

第六章 分子动力学方法

6.1 引言

对于一个多粒子体系的实验观测物理量的数值可以由总的平均得到。但是由于实验体系又非常大，我们不可能计算求得所有涉及到的态的物理量数值的总平均。按照产生位形变化的方法，我们有**两类方法**对有限的一系列态的物理量做统计平均：

第一类是随机模拟方法。它是实现 Gibbs 的统计力学途径。在此方法中，体系位形的转变是通过马尔科夫(Markov)过程，由随机性的演化引起的。这里的马尔科夫过程相当于是内禀动力学在概率方面的对应物。该方法可以被用到没有任何内禀动力学模型体系的模拟上。随机模拟方法计算的程序简单，占内存少，但是该方法难于处理非平衡态的问题。

另一类为确定性模拟方法，即统计物理中的所谓分子动力学方法（Molecular Dynamics Method）。这种方法广泛地用于研究经典的多粒子体系的研究中。该方法是按该体系内部的内禀动力学规律来计算并确定位形的转变。它首先需要建立一组分子的运动方程，并通过直接对系统中的一个个分子运动方程进行数值求解，得到每个时刻各个分子的坐标与动量，即在相空间的运动轨迹，再利用统计计算方法得到多体系统的静态和动态特性，从而得到系统的宏观性质。因此，分子动力学模拟方法可以看作是体系在一段时间内的发展过程的模拟。在这样的处理过程中我们可以看出：分子动力学方法中不存在任何随机因素。

系统的动力学机制决定运动方程的形式：

在分子动力学方法处理过程中，方程组的建立是通过对物理体系的微观数学描述给出的。在这个微观的物理体系中，每个分子都各自服从经典的牛顿力学。每个分子运动的内禀动力学是用理论力学上的哈密顿量或者拉格朗日量来描述，也可以直接用牛顿运动方程来描述。这种方法可以处理与时间有关的过程，因而可以处理非平衡态问题。但是使用该方法的程序较复杂，计算量大，占内存也多。

适用范围广泛：

原则上，分子动力学方法所适用的微观物理体系并无什么限制。这个方法适用的体系既可以是少体系统，也可以是多体系统；既可以是点粒子体系，也可以是具有内部结构的体系；处理的微观客体既可以是分子，也可以是其它的微观粒子。

自五十年代中期开始，分子动力学方法得到了广泛的应用。它与蒙特卡洛方法一起已经成为计算机模拟的重要方法。应用分子动力学方法取得了许多重要成果，例如气体或液体的状态方程、相变问题、吸附问题等，以及非平衡过程的研究。其应用已从化学反应、生物学的蛋白质到重离子碰撞等广泛的学科研究领域。

实际使用的限制:

实际上，分子动力学模拟方法和随机模拟方法一样都面临着两个基本限制：一个是有限观测时间的限制；另一个是有限系统大小的限制。通常人们感兴趣的是体系在热力学极限下（即粒子数目趋于无穷时）的性质。但是计算机模拟允许的体系大小要比热力学极限小得多，因此可能会出现有限尺寸效应。为了减小有限尺寸效应，人们往往引入周期性、全反射、漫反射等边界条件。当然边界条件的引入显然会影响体系的某些性质。

数值求解时的离散化方法:

对体系的分子运动方程组采用计算机进行数值求解时，需要将运动方程离散化为有限差分方程(参见第四章)。常用的求解方法有欧拉法、龙格-库塔法、辛普生法等（参见附录 D）。数值计算的误差阶数显然取决于所采用的数值求解方法的近似阶数。原则上，只要计算机计算速度足够大，内存足够多，我们可以使计算误差足够小。

6. 2 分子动力学基础知识

一、分子运动方程及其数值求解

采用分子动力学方法时，必须对一组分子运动微分方程做数值求解。从计算数学的角度来看，这是个求一个初值问题的微分方程的解。实际上计算数学为了求解这种问题已经发展了许多的算法，但是并不是所有的这些算法都可以用来解决物理问题。

例子：

以一个一维谐振子为例，来看一下如何用计算机数值计算方法求解初值问题。一维谐振子的经典哈密顿量为：

$$H = \frac{p^2}{2m} + \frac{1}{2}kx^2. \quad (6. 2. 1)$$

这里的哈密顿量（即能量）为守恒量。

假定初始条件为 $x(0), p(0)$ ，则它的哈密顿方程是对时间的一阶微分方程

$$\frac{dx}{dt} = \frac{\partial H}{\partial p} = \frac{p}{m}, \quad \frac{dp}{dt} = -\frac{\partial H}{\partial x} = -kx. \quad (6.2.2)$$

计算在相空间中的运动轨迹 $(x(t), p(t))$ ：采用有限差分法，将微分方程变为有限差分方程，以便在计算机上做数值求解，并得到空间坐标和动量随时间的演化关系。

首先，取差分计算的时间步长为 h ，采用一阶微分形式的向前差商表示，它是直接运用展开到 h 的一阶泰勒展开公式

$$f(t+h) = f(t) + h \frac{df}{dt} + O(h^2),$$

即得到

$$\frac{df}{dt} \approx \frac{f(t+h) - f(t)}{h}, \quad (6.2.3)$$

则微分方程 (6.2.2) 可以被改写为差分形式

$$\frac{dx}{dt} = \frac{x(t+h) - x(t)}{h} = \frac{p(t)}{m}. \quad (6.2.4)$$

$$\frac{dp}{dt} = \frac{p(t+h) - p(t)}{h} = -kx(t). \quad (6.2.5)$$

将上面两个公式整理后，我们得到解微分方程 (6.2.2) 的欧拉 (Euler) 算法（参见附录 C）：

$$x(t+h) = x(t) + \frac{hp(t)}{m}. \quad (6.2.6)$$

$$p(t+h) = p(t) - hkx(t). \quad (6.2.7)$$

这是 $x(t), p(t)$ 的一组递推公式。有了初始条件 $x(0), p(0)$ ，就可以一步一步地使用前一时刻的坐标、动量值确定下一时刻的坐标、动量值。这个方法是一步法的典型例子。

由于在实际数值计算时 h 的大小是有限的，因而在上述算法中微分被离散化为差分形式来计算时总是有误差的。可以证明一步法的局部离散化误差与总体误差是相等的，都为 $O(h^2)$ 的量级。在实际应用中，适当地选择 h 的大小是十分重要的。 h 取得太大，得到的结果偏离也大，甚至于连能量都不守恒； h 取得太小，有可能结果仍然不够好。这就要求我们改进计算方法，进一步考虑**二步法**。

差分计算的二步法：

实际上泰勒展开式的一般形式为

$$f(t+h) = f(t) + \sum_{i=1}^n \frac{h^i}{i!} f^{(i)}(t) + O(h^{n+1}). \quad (6.2.8)$$

其中 $O(h^{n+1})$ 表示误差的数量级。前面叙述的欧拉算法就是取 $n=1$ 。

现在考虑公式(6.2.8)中直到含 h 的二次项的展开（即取 $n=2$ ），则得到

$$f(t+h) = f(t) + h \frac{df}{dt} + \frac{h^2}{2} \frac{d^2 f}{dx^2} + O(h^3). \quad (6.2.9)$$

$$f(t-h) = f(t) - h \frac{df}{dt} + \frac{h^2}{2} \frac{d^2 f}{dx^2} + O(h^3). \quad (6.2.10)$$

将上面两式相加、减得到含二阶和一阶导数的公式

$$\frac{d^2 f}{dx^2} = \frac{1}{h^2} [f(t+h) - 2f(t) + f(t-h)]. \quad (6.2.11)$$

$$\frac{df}{dt} = \frac{f(t+h) - f(t-h)}{2h}. \quad (6.2.12)$$

令 $f(t) = x(t)$ ，利用牛顿第二定律公式 $F(t) = m \frac{d^2 x}{dt^2}$ ，公式 (6.2.11) 写为坐标的递推公式

$$x(t+h) = 2x(t) - x(t-h) + h^2 \frac{F(t)}{m} . \quad (6.2.13)$$

公式 (6.2.12) 写为计算动量的公式得到

$$p(t) = m\dot{x}(t) = mv(t) = \frac{m}{2h} [x(t+h) - x(t-h)] . \quad (6.2.14)$$

这样我们就推导出了一个比 (6.2.6) 和 (6.2.7) 更精确的递推公式。这是二步法的一种，称为 **Verlet 方法**。

当然我们还可以建立更高阶的多步算法，然而大部分更高阶的方法所需要的内存比一步法和二步法所需要的大得多，并且有些更高阶的方法还需要用迭代来解出隐式给定的变量，内存的需求量就更大。并且当今的计算机都仅仅只有有限的内存，因而并不是所有的高阶算法都适用于物理系统的计算机计算。**Verlet 算法是分子动力学模拟中求解常微分方程最通用的方法。**

二、多体系统的基本概念与分子动力学方法

N 体系统中，一个 n 体的密度函数一般可以写为

$$\rho_n(\vec{r}_1, \vec{r}_2, \dots, \vec{r}_n) = \frac{1}{\mathbf{Z}} \frac{N!}{(N-n)!} \int W(\vec{R}) d\vec{r}_{n+1} d\vec{r}_{n+2} \dots d\vec{r}_N, \quad (6.2.15)$$

其中 $W(\vec{R})$ 是描写系统的几率函数， $\mathbf{Z} = \int W(\vec{R}) d\vec{R}$ 为系统的配分函数， \vec{R} 通常为由系统中所有粒子的坐标、动量构成的相空间中的任意一点。在 $n=1$ 的情况下粒子密度函数为 $\rho(\vec{r}) = \rho_1(\vec{r})$ 。

两体密度函数与对关联函数 $g(\vec{r}' - \vec{r})$ 相关，即

$$\rho_2(\vec{r}, \vec{r}') = \rho(\vec{r}) g(\vec{r}' - \vec{r}) \rho(\vec{r}'). \quad (6.2.16)$$

式中的 $g(\vec{r}' - \vec{r})$ 就是对关联函数，它是描述与时间无关的粒子间关联性的量度。

$g(\vec{r}' - \vec{r})$ 的物理意义是：当在空间 \vec{r} 处有一个粒子时，在另一个空间位置 \vec{r}' 的点周围单位体积内发现另一个粒子的几率。

能够很容易得到

$$\rho_2(\vec{r}, \vec{r}') = \langle \hat{\rho}(\vec{r}) \hat{\rho}(\vec{r}') \rangle - \delta(\vec{r}' - \vec{r}) \rho(\vec{r}). \quad (6.2.17)$$

其中公式右边第一项叫做密度关联函数。 $\hat{\rho}(\vec{r})$ 为密度算符，其定义为

$$\hat{\rho}(\vec{r}) = \sum_{i=1}^N \delta(\vec{r} - \vec{r}_i). \quad (6.2.18)$$

系统的密度为密度算符的平均值，

$$\rho(\vec{r}) = \langle \hat{\rho}(\vec{r}) \rangle. \quad (6.2.19)$$

如果系统的密度接近一个常数，对关联函数 $g(\vec{r}' - \vec{r})$ 可以导出一个简单的形式

$$g(\vec{r}) = \frac{1}{\rho N} \left\langle \sum_{i \neq j}^N \delta(\vec{r} - \vec{r}_{ij}) \right\rangle, \quad (6.2.20)$$

式中 ρ 是 $\vec{r}' = 0$ 和 \vec{r} 点的密度的平均。

对球坐标的方位角 θ 和极角 φ 求平均后，得到径向对关联函数为

$$g(r) = \frac{1}{2\pi} \int g(\vec{r}) \sin \theta \, d\theta \, d\varphi, \quad (6.2.21)$$

在分子动力学模拟的数值计算中，在空间某点上的密度函数 $\rho(\vec{r})$ 可由下式计算得到：

$$\rho(\vec{r}) \approx \frac{\langle N(\vec{r}, \Delta r) \rangle}{\Omega(\vec{r}, \Delta r)}, \quad (6.2.22)$$

其中 $\Omega(\vec{r}, \Delta r)$ 为原点在距离球坐标中心 \vec{r} 处，半径为 Δr 的球体积。 $N(\vec{r}, \Delta r)$ 为在该体积内的粒子数。这里我们可以通过调整半径 Δr ，来得到特定系统的平滑、真实的密度分布函数 $\rho(\vec{r})$ 。上式中的求平均是对时间步所做的。

采用类似的方法，可以得到径向对关联函数的数值。若 \mathbf{r} 是从一个特定粒子位置 \vec{r}_i 点为原点的球半径，径向对关联分布函数 $g(r)$ 就是另一个粒子在距离 r 处出现的几率。由此可以算出，

$$g(r) \approx \frac{\Delta N(\vec{r}, \Delta r)}{\rho 4\pi r^2 \Delta r}, \quad (6.2.23)$$

$\Delta N(\vec{r}, \Delta r)$ 为在以 \vec{r}_i 为球心， r 为半径， Δr 厚的球壳内的粒子数。

分子动力学元胞：

分子动力学模拟方法往往用于研究大块物质在给定密度下的性质，而实际计算模拟不可能在几乎是无穷大的系统中进行。所以必须引进一个叫做分子动力学元胞的体积元，以维持一个恒定的密度。对气体和液体，如果所占体积足够大，并且系统处于热平衡状态的情况下，那么这个体积的形状是无关紧要的[1]。对于晶态的系统，元胞的形状是有影响的。为了计算简便，对于气体和液体，我们取一个立方形的体积为分子动力学元胞。设分子动力学元胞的线度大小为 L ，则其体积为 L^3 。由于引进这样的立方体箱子，将产生六个我们不希望出现的表面。模拟中碰撞这些箱的表面的粒子应当被反射回到元胞内部，特别是对粒子数目很少的系统。然而这些表面的存在对系统的任何一种性质都会有重大的影响。

元胞的周期性边界条件:

为了将分子动力学元胞有限立方体内的模拟, 扩展到真实大系统的模拟, 我们通常采用周期性边界条件。采用这种边界条件, 我们就可以消除引入元胞后的表面效应, 构造出一个准无穷大的体积来更精确地代表宏观系统。实际上, 这里我们做了一个假定, 即让这个小体积元胞镶嵌在一个无穷大的大块物质之中。

周期性边界条件的数学表示形式为

$$A(\vec{x}) = A(\vec{x} + \vec{n}L), \quad \vec{n} = (n_1, n_2, n_3). \quad (6.2.24)$$

其中 A 为任意的可观测量, n_1, n_2, n_3 为任意整数。这个边界条件就是命令基本分子动力学元胞完全等地重复无穷多次。

该边界条件的具体实现是这样操作的: 当有一个粒子穿过基本分子动力学元胞的六方体表面时, 就让这个粒子以相同的速度穿过此表面对面的表面重新进入分子动力学元胞内。

不同分子动力学元胞盒子内粒子间的相互作用：

对于不同分子动力学元胞盒子内粒子间的相互作用，如果相互作用是短程力，我们可以在长度 r_c 处截断。这里 $V(r_c)$ 必须要足够小，以使截断不会显著地影响模拟结果。典型的分子动力学元胞尺度 L 通常选得比 r_c 大很多。我们往往选择元胞尺度满足不等式条件 $L/2 > r_c$ ，使得距离大于 $L/2$ 的粒子的相互作用可以忽略，以避免有限尺寸效应。通常 L 的数值应当选得很大。

在考虑粒子间的相互作用时，通常采用**最小像力约定**。最小像力约定是在由无穷重复的分子动力学基本元胞中，每一个粒子只同它所在的基本元胞内的另外 $N-1$ 个中（设在此元胞内有 N 个粒子）的每个粒子或其最邻近的影像粒子发生相互作用。

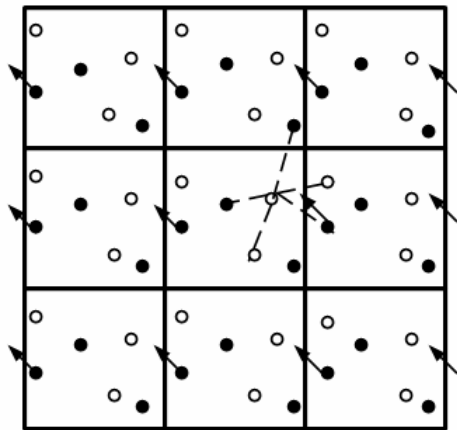


图 6.2.1 分子动力学模拟的最小像力约定

如图 6.2.1 所示，其中一个白色的粒子通过图上虚线连线，与它所在元胞内的其它粒子或其影像粒子相互作用。如果 \vec{r}_i 处的粒子 i 同 \vec{r}_j 处的粒子 j 之间的距离为

$$r_{ij} = \min(|\vec{r}_i - \vec{r}_j + \vec{n}L|), \quad (\text{对一切的 } \vec{n}). \quad (6.2.25)$$

实际上这个约定就是通过满足不等式条件 $r_c < L/2$ 来截断位势。

采用最小像力约定后，元胞内第*i*个粒子与周围粒子的相互作用势和相互作用力为

$$\begin{aligned} U_i(\vec{R}) &= \sum_{j=1, N}^{i \neq j} u(r_{ij}) \\ \vec{F}_i(\vec{R}) &= \sum_{j=1, N}^{i \neq j} F(r_{ij}) \hat{r}_{ij} . \end{aligned} \tag{6.2.26}$$

$\vec{R} = \{\vec{r}_1, \vec{r}_2, \dots, \vec{r}_N\}$ 表示元胞内所有粒子的坐标。 \hat{r}_{ij} 是沿 $\vec{r}_j - \vec{r}_i$ 方向的单位矢量。采用最小像力约定会使得在截断处粒子的受力有一个 δ -函数的奇异性，这会给模拟计算带来误差。为减小这种误差，我们总可以将相互作用势能移到 $V(r) - V(r_c)$ ，以保证在截断处相互作用为零。

6. 3 分子动力学模拟的基本步骤

在计算机上对分子系统的分子动力学模拟的实际步骤可以划分为四步：

首先是设定模拟所采用的模型；第二，给定初始条件；第三，趋于平衡的计算过程；最后是宏观物理量的计算。

1. 模拟模型的设定

设定模型是分子动力学模拟的第一步工作。例如在一个分子系统中，假定两个分子间的相互作用势为硬球势，其势函数表示为

$$U(r) = \begin{cases} +\infty, & \text{如果 } r < \sigma, \\ 0, & \text{如果 } r \geq \sigma. \end{cases} \quad (6.3.1)$$

实际上，更常用的是图(6.3.2)所示的 Lennard-Jones 型势。

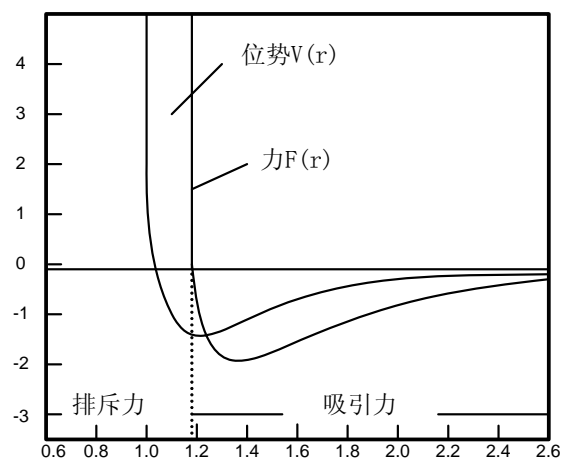


图 6.3.2 Lennard-Jones 势

它的势函数表示为

$$U(r) = 4\varepsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right]. \quad (6.3.2)$$

其中 $-\varepsilon$ 是位势的最小值（ ε 可以确定能量的单位），这个最小值出现在距离 r 等于 $2^{1/6}\sigma$ 的地方。
 $r = \sigma$ 时位势为零。

在 Lennard-Jones 型势作用下, 第 i 个粒子与元胞内其它 $N-1$ 个粒子或其最邻近的影像粒子的相互作用力在 x 方向的分量为

$$F_{i,x} = 48 \left(\frac{\varepsilon}{\sigma^2} \right) \sum_{j \neq i}^N (x_i - x_j) \left[\left(\frac{\sigma}{r_{ij}} \right)^{14} - \frac{1}{2} \left(\frac{\sigma}{r_{ij}} \right)^8 \right], \quad (6.3.3)$$

它的 y 和 z 分量的表示式与上式相似。

模型确定后, 根据经典物理学的规律我们就可以知道在系综模拟中的守恒量。例如对在微正则系综的模拟中能量、动量和角动量均为守恒量。在此系综中他们分别表示为

$$E = \sum_i \left[\frac{1}{2} m (\dot{\vec{r}}_i)^2 + V(\vec{r}_i) \right]. \quad (6.3.4)$$

$$\vec{P} = \sum_i \vec{p}_i. \quad (6.3.5)$$

$$\vec{M} = \sum_i \vec{r}_i \times \vec{p}_i. \quad (6.3.6)$$

其中 $\vec{p}_i = m \dot{\vec{r}}_i$ 。

为了计算方便，取分子动力学元胞为立方形，元胞的线度大小为 L ，则其体积为 L^3 。根据给定密度 ρ 和指定的单个元胞中的粒子数 N ，确定出元胞的 L 值。采用周期性边界条件和最小像力约定。周期性边界条件表示形式见式（6.2.24）。最小像力约定下，元胞内第 i 个粒子所受力参见公式（6.2.26）

2. 给定初始条件

分子动力学模拟的过程进入对系统微分方程组做数值求解时，需要知道粒子的初始位置和速度的数值。不同的算法要求不同的初始条件。

例如，Verlet 方法需要两组坐标来启动计算：一组是零时刻的坐标，另一组是前进一个时间步长时的坐标，或者是一组零时刻的速度值。

但是，一般来说系统的初始条件都是不可能知道的。表面上看这是一个难题。实际上，精确选择待求系统的初始条件是没有意义的，因为模拟时间足够长时，系统就会忘掉初始条件。但是初始条件的合理选择将可以加快系统趋于平衡。

常用的初始条件可以选择为：（1）令初始位置在差分划分网格的格子上，初始速度则从玻尔兹曼分布随机抽样得到。（2）令初始位置随机地偏离差分划分网格的格子，初始速度为零。（3）令初始位置随机地偏离差分划分网格的格子，初始速度也是从玻尔兹曼分布随机抽样得到。

3. 趋于平衡

按照上面给出的运动方程、边界条件和初始条件，就可以进行分子动力学模拟计算。但是，这样计算出的系统不会具有所要求的系统能量，并且这个状态本身也还不是一个平衡态。为了使系统达到平衡，模拟中需要一个趋衡过程。在这个过程中，我们增加或从系统中移出能量，直到系统具有所要求的能量。然后，再对运动方程中的时间向前积分若干步，使系统持续给出确定能量值。我们称这时系统已经达到平衡态。这段达到平衡所需的时间称为弛豫时间。

在分子动力学模拟中，时间步长 h 的大小选择是十分重要的。它决定了模拟所需要的时间。为了减小误差，步长 h 必须取得小一些；但是取得太小，系统模拟的弛豫时间就很长。这里需要积累一定的模拟经验，选择适当的时间步长 h 。

4. 宏观物理量的计算

实际计算宏观物理量往往是在分子动力学模拟的最后阶段进行的。它是沿着相空间轨迹求平均来计算得到的。

例如对于一个宏观物理量 A ，它的测量值为平均值 \bar{A} 。如果已知初始位置和动量为 $\{\vec{r}^{(N)}(0)\}$ 和 $\{\vec{p}^{(N)}(0)\}$ （上标 N 表示系综 N 个粒子的对应坐标和动量参数），选择某种分子动力学算法求解具有初值问题的运动方程，便得到相空间轨迹 $(\{\vec{r}^{(N)}(t)\}, \{\vec{p}^{(N)}(t)\})$ 。

对轨迹平均的宏观物理量 A 的表示为

$$\bar{A} = \lim_{t' \rightarrow \infty} \frac{1}{(t' - t_0)} \int_{t_0}^{t'} d\tau A(\{\vec{r}^{(N)}(\tau)\}, \{\vec{p}^{(N)}(\tau)\}). \quad (6.3.7)$$

如果宏观物理量为动能，它的平均为

$$\bar{E}_k = \lim_{t' \rightarrow \infty} \frac{1}{(t' - t_0)} \int_{t_0}^{t'} d\tau E_k(\{\vec{p}^{(N)}(\tau)\}). \quad (6.3.8)$$

由于在模拟过程中计算出的动能值是在不连续的路径上的值，因此公式(6.3.8)可以表示为在时间的各个间断点 μ 上计算动能的平均值

$$\bar{E}_k = \frac{1}{n - n_0} \sum_{\mu > n_0}^n \sum_{i=1}^N \frac{(p_i^2)^{(\mu)}}{2m}. \quad (6.3.9)$$

在分子动力学模拟过程中，**温度是需要加以监测的物理量**，特别是在模拟的起始阶段。根据能量均分定理，我们可以从平均动能值计算得到温度值，

$$T = \frac{\bar{E}_k}{\frac{d}{2} N k_B}. \quad (6.3.10)$$

其中 d 为每个粒子的自由度，如果不考虑系统所受的约束，则 $d = 3$ 。

系统内部的位势能量的平均值为：

$$\bar{U} = \frac{1}{n - n_0} \sum_{\mu > n_0}^n \sum_{i < j} u(r_{ij}^{(\mu)}). \quad (6.3.11)$$

假定位势在 r_c 处被截断，那么上式计算出的势能以及由此得到的总能量就包含有误差。

为了对此偏差做出修正，采用对关联函数来表示位能

$$U / N = 2\pi\rho \int_0^\infty u(r)g(r)r^2 dr. \quad (6.3.12)$$

若 $n(r)$ 为距离原点 r 到 $r + \Delta r$ 之间的平均粒子数，参照公式(6.2.23)可以得到

$$g(r) = \frac{V}{N} \frac{n(r)}{4\pi r^2 \Delta r}. \quad (6.3.13)$$

在分子动力学模拟过程中，所有的距离已经在力的计算中得到，因而很容易计算对关联函数的值。

图 6.3.3 为由计算机模拟得到的两组不同参数下的对关联函数的例子。

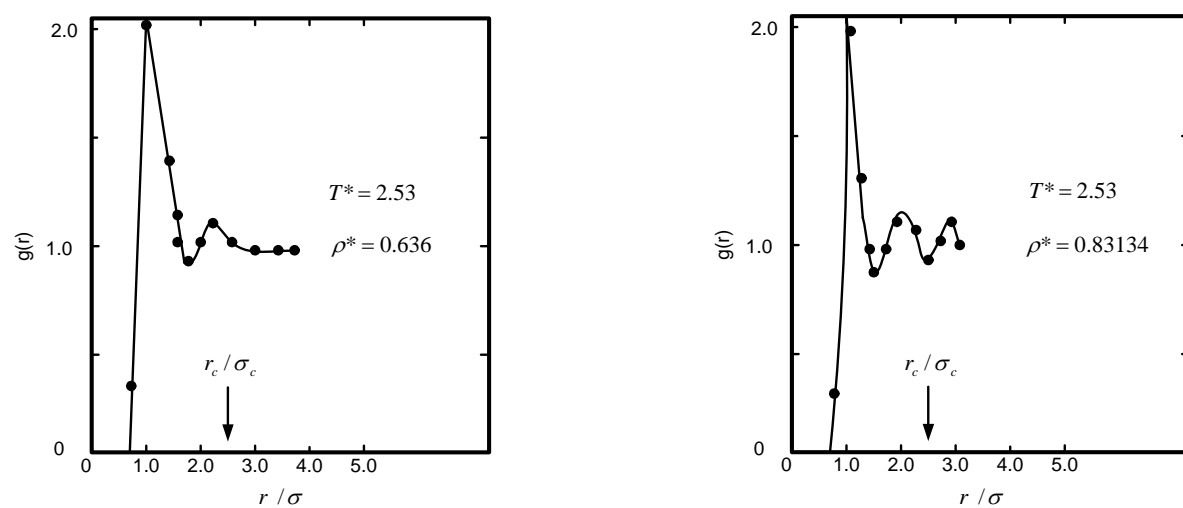


图 6.3.3 由计算机模拟得到的两组不同参数下的对关联函数。

由于位势的截断，对关联函数仅对 $r_c < L/2$ 以下的距离有意义。

在公式 (6.3.11) 中，所有的位能都加到截断距离为止，尾部修正可以取为

$$U_c = 2\pi\rho \int_{r_c}^{\infty} u(r)g(r)r^2 dr . \quad (6.3.14)$$

压强可以通过计算在面积元 dA 的法线方向上净动量转移的时间平均值来得到，也可以利用含对关联函数的维里状态方程计算。该维里状态方程可以写为

$$P = \rho k_B T - \frac{\rho^2}{6} \int_0^{\infty} g(r) \frac{\partial u}{\partial r} 4\pi r^3 dr . \quad (6.3.15)$$

在上式的计算中，把积分划分为两项，一项是由相互作用力程之内的贡献引起的，一项是对位势截断的改正项：

$$P = \rho k_B T - \overline{\frac{\rho^2}{6N} \sum_{i < j} r_{ij} \frac{\partial u}{\partial r_{ij}}} - P_c. \quad (6.3.16)$$

其中长程改正项为：

$$P_c = \frac{\rho^2}{6} \int_{r_c}^{\infty} g(r) \frac{\partial u}{\partial r} 4\pi r^3 dr. \quad (6.3.17)$$

6.4 平衡态分子动力学模拟

在经典分子动力学模拟方法的应用当中，存在着对两种系统状态的分子动力学模拟。一种是对平衡态的 MD 模拟，另一类是对非平衡态的分子动力学模拟。

对平衡态系综分子动力学模拟又可以分为如下类型：微正则系综的分子动力学(NVE)模拟，正则系综的分子动力学(NVT)模拟，等温等压系综分子动力学(NPT)模拟和等焓等压系综分子动力学(NPH)模拟等。

下面我们仅对平衡态的分子动力学方法中前两类模拟做简单的介绍。

1. 微正则系综的分子动力学模拟

在进行对微正则系综的分子动力学模拟时，首先我们要确定所采用的相互作用模型。

我们假定一个孤立的多粒子体系，其粒子间的相互作用位势是球对称的，则其哈密顿量可以写为

$$H = \frac{1}{2} \sum_i \frac{p_i^2}{m} + \sum_{i < j} u(r_{ij}). \quad (6.4.1)$$

其中 r_{ij} 是第 i 个粒子与第 j 个粒子之间的距离。

该微正则系综特点：由于这个系统的哈密顿量中不显式地出现时间关联，因而系统的能量是个守恒量。系统的体积和粒子数也是不变的。此外，由于整个系统并未运动，所以整个系统的总动量 \vec{P} 恒等于零。这就是系统受到的四个约束。

由该系统的哈密顿量可以推导出牛顿方程形式的运动方程组

$$\frac{d^2 \vec{r}_i(t)}{dt^2} = \frac{1}{m} \sum_{i < j} \vec{F}_i(r_{ij}), \quad (i = 1, 2, \dots, N). \quad (6.4.2)$$

用 Verlet 方法数值求解 (6.4.2) 微分方程组，方程组 (6.4.2) 的求解变成求解方程组：

$$\vec{r}_i(t+h) = 2\vec{r}_i(t) - \vec{r}_i(t-h) + \vec{F}_i(t)h^2 / m, \quad (i = 1, 2, \dots, N). \quad (6.4.3)$$

该方程组反映出：从前面 t 和 $t-h$ 时刻这两步的空间坐标位置及 t 时刻的作用力，就可以算出下一步 $t+h$ 时刻的坐标位置。

为了将 (6.4.3) 式写成更简洁的形式，我们令

$$t_n = nh, \quad \vec{r}_i^{(n)} = \vec{r}_i(t_n), \quad \vec{F}_i^{(n)} = \vec{F}_i(t_n). \quad (6.4.4)$$

则从 (6. 4. 3) 式可以得到如下差分方程组的形式

$$\vec{r}_i^{(n+1)} = 2\vec{r}_i^{(n)} - \vec{r}_i^{(n-1)} + \vec{F}_i^{(n)} h^2 / m, \quad (i = 1, 2, \dots, N). \quad (6. 4. 5)$$

如果已知一组初始空间位置 $\{\vec{r}_i^{(0)}, \{\vec{r}_i^{(1)}\}$, 则通过求解方程组 (6. 4. 5) 一步步得到 $\{\vec{r}_i^{(2)}\}, \{\vec{r}_i^{(3)}\}, \dots$ 。

由空间坐标又可以算出粒子的运动速度为

$$\vec{v}_i^{(n)} = (\vec{r}_i^{(n+1)} - \vec{r}_i^{(n-1)}) / 2h. \quad (6. 4. 6)$$

这里在第 $n+1$ 步算出的速度是前一刻, 即第 n 步的速度。因而动能的计算比势能的计算落后一步。

根据上述原理我们可以将粒子数恒定、体积恒定、能量恒定的微正则系综 (NVE) 的分子动力学模拟步骤设计如下：

- (1) 给定初始空间位置 $\{\vec{r}_i^{(0)}\}, \{\vec{r}_i^{(1)}\}$, ($i = 1, 2, \dots, N$)。
- (2) 计算在第 n 步时粒子所受的力 $\{\vec{F}_i^{(n)}\}$: $\vec{F}_i^{(n)} = \vec{F}_i(t_n)$ 。
- (3) 利用公式: $\vec{r}_i^{(n+1)} = 2\vec{r}_i^{(n)} - \vec{r}_i^{(n-1)} + \vec{F}_i^{(n)} h^2 / m$, 计算在第 $n+1$ 步时所有粒子所处的空间位置 $\{\vec{r}_i^{(n+1)}\}$ 。
- (4) 计算第 n 步的速度: $\vec{v}_i^{(n)} = (\vec{r}_i^{(n+1)} - \vec{r}_i^{(n-1)}) / 2h$ 。
- (5) 返回到步骤 (2), 开始下一步的模拟计算。

用上述形式的 Verlet 算法, 动能的计算比势能的计算落后一步。此外, 这种算法不是自启动的。要真正求出微分方程组 (6.4.2) 的解, 除了需要给出初始空间位置 $\{\vec{r}_i^{(0)}\}$ 外, 还要求另外给出一组空间位置 $\{\vec{r}_i^{(1)}\}$ 。

改进后的计算方法：即把 N 个粒子的初始位置放在网格的格点上，然后加以扰动。如果初始条件是空间位置和速度，则采用下面的公式来计算空间位置 $\{\vec{r}_i^{(1)}\}$

$$\vec{r}_i^{(1)} = \vec{r}_i^{(0)} + h\vec{v}_i^{(0)} + \vec{F}_i^{(0)}h^2 / 2m. \quad (6.4.7)$$

然后再按上述模拟步骤进行计算。

Verlet 算法的速度变型形式将会使其数值计算的稳定性得到加强。下面我们就此做简单介绍。

令

$$\vec{z}_i^{(n)} = (\vec{r}_i^{(n+1)} - \vec{r}_i^{(n)}) / h. \quad (6.4.8)$$

则公式(6.4.5)写为

$$\begin{cases} \vec{r}_i^{(n)} = \vec{r}_i^{(n-1)} + h\vec{z}_i^{(n-1)} \\ \vec{z}_i^{(n)} = \vec{z}_i^{(n-1)} + m^{-1}h\vec{F}_i^{(n)}. \end{cases} \quad (6.4.9)$$

上式在数学上与(6.4.5)式是等价的，并称为相加形式。

由此 Verlet 算法的速度形式的模拟步骤可以表述为：

- (1) 给定初始空间位置 $\{\vec{r}_i^{(1)}\}$, ($i = 1, 2, \dots, N$)。
- (2) 给定初始速度 $\{\vec{v}_i^{(1)}\}$ 。
- (3) 利用: $\vec{r}_i^{(n+1)} = \vec{r}_i^{(n)} + h\vec{v}_i^{(n)} + \vec{F}_i^{(n)} h^2 / 2m$, 计算在第 $n+1$ 步时所有粒子所处的空间位置 $\{\vec{r}_i^{(n+1)}\}$ 。
- (4) 计算在第 $n+1$ 步时所有粒子的速度 $\{\vec{v}_i^{(n+1)}\}$:

$$\vec{v}_i^{(n+1)} = \vec{v}_i^{(n)} + h(\vec{F}_i^{(n+1)} + \vec{F}_i^{(n)}) / 2m .$$

- (5) 返回到步骤 (3), 开始第 $n+2$ 步的模拟计算。

Verlet 速度形式的算法比前一种算法好些。它不仅可以在计算中得到同一时间步上的空间位置和速度, 并且数值计算的稳定性也提高了。

一般情况下，对于给定能量的系统不可能给出精确的初始条件。这时需要先给出一个合理的初始条件，然后在模拟过程中逐渐调节系统能量达到给定值。

其步骤为：首先将运动方程组解出若干步的结果；然后计算出动能和位能；假如总能量不等于给定恒定值，则通过对速度的调整来实现能量守恒。也就是将速度乘以一个标度（scaling）因子，该因子一般取为

$$\beta = \left[\frac{T^*(N-1)}{16 \sum_i v_i^2} \right]^{1/2} . \quad (6.4.10)$$

然后再回到第一步，对下一时刻的运动方程求解。反复进行上面的过程，直到系统达到平衡。这样的模拟过程也称为平衡化阶段。

采用对速度标度的办法，可以使速度发生很大变化。为了消除可能带来的效应，必须要有足够的时间让系统再次建立平衡。在到达趋衡阶段以后，必须检验粒子的速度分布是否符合麦克斯韦-波尔兹曼分布。

(2) 正则系综的分子动力学模拟

在统计物理中的正则系综模拟是针对一个粒子数 N 、体积 V 、温度 T 和总动量 ($\vec{P} = \sum_i \vec{p}_i = 0$) 为守恒量的系综 (NVT)。

这种情况就如同一个系统置于热浴之中，此时系统的能量可能有涨落，但系统温度则已经保持恒定。在正则系综的 MD 模拟中施加的约束与微正则系综中的不一样。正则系综分子动力学方法是在运动方程组上加上动能恒定（即温度恒定）的约束，而不是象微正则系综的分子动力学模拟中对运动方程加上能量恒定的约束。

在正则系综分子动力学的平衡化过程中，速度标度（velocity scaling）因子一般选下面的形式较为合适

$$\beta = \left[\frac{(3N-4)kT}{\sum_i m v_i^2} \right]^{1/2}. \quad (6.4.11)$$

正则系综分子动力学的 Verlet 算法的速度形式的模拟具体步骤：

- (1) 给定初始空间位置 $\{\vec{r}_i^{(1)}\}$, ($i=1,2,\dots,N$),
- (2) 给定初始速度 $\{\vec{v}_i^{(1)}\}$,
- (3) 利用: $\vec{r}_i^{(n+1)} = \vec{r}_i^{(n)} + h\vec{v}_i^{(n)} + \vec{F}_i^{(n)} h^2 / 2m$ 计算在第 $n+1$ 步时所有粒子所处的空间位置 $\{\vec{r}_i^{(n+1)}\}$,
- (4) 计算在第 $n+1$ 步时所有粒子的速度: $\{\vec{v}_i^{(n+1)} = \vec{v}_i^{(n)} + h(\vec{F}_i^{(n+1)} + \vec{F}_i^{(n)}) / 2m\}$,

动能和速度标度因子: $E_k = \frac{1}{2} \sum_i m (v_i^{(n+1)})^2$, $\beta = \left[\frac{(3N-4)kT}{\sum_i m (v_i^{(n+1)})^2} \right]^{1/2}$,

(5) 计算将速度 $\{\vec{v}_i^{n+1}\}$ 乘以标度因子 β 的值, 并让该值作为下一次计算时, 第 $n+1$ 步粒子的速度:

$$\{\vec{v}_i^{n+1}\beta\} \rightarrow \{\vec{v}_i^{n+1}\}.$$

(6) 返回到步骤 (3), 开始第 $n+2$ 步的模拟计算。

按照上面的步骤, 对时间进行一步步的循环。待系统达到平衡后, 则退出循环。这就是正则系综的 MD 模拟过程。

例子: 下面我们举一个微正则系综的分子动力学模拟的应用示例来看看模拟的结果。

对一个总能量确定的单原子 (氩) 粒子系统的分子动力学模拟计算。

我们具体选取 256 个原子的模拟。粒子间的相互作用位势为 Lennard-Jones 势:

$$V(r) = 4\varepsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right]. \quad (6.4.12)$$

其中 $-\varepsilon$ 为位势的极小值 (取 ε 为能量单位), 其位置在 $r = 2^{1/6}\sigma$ 处。该体系的粒子限制在一个立方

体的箱子，边界上采用最小像力约定。我们采用自然单位制，长度和时间的标度单位分别为 σ 和 $(m\sigma^2/48\varepsilon)^{1/2}$ （对氩原子该时间单位为 3×10^{-12} 秒），这样就使得运动方程为无量纲形式。

模拟时我们考虑两个相图上的点： $(T^*, \rho^*) = (2.53, 0.636)$ ， $(0.722, 0.83134)$ ，它们分别具有两种立方体的尺寸，即 $L = 7.83$ 和 $L = 6.75$ 。初始条件假定为：各个原子处于一个面心立方格子的格点上，而速度按相应温度下的波尔兹曼分布抽样取值。位势的截断取两个值 $r_c = 2.5$ 和 $r_c = 3.6$ ，用以比较其对模拟结果的影响。在执行平衡化过程中，调节粒子速度的标度因子为

$$\beta = \left[\frac{T^*(N-1)}{16 \sum_i v_i^2} \right]^{1/2}. \quad (6.4.13)$$

反复上面的速度调节，直到系统能量达到给定值。在这个例子中，平衡化过程用了 1000 步 MD 模拟。模拟结果列于表 6.4.1 中。表中的误差为标准误差。系统总动能的模拟演化过程由图 6.4.1 给出。实际上，图中显示出在数百步后动能就达到平衡了。图 6.4.2 则显示出位能的平衡化过程。系统总能量

的平衡化过程则由图 6.4.3 表示，其平衡化是通过对粒子速度的调节跳跃式地达到的。图 6.4.4 为动能的分布图，模拟得到的平均速度为 $\bar{v} = 0.3654$ ，而理论上该值应当是 $\bar{v} = 1.13\sqrt{T^*/24} = 0.3668$ 。这个结果已经是相当不错了，因为我们只对 256 个粒子的系统进行了模拟。而且速度大于平均速度的粒子数所占百分比与期望值 46.7% 也一致。表中的数据表明模拟结果与所选择的截断距离值变化并不灵敏。

表 6.4.1 对 256 个粒子的氩原子系统进行 1000 步微正则系综 MD 模拟的结果

趋衡到 $T^* = 2.53$ ， $\rho^* = 0.636$

r_c	E_k^*	U^*	E
2.5	966.58 ± 22.1	-864.78 ± 22.4	101.79
3.6	972.15 ± 22.6	-920.10 ± 22.9	52.05

r_c	T^*	\bar{v}	\bar{v} 以上 %
2.5	2.53 ± 0.06	0.3654 ± 0.007	46.33
3.6	2.54 ± 0.06	0.3667 ± 0.007	46.71

趋衡到 $T^* = 0.722$, $\rho^* = 0.83134$

r_c	E_k^*	U^*	E
2.5	279.13 ± 9.57	-1421.98 ± 20.15	-1142.92
3.6	275.11 ± 9.72	-1496.45 ± 21.61	-1221.38

r_c	T^*	\bar{v}	\bar{v} 以上百分比
2.5	0.7297 ± 0.025	0.1965 ± 0.003	47.08
3.6	0.7192 ± 0.025	0.1949 ± 0.003	46.42

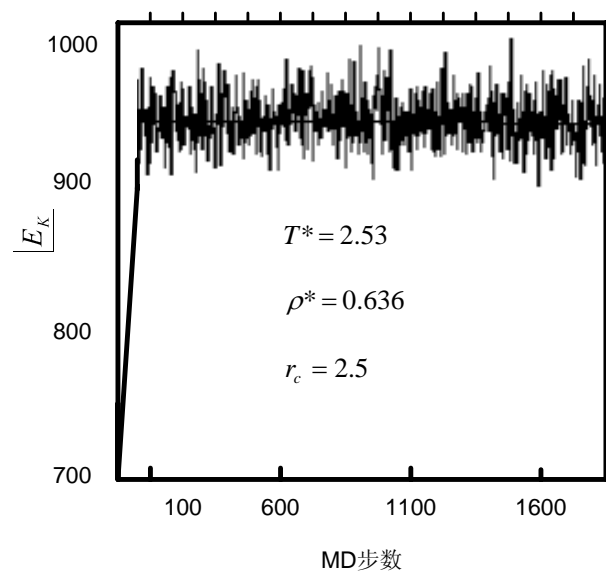


图 6.4.1 动能演化过程图 ($T^* = 2.53$)

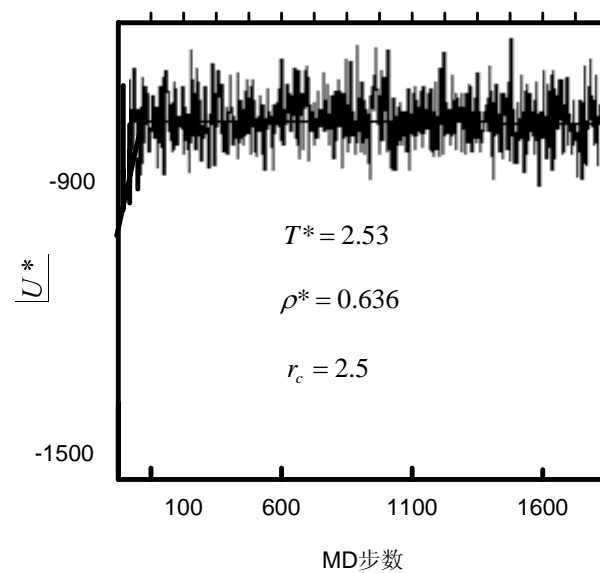


图 6.4.2 位能演化过程图 ($T^* = 2.53$)

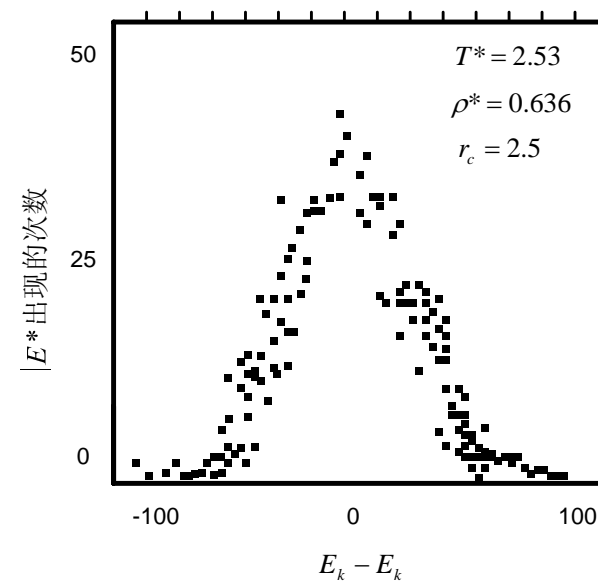
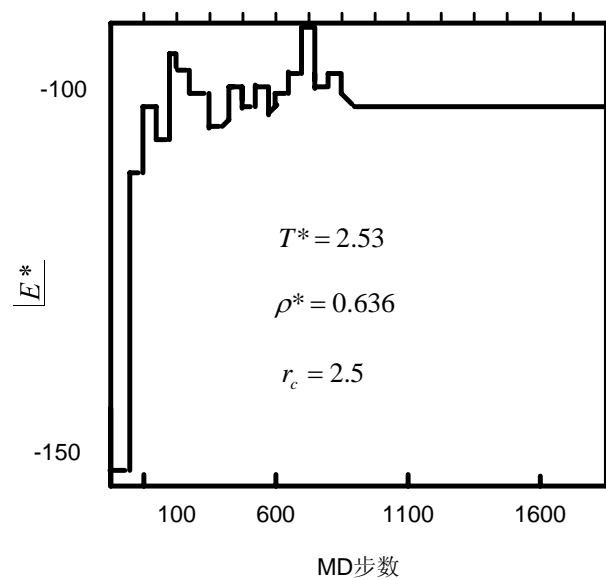


图 6.4.3 总能量演化过程图 ($T^* = 2.53$) 图 6.4.4 动能的分布图 ($T^* = 2.53$)