes from neighbours close to the molecule of interest, and for short-range forces we normally apply a spherical cutoff. This means setting the pair potential v(r) to zero for $r \ge r_c$, where r_c is the cutoff distance. The dashed circle in Fig. 1.16 represents a cutoff, and in this case molecules 2, 4_E and 5_C contribute to the force on 1, since their centres lie inside the cutoff, whereas molecule 3_D does not contribute. In a cubic simulation box of side L, the number of neighbours explicitly considered is reduced by a factor of approximately $4\pi r_c^3/3L^3$, and this may be a substantial saving. The introduction of a spherical cutoff should be a small perturbation, and the cutoff distance should be sufficiently large to ensure this. As an example, in the simulation of Lennard-Jones atoms the value of the pair potential at the boundary of a cutoff sphere of typical radius $r_{\rm c} = 2.5 \sigma$ is just 1.6 % of the well depth. Of course, the penalty of applying a spherical cutoff is that the thermodynamic (and other) properties of the model fluid will no longer be exactly the same as for (say) the non-truncated, Lennard-Jones fluid. As we shall see in Chapter 2, it is possible to apply long-range corrections to such results so as to recover, approximately, the desired information.

The cutoff distance must be no greater than $\frac{1}{2}L$ for consistency with the minimum image convention. In the non-cubic simulation boxes of Fig. 1.14, for a given density and number of particles, r_c may take somewhat larger values than in the cubic case. Looked at another way, an advantage of non-cubic boundary conditions is that they permit simulations with a given cutoff distance and density to be conducted using fewer particles. As an example, a simulation in a cubic box, with r_c set equal to $\frac{1}{2}L$, might involve N = 256 molecules; taking the same density, the same cutoff could be used in a simulation of 197 molecules in a truncated octahedron, or just 181 molecules in a rhombic dodecahedron.

1.6.4 Computer code for periodic boundaries

How do we handle periodic boundaries and the minimum image convention in a simulation program? Let us assume that, initially, the *N* molecules in the simulation lie within a cubic box of side *L*, with the origin at its centre, that is, all coordinates lie in the range $(-\frac{1}{2}L, \frac{1}{2}L)$. As the simulation proceeds, these molecules move about the infinite periodic system. When a molecule leaves the box by crossing one of the boundaries, it is usual to switch attention to the image molecule entering the box by simply adding *L* to, or subtracting *L* from, the appropriate coordinate. One simple way to do this uses an IF statement to test the positions immediately after the molecules have been moved (whether by MC or MD). For example,

IF (r(1,i) > box2) r(1,i) = r(1,i) - boxIF (r(1,i) < -box2) r(1,i) = r(1,i) + box where the first index 1 selects the x coordinate. Similar statements are applied to the y and z coordinates, or a vector assignment may be applied to all components at once

WHERE (r(:,i) > box2) r(:,i) = r(:,i) - boxWHERE (r(:,i) < -box2) r(:,i) = r(:,i) + box

Here, box is a variable containing the box length *L*, and box2 is just $\frac{1}{2}L$. An alternative to the IF statement is to use arithmetic functions to calculate the correct number of box lengths to be added or subtracted. For example,

r(:,i) = r(:,i) - box * ANINT (r(:,i) / box)

The function ANINT(x) returns the nearest integer to x, converting the result back to type REAL; thus ANINT(-0.49) has the value 0.0, whereas ANINT(-0.51) is -1.0. In Fortran, this function returns an array-valued result, computed component by component, if given an array argument. As we shall see in Chapter 5, there are faster ways of coding this up, especially for large system sizes.

By using these methods, we always have available the coordinates of the *N* molecules that currently lie in the 'central' box. It is not strictly necessary to do this; we could, instead, use uncorrected coordinates, and follow the motion of the *N* molecules that were in the central box at the start of the simulation. Indeed, as we shall see in Chapters 2 and 8, for calculation of transport coefficients it may be most desirable to have a set of uncorrected positions on hand. If it is decided to do this, however, care must be taken that the minimum image convention is correctly applied, so as to work out the vector between the two closest images of a pair of molecules, no matter how many 'boxes' apart they may be. This means, in general, adding or subtracting an integer number of box lengths (rather than just one box length).

The minimum image convention may be coded in the same way as the periodic boundary adjustments. Of the two methods just mentioned, the arithmetic formula is usually preferable, being simpler; the use of IF statements inside the inner loop may reduce program efficiency (see Appendix A). Immediately after calculating a pair separation vector, the following statements should be applied:

rij(:) = rij(:) - box * ANINT (rij(:) / box)

This code is guaranteed to yield the minimum image vector, no matter how many 'box lengths' apart the original images may be. For cuboidal, rather than cubic, boxes, the variable box may be an array of three elements, holding the x, y, and z box lengths, without essentially changing the code.

The calculation of minimum image distances is simplified by the use of reduced units: the length of the box is taken to define the fundamental unit of length in the simulation. By setting L = 1, with particle coordinates nominally in the range $\left(-\frac{1}{2}, +\frac{1}{2}\right)$, the minimum image correction becomes

which is simpler, and faster, than the code for a general box length. This approach is an alternative to the use of the pair potential to define reduced units as discussed in Appendix B, and is more generally applicable. For this reason a simulation box of unit length is adopted in most of the examples given in this book.

Code 1.4 Periodic boundaries for truncated octahedron

This code snippet applies the truncated octahedron periodic boundary correction to a position vector \mathbf{r}_i , or equivalently the minimum image convention to a displacement vector \mathbf{r}_{ij} , provided as the array r. The box is centred at the origin and the containing cube is of unit length (see Fig. 1.14(a)). The Fortran AINT function rounds towards zero, producing a real-valued integer result: for example AINT(-0.51) and AINT(0.51) both have the value 0.0, whereas AINT(-1.8) is -1.0. The result of the Fortran SIGN function has the absolute value of its first argument and the sign of its second.

```
REAL, DIMENSION(3) :: r
REAL :: corr
REAL, PARAMETER :: r75 = 4.0 / 3.0
r(:) = r(:) - ANINT ( r(:) )
corr = 0.5 * AINT ( r75 * SUM ( ABS ( r(:) ) ) )
r(:) = r(:) - SIGN ( corr, r(:) )
```

Code 1.5 Periodic boundaries for rhombic dodecahedron

This code snippet applies the rhombic dodecahedron periodic boundary correction to a position vector \mathbf{r}_i , or equivalently the minimum image convention to a displacement vector \mathbf{r}_{ij} , provided as the array r. The box is centred at the origin and the side of the containing cube is $\sqrt{2}$ (see Fig. 1.14(b)).

```
REAL, DIMENSION(3) :: r
REAL , DIMENSION(3) :: r
REAL , PARAMETER :: rt2 = SQRT(2.0), rrt2 = 1.0 / rt2
r(1) = r(1) - ANINT ( r(1) )
r(2) = r(2) - ANINT ( r(2) )
r(3) = r(3) - rt2 * ANINT ( rrt2 * r(3) )
corr = 0.5 * AINT ( ABS(r(1)) + ABS(r(2)) + rt2*ABS(r(3)) )
r(1) = r(1) - SIGN ( corr, r(1) )
r(2) = r(2) - SIGN ( corr, r(2) )
r(3) = r(3) - SIGN ( corr, r(3) ) * rt2
```

There are several alternative ways of coding the minimum image corrections, some of which rely on the images being in the same, central box (i.e. on the periodic boundary correction being applied whenever the molecules move). Some of these methods, for cubic boxes, are discussed in Appendix A. We have also mentioned the possibility of conducting simulations in non-cubic periodic boundary conditions. An implementation of the minimum image correction for the truncated octahedron (Adams, 1983a) is given

Code 1.6 Periodic boundaries for rhombus

Here we apply corrections for the rhombic box in two dimensions x, y. In most applications the molecules will be confined in the z direction by real walls rather than by periodic boundaries, so we assume that this coordinate may be left unchanged. The box is centred at the origin. The x axis lies along one side of the rhombus, which is of unit length (see Fig. 1.15). The acute angle of the rhombus is 60° .

in Code 1.4. A similar correction for the rhombic dodecahedron (Smith, 1983) appears in Code 1.5. This is a little more complicated than the code for the truncated octahedron, and the gain small, so that the latter is usually preferable. We also give in Code 1.6 the code for the two-dimensional rhombic box often used in surface simulation.

Now we turn to the implementation of a spherical cutoff, that is, we wish to set the pair potential (and all forces) to zero if the pair separation lies outside some distance r_c . It is easy to compute the square of the particle separation r_{ij} and, rather than waste time taking the square root of this quantity, it is fastest to compare this with the square of r_c which might be computed earlier and stored in a variable r_cut_sq . After computing the minimum image intermolecular vector, the following statements would be employed:

```
rij_sq = SUM ( rij(:) ** 2 )
IF ( rij_sq < r_cut_sq ) THEN
    ... compute i-j interaction ...
END IF</pre>
```

In a large system, it may be worthwhile to apply separate tests for the *x*, *y*, and *z* directions or some similar scheme.

```
IF ( ABS ( rij(1) ) < r_cut ) THEN
IF ( ABS ( rij(2) ) < r_cut ) THEN
IF ( ABS ( rij(3) ) < r_cut ) THEN
rij_sq = SUM ( rij(:) ** 2 )
IF ( rij_sq < r_cut_sq ) THEN
... compute i-j interaction ...
END IF
END IF
END IF
END IF
END IF</pre>
```

44 Introduction

The time saved in dropping out of this part of the program at any early stage must be weighed against the overheads of extra calculation and testing. In Chapter 5 we discuss the more complicated time-saving tricks used in the simulations of large systems.

1.6.5 Spherical boundary conditions

As an alternative to the standard periodic boundary conditions for simulating bulk liquids, a two-dimensional system may be embedded in the surface of a sphere without introducing any physical boundaries (Hansen et al., 1979), and the idea may be extended to consider a three-dimensional system as being the surface of a hypersphere (Kratky, 1980; Kratky and Schreiner, 1982). The spherical or hyperspherical system is finite: it cannot be considered as part of an infinitely repeating periodic system. In this case, non-Euclidean geometry is an unavoidable complication, and distances between particles are typically measured along the great circle geodesics joining them. However, the effects of the curved geometry will decrease as the system size increases, and such 'spherical boundary conditions' are expected to be a valid method of simulating bulk liquids. Interesting differences from the standard periodic boundary conditions, particularly close to any solid-liquid phase transition, will result from the different topology. Periodic boundaries will be biased in favour of the formation of a solid with a lattice structure which matches the simulation box. Spherical boundaries, on the other hand, are not consistent with periodic lattices, so the liquid state will be thermodynamically favoured in most simulations using this technique, and crystalline phases will inevitably contain defects. Similar considerations may apply to liquid-crystalline phases.

1.6.6 Periodic boundary conditions for three-body potentials

Finally, we note that some care is required when using the minimum image convention with three-body potentials such as the Axilrod–Teller potential (see Appendix C). This problem is illustrated in Fig. 1.17. In Fig. 1.17(a), atom 1 is at the centre of its box, of side L, and atoms 2 and 3_E are the two minimum images used in the calculation of the pair potential. However atom 3 is the minimum image of atom 2 and a straightforward application of the minimum image algorithm will lead to the incorrect triplet 123 rather than 123_E .

Attard (1992) has shown that this problem can be solved using the following statements for the separation vector

```
REAL, DIMENSION(3) :: rij, rik, rjk, tij, tik
tij(:) = box * ANINT ( rij(:) / box )
tik(:) = box * ANINT ( rik(:) / box )
rij(:) = rij(:) - tij(:)
rik(:) = rik(:) - tik(:)
rjk(:) = rjk(:) + tij(:) - tik(:)
```

Normally the three-body potential is set to zero if one side of the triangle is greater than L/2.

Some workers have taken a more brute-force approach (Sadus and Prausnitz, 1996; Marcelli and Sadus, 2012). If the potential cutoff r_c is set to L/4, the only triplets that contribute to the potential are those where all of the three atoms are within a box of side



Fig. 1.17 Periodic boundary conditions and the minimum image convention for a triplet interaction: (a) an inconsistency in the triplet configuration for a cutoff of L/2; (b) a consistent triplet with a cutoff of L/4.

L/2 (as shown in Fig. 1.17(b)). Each of the atoms is then always the unique minimum image of the other two and the triplet is unambiguously determined with the normal minimum image calculation. This method works well. However, at a fixed density the simulation will need to include eight times as many atoms in circumstances where the additional calculation of the three-body force is particularly expensive.

Statistical mechanics

Computer simulation generates information at the microscopic level (atomic and molecular positions, velocities, etc.) and the conversion of this very detailed information into macroscopic terms (pressure, internal energy, etc.) is the province of statistical mechanics. It is not our aim to provide a text in this field since many excellent sources are available (Hill, 1956; McQuarrie, 1976; Landau and Lifshitz, 1980; Friedman, 1985; Chandler, 1987; Tuckerman, 2010; Swendsen, 2012; Hansen and McDonald, 2013). In this chapter, our aim is to summarize those aspects of the subject which are of most interest to the computer simulator.

Sampling from ensembles 2.1

Let us consider, for simplicity, a one-component macroscopic system; extension to a multicomponent system is straightforward. The thermodynamic state of such a system is usually defined by a small set of parameters (such as the number of particles N, the temperature T, and the pressure P). Other thermodynamic properties (density ρ , chemical potential μ , heat capacity C_V , etc.) may be derived through knowledge of the equations of state and the fundamental equations of thermodynamics. Even quantities such as the diffusion coefficient D, the shear viscosity η , and the structure factor S(k) are state functions: although they clearly say something about the microscopic structure and dynamics of the system, their values are completely dictated by the few variables (e.g. *NPT*) characterizing the thermodynamic state, not by the very many atomic positions and momenta that define the instantaneous mechanical state. These positions and momenta can be thought of as coordinates in a multidimensional space: phase space. For a system of N atoms, this space has 6N dimensions. Let us use the abbreviation Γ for a particular point in phase space, and suppose that we can write the instantaneous value of some property \mathcal{A} (it might be the potential energy) as a function $\mathcal{A}(\Gamma)$. The system evolves in time so that Γ , and hence $\mathcal{A}(\Gamma)$ will change. It is reasonable to assume that the experimentally observable 'macroscopic' property \mathcal{R}_{obs} is really the time average of $\mathcal{R}(\Gamma)$ taken over a long time interval:

$$\mathcal{A}_{\rm obs} = \langle \mathcal{A} \rangle_{\rm time} = \left\langle \mathcal{A} \left(\Gamma(t) \right) \right\rangle_{\rm time} = \lim_{t_{\rm obs} \to \infty} \frac{1}{t_{\rm obs}} \int_0^{t_{\rm obs}} \mathcal{A} \left(\Gamma(t) \right) \mathrm{d}t.$$
(2.1)

The equations governing this time evolution, Newton's equations of motion in a simple classical system, are of course well known. They are just a system of ordinary differential

Computer Simulation of Liquids. Second Edition. M. P. Allen and D. J. Tildesley.

© M. P. Allen and D. J. Tildesley 2017. Published in 2017 by Oxford University Press.

equations: solving them on a computer, to a desired accuracy, is a practical proposition for, say, 10⁵ particles, although not for a truly macroscopic number (e.g. 10²³). So far as the calculation of time averages is concerned, we clearly cannot hope to extend the integration of eqn (2.1) to infinite time, but might be satisfied to average over a long finite time τ_{obs} . This is exactly what we do in a molecular dynamics simulation. In fact, the equations of motion are usually solved on a step-by-step basis, that is, a large finite number τ_{obs} of timesteps, of length $\delta t = t_{obs}/\tau_{obs}$, are taken. In this case, we may rewrite eqn (2.1) in the form

$$\mathcal{A}_{\rm obs} = \left\langle \mathcal{A} \right\rangle_{\rm time} = \frac{1}{\tau_{\rm obs}} \sum_{\tau=1}^{\tau_{\rm obs}} \mathcal{A} \big(\Gamma(t) \big). \tag{2.2}$$

In the summation, τ simply stands for an index running over the succession of timesteps. This analogy between the discrete τ and the continuous t is useful, even when, as we shall see in other examples, τ does not correspond to the passage of time in any physical sense.

The practical questions regarding the method are whether or not a sufficient region of phase space is explored by the system trajectory to yield satisfactory time averages within a feasible amount of computer time, and whether thermodynamic consistency can be attained between simulations with identical macroscopic parameters (density, energy, etc.) but different initial conditions (atomic positions and velocities). The answers to these questions are that such simulation runs are indeed within the power of modern computers, and that thermodynamically consistent results for liquid state properties can indeed be obtained, provided that attention is paid to the selection of initial conditions. We will turn to the technical details of the method in Chapter 3.

The calculation of time averages by MD is not the approach to thermodynamic properties implicit in conventional statistical mechanics. Because of the complexity of the time evolution of $\mathcal{A}(\Gamma(t))$ for large numbers of molecules, Gibbs suggested replacing the time average by the ensemble average. Here, we regard an ensemble as a collection of points Γ in phase space. The points are distributed according to a probability density $\rho(\Gamma)$. This function is determined by the chosen fixed macroscopic parameters (*NPT*, *NVT*, etc.), so we use the notation ρ_{NPT} , ρ_{NVT} , or, in general, ρ_{ens} . Each point represents a typical system at any particular instant of time. Each system evolves in time, according to the usual mechanical equations of motion, quite independently of the other systems. Consequently, in general, the phase space density $\rho_{ens}(\Gamma)$ will change with time. However, no systems are destroyed or created during this evolution, and Liouville's theorem, which is essentially a conservation law for probability density, states that $d\rho/dt = 0$ where d/dt denotes the total derivative with respect to time (following a state Γ as it moves). As an example, consider a set of N atoms with Cartesian coordinates \mathbf{r}_i , and momenta \mathbf{p}_i , in the classical approximation. The total time derivative is

$$\frac{\mathrm{d}}{\mathrm{d}t} = \frac{\partial}{\partial t} + \sum_{i} \dot{\mathbf{r}}_{i} \cdot \boldsymbol{\nabla}_{\mathbf{r}_{i}} + \sum_{i} \dot{\mathbf{p}}_{i} \cdot \boldsymbol{\nabla}_{\mathbf{p}_{i}}$$
(2.3a)

$$= \frac{\partial}{\partial t} + \dot{\mathbf{r}} \cdot \boldsymbol{\nabla}_{\mathbf{r}} + \dot{\mathbf{p}} \cdot \boldsymbol{\nabla}_{\mathbf{p}}.$$
 (2.3b)

In eqn (2.3a), $\partial/\partial t$ represents differentiation, with respect to time, of a function; $\nabla_{\mathbf{r}_i}$, and $\nabla_{\mathbf{p}_i}$, are derivatives with respect to atomic position and momentum respectively; and $\dot{\mathbf{r}}_i$,

48 Statistical mechanics

 $\dot{\mathbf{p}}_i$, signify the time derivatives of the position and momentum. Equation (2.3b) is the same equation written in a more compact way, and the equation may be further condensed by defining the Liouville operator *L*

$$iL = \left(\sum_{i} \dot{\mathbf{r}}_{i} \cdot \boldsymbol{\nabla}_{\mathbf{r}_{i}} + \sum_{i} \dot{\mathbf{p}}_{i} \cdot \boldsymbol{\nabla}_{\mathbf{p}_{i}}\right) = \left(\dot{\mathbf{r}} \cdot \boldsymbol{\nabla}_{\mathbf{r}} + \dot{\mathbf{p}} \cdot \boldsymbol{\nabla}_{\mathbf{p}}\right)$$
(2.4)

so that $d/dt = \partial/\partial t + iL$ and, using Liouville's theorem, we may write

$$\frac{\partial \rho_{\rm ens}(\Gamma, t)}{\partial t} = -iL\rho_{\rm ens}(\Gamma, t).$$
(2.5)

This equation tells us that the rate of change of ρ_{ens} at a particular fixed point in phase space is related to the flows into and out of that point. This equation has a formal solution

$$\rho_{\rm ens}(\Gamma, t) = \exp(-iLt)\,\rho_{\rm ens}(\Gamma, 0) \tag{2.6}$$

where the exponential of an operator really means a series expansion

$$\exp(-iLt) = 1 - iLt - \frac{1}{2}L^2t^2 + \cdots$$
 (2.7)

The equation of motion of a function like $\mathcal{A}(\Gamma)$, which does not depend explicitly on time, takes a conjugate form (McQuarrie, 1976):

$$\dot{\mathcal{A}}(\Gamma(t)) = iL\mathcal{A}(\Gamma(t))$$
 (2.8)

or

$$\mathcal{A}(\Gamma(t)) = \exp(iLt)\mathcal{A}(\Gamma(0)).$$
(2.9)

To be quite clear: in eqns (2.5) and (2.6) we consider the time-dependence of ρ_{ens} at a fixed point Γ in phase space; in eqns (2.8) and (2.9), $\mathcal{A}(\Gamma(t))$ is time-dependent because we are following the time evolution $\Gamma(t)$ along a trajectory. This relationship is analogous to that between the Schrödinger and Heisenberg pictures in quantum mechanics.

If $\rho_{ens}(\Gamma)$ represents an equilibrium ensemble, then its time-dependence completely vanishes, $\partial \rho_{ens}/\partial t = 0$. The system evolution then becomes quite special. As each system leaves a particular state $\Gamma(\tau)$ and moves on to the next, $\Gamma(\tau + 1)$, another system arrives from state $\Gamma(\tau - 1)$ to replace it. The motion resembles a long and convoluted conga line at a crowded party (see Fig. 2.1). There might be several such processions, each passing through different regions of phase space. However, if these are all connected into just one trajectory that passes through all the points in phase space for which ρ_{ens} is non-zero (i.e. the procession forms a single, very long, closed circuit) then each system will eventually visit all the state points. Such a system is termed 'ergodic' and the time taken to complete a cycle (the Poincaré recurrence time) is immeasurably long for a many-particle system (and for many parties as well it seems).

One way of answering the question 'was it a good party?' would be to interview one of the participants, and ask for their time-averaged impressions. This is essentially what we do in a molecular dynamics simulation, when a representative system evolves deterministically in time. However, as indicated in Fig. 2.1, this time average might not



Fig. 2.1 A schematic representation of phase space. The circles represent different state points (q, p), and they are connected by a path representing the classical trajectory, analogous to a conga line at a party. Each state is characterized by some property (e.g. 'happiness' at the party). In an ergodic system, the single long trajectory would eventually pass through (or arbitrarily near) all states; in the bottom left corner of the diagram we symbolically indicate a disconnected region of six states which may or may not be practically important.

be representative of the whole trajectory: to be sure, it would have to be long enough to sample all the states. An alternative route to the average properties of our partygoers, would be to take photographs of all of them at the same time, assemble the complete collection of 'happy' and 'sad' faces, and take an average over them. This corresponds to replacing the time average in eqn (2.1) by an average taken over all the members of the ensemble, 'frozen' at a particular time:

$$\mathcal{A}_{\rm obs} = \langle \mathcal{A} \rangle_{\rm ens} = \left\langle \mathcal{A} \mid \rho_{\rm ens} \right\rangle = \sum_{\Gamma} \mathcal{A}(\Gamma) \rho_{\rm ens}(\Gamma).$$
(2.10)

The $\langle \mathcal{A} | \rho \rangle$ notation reminds us of the dependence of the average on both \mathcal{A} and ρ : this is important when taking a thermodynamic derivative of \mathcal{A}_{obs} (we must differentiate both parts) or when considering time-dependent properties (when the Schrödinger–Heisenberg analogy may be exploited). Actually, we will be concerned with the practical question of efficient and thorough sampling of phase space, which is not quite the same as the rigorous definition of ergodicity (for a fuller discussion, see Tolman, 1938). In terms of our analogy of conga lines, there should not be a preponderance of independent closed circuits ('cliques') in which individuals can become trapped and fail fully to sample the available space (this is important in parties as well as in simulations). An MD simulation which started in the disconnected six-state region of Fig. 2.1, for example, would be disastrous. On the other hand, small non-ergodic regions are less likely to be dangerous and more likely to be recognized if they are unfortunately selected as starting points for

50 Statistical mechanics

a simulation. In a similar way, regions of phase space which act as barriers and cause bottlenecks through which only a few trajectories pass can result in poor sampling by the relatively short simulation runs carried out in practice, even if the system is technically ergodic.

Finally, we might use a different kind of evolution to sample the states of the system: a random walk. This is the Monte Carlo approach: it may be more or less efficient than molecular dynamics. It also fits quite well the analogy of a party, in which the participants sample the different situations randomly, rather than systematically. Once again, trajectory averages are calculated over a finite duration, so these are not necessarily identical to full ensemble averages, and the approach might or might not alleviate some of the ergodicity issues.

It is sometimes convenient to use, in place of $\rho_{ens}(\Gamma)$, a 'weight' function $w_{ens}(\Gamma)$, which satisfies the following equations:

$$\rho_{\rm ens}(\Gamma) = Q_{\rm ens}^{-1} w_{\rm ens}(\Gamma) \tag{2.11}$$

$$Q_{\rm ens} = \sum_{\Gamma} w_{\rm ens}(\Gamma) \tag{2.12}$$

$$\langle \mathcal{A} \rangle_{\text{ens}} = \sum_{\Gamma} w_{\text{ens}}(\Gamma) \mathcal{A}(\Gamma) / \sum_{\Gamma} w_{\text{ens}}.$$
 (2.13)

The weight function is essentially a non-normalized form of $\rho_{ens}(\Gamma)$, with the partition function Q_{ens} (also called the sum over states) acting as the normalizing factor. Both w_{ens} and Q_{ens} contain an arbitrary multiplicative constant, whose choice corresponds to the definition of a zero of entropy. Q_{ens} is simply a function of the macroscopic properties defining the ensemble, and connection with classical thermodynamics is made by defining a thermodynamic potential Ψ_{ens} (see e.g. McQuarrie, 1976)

$$\Psi_{\rm ens} = -\ln Q_{\rm ens}.\tag{2.14}$$

This is the function that has a minimum value at thermodynamic equilibrium. For example, Ψ_{ens} might be the negative of the entropy *S* for a system at constant *NVE*, where *V* is the volume and *E* the total internal energy, or the Gibbs function *G* for a constant-*NPT* system, where *P* is the pressure and *T* the temperature.

Throughout the foregoing discussion, although we have occasionally used the language of classical mechanics, we have assumed that the states are discrete (e.g. a set of quantum numbers) and that we may sum over them. If the system were enclosed in a container, there would be a countably infinite set of quantum states. In the classical approximation, Γ represents the set of (continuously variable) particle positions and momenta, and we should replace the summation by a classical phase-space integral. w_{ens} and Q_{ens} are then usually defined with appropriate factors included to make them dimensionless, and to match up with the usual semiclassical 'coarse-grained' phase-space volume elements. On a computer, of course, all numbers are held to a finite precision and so, technically, positions and momenta are represented by discrete, not continuous, variables; we now have a countable and finite set of states. We assume that the distinction between this case and the classical limit is of no practical importance, and will use whichever representation is most convenient.