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, no less than 488 molecules appear on the cube faces. Whether or not the cube is surrounded by a containing wall, molecules on the surface will experience quite different forces from molecules in the bulk.

1.5.2 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. 1.9. The duplicate boxes are labelled A, B, C, etc., in an
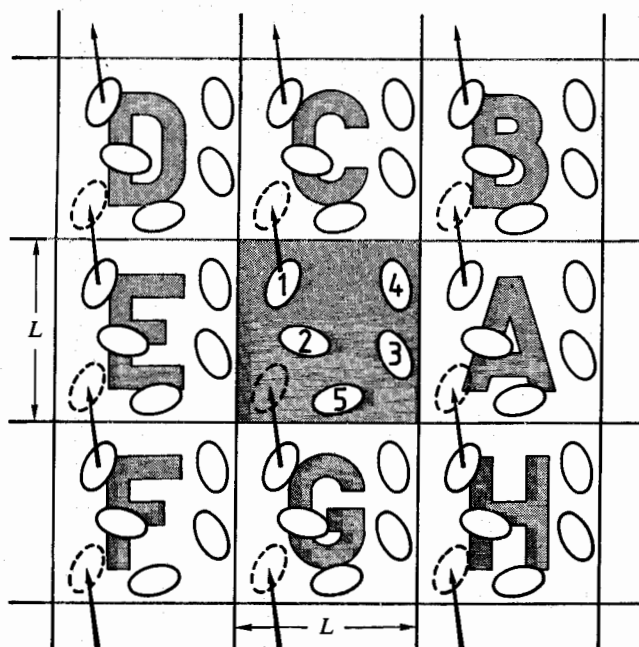

Fig. 1.9 A two-dimensional periodic system. Molecules can enter and leave each box across each of the four edges. In a three-dimensional example, molecules would be free to cross any of the six cube faces.

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 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 ranged (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. Recent work has shown that, even in the case of short-range potentials, the periodic boundary conditions can induce anisotropies in the fluid structure [Mandell 1976; Impey, Madden, and Tildesley 1981]. These effects are pronounced for small system sizes ($N \approx 100$) 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 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 to isotropic transition in liquid crystals [Luckhurst and Simpson 1982] and the solid to plastic crystal transition for N_2 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 wave-vectors (i.e.

$\mathbf{k} = (k_x, k_y, k_z)2\pi/L$, where k_x, k_y, k_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 above 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. 1.10. 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 6).
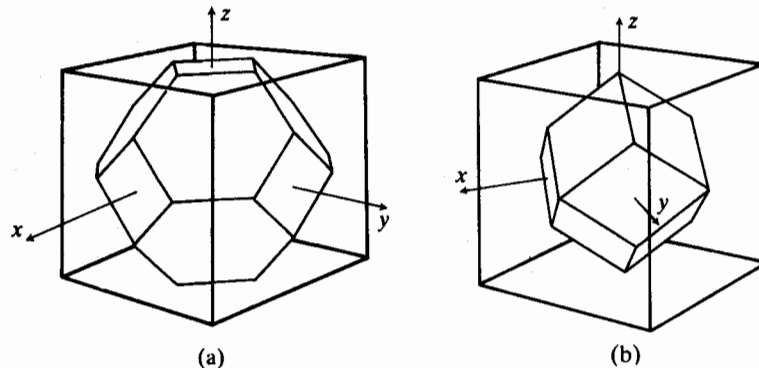
Fig. 1.10 Non-cubic simulation boxes. (a) The truncated octahedron and its containing cube; (b) the rhombic dodecahedron and its containing cube. The axes are those used in microfiche F.1.

So far, we have tacitly assumed that there is no external potential, i.e. no v_1 term in eqns (1.4), (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. 1.11, is periodic in the plane of the surface. In the z-direction, the graphite surface
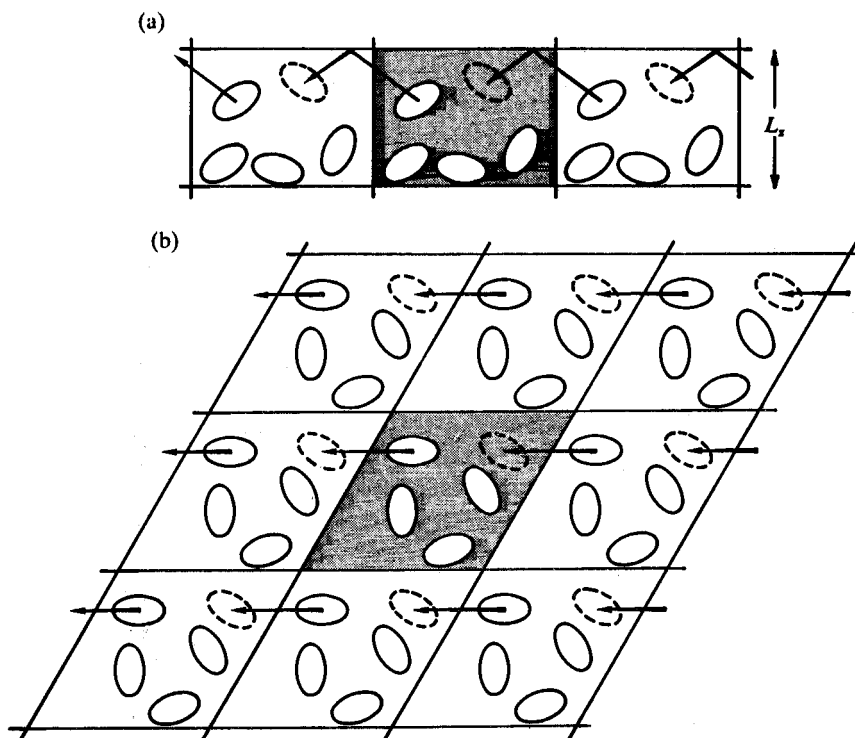
Fig. 1.11 Periodic boundaries used in the simulation of adsorption [Severin and Tildesley 1980]. (a) A side view of the box. There is a reflecting boundary at a height L_z. (b) A top view, showing the rhombic shape (i.e. the same geometry as the underlying graphite lattice).

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].

1.5.3 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. 1.12). 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. 1.12 molecule 1 interacts with
molecules 2, 3_E, 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
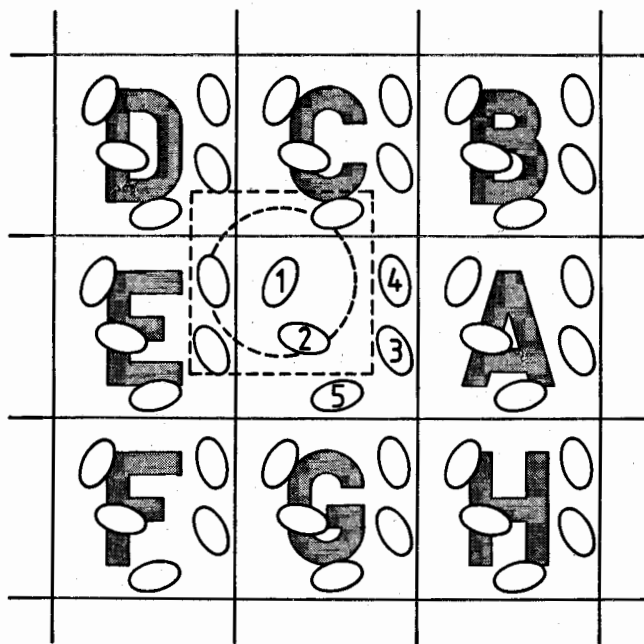
Fig. 1.12 The minimum image convention in a two-dimensional system. The central box contains
five molecules. The 'box' constructed with molecule 1 at its centre also contains five molecules. The
dashed circle represents a potential cutoff.

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 $r \geqslant r_c$, where r_c is the cutoff distance. The dashed circle in Fig. 1.12 represents a cutoff, and in this case molecules 2 and 4_E contribute to the force on 1, since their centres lie inside the cutoff, whereas molecules 3_E and 5_C do 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 $r_c = 2.5\sigma$ is just 1.6 per cent 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. 1.10, 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 $N = 197$ molecules in a truncated octahedron, or just $N = 181$ molecules in a rhombic dodecahedron.

1.5.4 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, i.e. 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 a FORTRAN IF statement to test the positions immediately after the molecules have been moved (whether by MC or MD):

```
IF ( RX(I) .GT.  BOXL2 ) RX(I) = RX(I) - BOXL
IF ( RX(I) .LT. -BOXL2 ) RX(I) = RX(I) + BOXL
```

Here, BOXL is a variable containing the box length L, and BOXL2 is just half the box length. Similar statements are applied to the y and z coordinates. An alternative to the IF statement is to use FORTRAN arithmetic functions to calculate the correct number of box lengths to be added or subtracted:

```
RX(I) = RX(I) - BOXL * ANINT ( RX(I) / BOXL )
```

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. 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 6, 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.

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

```
RXIJ = RXIJ - BOXL * ANINT ( RXIJ / BOXL )
RYIJ = RYIJ - BOXL * ANINT ( RYIJ / BOXL )
RZIJ = RZIJ - BOXL * ANINT ( RZIJ / BOXL )
```

The above code is guaranteed to yield the minimum image vector, no matter how many 'box lengths' apart the original images may be.

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. Some workers define $L = 1$, others prefer to take $L = 2$. By setting $L = 1$, with particle coordinates nominally in the range $(-\frac{1}{2}, +\frac{1}{2})$, the minimum image correction above becomes

```
RXIJ = RXIJ - ANINT ( RXIJ )
RYIJ = RYIJ - ANINT ( RYIJ )
RZIJ = RZIJ - ANINT ( RZIJ )
```

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 some of the examples given in this book and on the attached microfiche.

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. The FORTRAN code for implementing the minimum image correction in the cases of the truncated octahedron and the rhombic dodecahedron are given in program F.1 [see also Adams, 1983a; Smith 1983]. The code for the rhombic dodecahedron is a little more complicated than the code for the truncated octahedron, and the gain small, so that the truncated octahedron is preferable. We also give on the microfiche the code for computing minimum image corrections in the two-dimensional rhombic box often used in surface simulations.

Now we turn to the implementation of a spherical cutoff, i.e. 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 FORTRAN variable RCUTSQ. After computing the minimum image intermolecular vector, the following statements would be employed:

```
RIJSQ = RXIJ ** 2 + RYIJ ** 2 + RZIJ ** 2

IF ( RIJSQ .LT. RCUTSQ ) THEN

    ... compute i-j interaction ...
    ... accumulate energy and forces ...

ENDIF
```

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 ( RXIJ ) .LT. RCUT ) THEN

   IF ( ABS ( RYIJ ) .LT. RCUT ) THEN

     IF ( ABS ( RZIJ ) .LT. RCUT ) THEN

       RIJSQ = RXIJ ** 2 + RYIJ ** 2 + RZIJ ** 2

       IF ( RIJSQ .LT. RCUTSQ ) THEN

           ... compute i-j interaction ...
           ... accumulate energy and forces ...

       ENDIF

     ENDIF

   ENDIF

ENDIF
```

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. A different approach is needed on a pipeline machine, since the IF statements may prevent vectorization of the inner loop. In this situation, it is generally simplest to compute all the minimum image interactions, and then set to zero the potential energy (and forces) arising from pairs separated by distances greater than r_c. The extra work involved here is more than offset by the speed increase on vectorization. Within the inner loop, this is simply achieved by setting the inverse squared separation $1/r_{ij}^2$ to zero, where appropriate, before calculating energies, forces etc. as functions of this quantity. The following code performs this task on the CRAY 1-S [Fincham and Ralston 1981]

```
RIJSQ  = RXIJ ** 2 + RYIJ ** 2 + RZIJ ** 2
RIJSQI = 1.0 / RIJSQ
RIJSQI = CVMGP ( RIJSQI, 0.0, RCUTSQ - RIJSQ )

... compute i-j interaction ...
... as functions of RIJSQI  ...
```

The function CVMGP(A, B, C) is a vector merge statement which returns the value A if C is non-negative and the value B otherwise. Note that no time is saved by using a spherical cutoff in this way on a pipeline machine. The only reason for implementing the spherical cutoff in this case is so that the usual long-range corrections may be applied to the simulation results (see Chapter 2). In Chapter 5 we discuss the more complicated time-saving tricks used in the simulations of large systems.

1.5.5 Spherical boundary conditions

Before leaving this section, we should mention 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, Levesque, and Weis 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; in general, a periodic lattice is not consistent with spherical boundaries, and so the liquid state will be favoured in most simulations using this technique.