Monte Carlo方法：采用随机数于各种物理计算和模拟实验，以类似于赌博中投骰子的方式来随机决定其中某个单独事件的结果。

Lehmer线性同余法：( 乘同余法 混合同余法)

Schrage方法：m=aq+r, q=[m/a],r=m mod a。如2147483647=16807\*127773+2836.（q=127773,r=2836）

az mod m={z/q\*(aq+r)-rz/q} mod m={[z/q](aq+r)+(z mod q)/q\*(aq+r)-r[z/q]-r(z mod q)/q}mod m={a(z mod q)-r[z/q]}mod m=

也可以当4字节整数超过2^31-1时只保留后32位。如果是负数就加再加1。该方法应取奇数种子值使得整数值恒为奇数。

Tausworthe 位移计数器:。R250: p=250, q=103

避开随机数列缺陷的方法：对各个物理量不要连续地使用连续的随机数列，各自采用分别的随机数列并有各自的种子值，还应使循环的周期错开，或者淘汰序列中任意个随机数。

Fibonacci延迟产生器：。优点: 周期长

带载减法产生器. 的取值:

带载减法Weyl产生器. 的取值:

Marsaglia 产生器：组合产生器，用两个不同的随机数产生器序列生成另外一个随机数序列

独立性检验：自相关函数(线性关系)

均匀性检验：[0,1]分为K个子区间，每个区间内的应趋于。令统计量

很大：表示远远偏离理想值，趋于0：有可能N已经进入循环。通常使得求和中每一项大小约为1，此时。的极限分布是卡方分布，自由度。这是因为。

多维频率检验：设每S个随机数作为S空间中的一个点的坐标值，于是可以构成一个点序列。把S维空间中的单位立方分割成K个子立方体。是方体边长。理论频数。

直接抽样法：Poisson分布抽样： 指数分布: ；抽样:散射方位角余弦分布:

变换抽样法： 。设为上的均匀分布。寻找,使得，对进行均匀抽样，代入变换式得和的抽样。

Box-Muller法：， x和y为正态分布

球面上的均匀分布： ，大于1则重新抽样。, 将该方法用到上面的gauss中代替三角函数运算:

三维球面Marsaglia方法：，，大于1则重新抽样。。

四维超球面Marsaglia方法：; ，大于1则重新抽样。，大于1则重新抽样。

舍选抽样法：设。1. 由产生一对抽样值：，。2. 判断是否成立，成立则取为抽样。

简单分布：假设函数在有限区域[a,b]上有上界M，取h(x)=p(x)，取g(x,y)=1/[M(b-a)]。1.对g抽样， 2.判断3.成立，, y=epsilon\*M。判断y<p(x)是否成立。

比较函数：设F(x)形状和p类似但处处比p大，则先抽取出满足F(x)分布的满足，取。如果，则选取。

乘分布：p(x)=h(x)q(x), q归一可积。h(x)上界是M。取g(x,y)=q(x)/M, 0<y<M ; 0 otherwise。抽取服从q(x)分布， 。成立。取。取

定积分：

方差 协方差

相关系数

若和无关联:

中心极限定理：

大数定理：

提取法：与形状相似且积分值已知（较为平坦）

重要抽样法：与形状相似。按照几率分布选取。

权重MC积分：设，使得

Brown运动: 阻力: 涨落力:

Brown运动方程: m + 得到Virial定理(对颗粒总数做平均): 代入、得到 ；

扩散系数

一维RW：几率p向左，几率q向右，一步长。；

Smoluchowski理论: 如果粒子向右移动比向左移动多 m次,。n次移动中右移多于左移 m次的几率 令 得到

Einstein理论: Brown粒子在特征时刻内在方向位移的几率密度：

设单位体积内的粒子数密度，则时刻在内的粒子数为 按时间和空间分别进行展开得解得

Einstein关系:

由方差的推导:有N步，共有n条轨迹总距离

第i步，第j条轨迹:

单独一步对所有轨迹的平均:

走 N步后对所有轨迹的平均:

涨落:

自相关函数: 同时间下的相关函数为有限大小。

自发涨落回归:

液体分子速度的自相关函数:

第一涨落耗散定理(Green-Kuno公式):

Langevin理论,Brownian动力学:粘滞阻力:(是迁移率),快速涨落力:(系统特征时间碰撞事件)

Langevin方程(随机微分方程):

长时间平均: 是宏观唯象系数: 系综平均:

解出

Einstein关系:

与初始速度无关()趋于Maxwell速度分布

第二涨落耗散定理: 阻力摩擦力等与涨落有关。摩擦越小，涨落力越大。

计算得到

标度指数：

扩散方程: 。是粒子在时刻存在于 至之间的概率。两边乘与积分，,扩散方程的解是高斯正态分布，

熵：(平面上划分网格，为第格子里面的粒子数除以总数)

雨滴模型：。反射壁模型：无穷大势垒，到达时，下一步反射到。持续性RW: 某一步行进几率与之前的一步有关，同向: ，反向:

自规避随机行走(SAW)的标度指数计算

比值法：

权重法：1、起始一步权重为w(1)=1; 2、自相交时w(N)=0，重新开始另外一条行走; 3、如果可以选择3个方向：w(N)=w(N-1);4、如果可以选择的方向数w(N)=(m/3)\*w(N-1); 5、最后对行走数以权重加和求方均根值

标度律对各种维数都成立，可以假设三维随机行走时概率密度函数为 径向分布函数 峰值:

完全无序的气体：RDF为抛物线型

非晶态固体：r较小时有小峰，r很大时趋于抛物线

Flory-Fisher 平均场理论: 将链段的空间看做是按照光滑的Gauss型概率分布在质心周围，然后利用配分函数求得最佳构型。

设N个链段以恒定的密度分布在半径为R的球体内。总排斥能: 。配分函数: 自由能取最小值时对应最可几位形当时，右边第二项对于大N可以忽略: 当时，；。边缘维数: d=4

Eden 生长模型：1、在任何一种网格上随机选取一点，2、在已形成的占据集团的周界上任取一点

癌细胞模型：占据点表示癌细胞，未被占据点代表正常细胞。在三角网格上。1.将网格中心点设成是占据点即癌细胞，周围有6个最近邻格点，2.从周界上已占据的格点或周边的未被占据的格点中以几率比 k :1随机选取一点 A，再随机选取它的一个最近邻格点 B。如果 A 和 B 点有不同的标识（即有一个是癌细胞）的话，则将 B 的标识改为 A 的标识。 k 因子可以看成是癌细胞分裂率与正常细胞分裂率之比， k 大于0.5时就可以生长，当 k →∞极限时即为上面的 Eden 模型。

弹道聚集模型：粒子是从各处按随机选择的方向以直线飞行轨迹撞在聚集的集团上，粒子粘结在与集团第一次接触的位置上。这个模型生长出的图形是有支叉形结构的。

扩散受限聚集模(DLA): 1.取一个2维的方形点阵，在点阵中央原点处放置一个粒子作为生长的种子2. 从距原点足够远的圆周界处释放一个粒子，让它作 Brown 运动或随机行走3.该粒子走到种子的最近邻位置与种子相碰，这时让粒子粘结到种子上不再运动;粒子走到大于起始圆的更远处（如2-3倍的半径处）或干脆走到点阵边界，取消该粒子，把它重新放回原点。改进:让起始圆比聚集集团约为大一些即可，省略去从远处走进这个圆周之前所需化的时间。屏蔽效应: 越是尖端处生长得越快，从而形成枝蔓向外延伸，越是平坦处生长得越慢，从而出现沟槽中的空隙疏松结构。

Laplace生长: 边界条件: 在足够远处的外边界，势为常数。中央电极处的势为0。

RW求解 Poisson 方程:

从体系内第格点即0点开始游走，选中邻近4个格点之一的m点时，。直到碰到边界

levy随机行走或飞行，步长s按幂次律分布。。行走：粘在路径中所遇见的聚集集团的第一个粒子上。飞行：只粘在在航程终点处

偏压效应: 粘接几率正比于位置径向距离的指数律

介电击穿模型: 假设已占据格点上，远处为0。数值求解。假设格点扩散的速率为。环绕已占据格点周界上的一个空格子被占据的几率是

逾渗模型：二级相变

键逾渗：阀门在管子中间 座逾渗：阀门在接头处 座逾渗的值比键逾渗大

集团大小分布：。随机选取一个格点属于大小为s的集团的概率(排除无限大) 集团平均大小

得到

跨越长度: 集团的跨越直径或跨越长度，跨越长度: 集团中的两个座(对键逾渗则为两条键的中心)的最大间距。或者从集团重心计算出的平均距离或方均根距离。回转半径平方 , 为是从质心开始测量的集团半径的方均根距离。

当占据概率为时，点阵上任意一点属于无限大集团的概率定义为逾渗概率(序参量)

得到时，

对关联函数: 是离占据位点距离为 r 处的格点属于同一有限集团的概率。

集团的标识：用二维数组A(I,j)来标识格点所处的i行和j列,取值是集团序号。从左下角(1,1)处开始向右填充，如果随机数选取使得某一个格点为占据点时,则对应数组元素取值为1。对于后续占据的格子,如果它下方和左方均无占据格子的话标识+1。如果下方有占据格子，则等同于下方。如果下方无左方有,则等同于左方。取一标识集团所属的数组B，数组下标k就是上面标出来的集团序号，数组元素取值则为按上述规则该集团所应取的正确标识值。

逾渗概率, ；电导率, 跨越长度集团; 集团平均大小。跨越长度集团和集团平均大小均考虑。

对于有限尺度的体系，我们可以这样子考虑，对于时，是成立; 当时，。反解: 得到，再根据标度律求指数。

不同的点阵和逾渗类型不一样，但是标度指数一样，只与维数有关。

齐次函数就是幂指数函数 标度律： 逾渗边缘维数: 6

重整化：

Monte Carlo 重整化方法：一般来说集团上下连接的概率可以表示成

是由 n 个随机占据格子形成的上下端连接的几率。

设在空点阵上随机地添加上一个粒子使某格点成为占据态，如果不存在上下端连接路径，则在剩余的空格子中随机选择地再添加上一个粒子，产生一个新的构型。当有 m 个粒子添加并出现上下端连接路径之时起，对于的粒子，每添加一个粒子时计数一次，，直至构型数目足够多甚至穷尽所有可能的构型，然后以总的构型数目将 进行归一化。为了提高效率，可以在添加粒子的总数达到之后才开始检查是否出现上下端连接路径。

座-键逾渗的重整化群: 既有阀门在通道处，也有阀门在接口处。

刘维尔定理： ，平衡态时，，

满足的一种可能是，即系综体系在任意时候都是均匀分布的，系综成员等几率地分布在所有微观态中，对应微正则系综； 另外一种可能是，对应正则系综

正则系综： F(N,V,T)，恒温热浴; 微正则系综：S(N,V,E)，N 个粒子放入体积为 V 的盒子内，固定总能量 E, 无能量与粒子交换; 巨正则系综： J(,V,T)，有粒子源的恒温热浴； 等温等压系综： G(N,P,T);

正则系综：

随能量剧烈增加， 但玻尔兹曼分布急剧减小，所以两函数的乘积会在某个 值附近有尖锐的分布。所以正则系综和能量严格为的体系几乎等价，正则系综中能量的涨落不会很大。当粒子数和体积趋于无穷的时候，两系统的平均值相等。

正则系综的特征函数是Helmholtz自由能 F(到达平衡态时F取极小值

总能量

我们称一个序列是 Markov 的，如果某一时刻 x 取值的条件几率是独立于上一时刻之前的所有 x 值的话。

定义，则

设状态数目为 M，转移概率构成的矩阵， ,Markov 的极限和初态无关。平衡态: Boltzmann分布

任务就是寻找合适的转移概率矩阵 M，使其极限分布为 P。

主方程(相当于几率守恒方程, 即对所有时间都有)。平稳分布, 细致平衡解, 可逆 markov 链(即由某步到达 并在下一步到的概率和反向的概率相等): 有 2M 个方程，但是矩阵元有，所以细致平衡下的主方程式不能唯一地确定跃迁矩阵。

Markov 过程不能应用于完全决定论过程，但是更多情况下我们只关心体系的粗粒平均，在不同的时间尺度下有可能将刘维尔方程改写为主方程。

Metropolis 方法：设， 是由选择步进到的概率，矩阵对称。代表接受的概率非对称，根据待满足的几率分布 p 而定。转移概率表示式为：

i和j相同：

Barker抽样规则：矩阵对称。代表接受的概率非对称，

Metropolis-Hasting抽样规则：更一般地，建议分布和接受几率都是非对称的，接受几率根据待满足的几率分布 p 形状而定：

设为所考虑的几率密度分布(正则系综:)，并且已经产生了个抽样点，产生下一个抽样点。可以在上一个点附近构造一个试探解，，是试探步长（可正可负，例如可取，是固定步长， 是均匀分布的随机数），该点是否被选取决定于比值：

1、 如果则选取，即。2、 否则，产生区间内均匀分布的随机数 ，如果则选取，否则，放弃；即。

热化处理试探步长或 的选取对抽样效率和结果分布也很重要，通常的选取是使得接受的效率为一半左右

正则系综：将玻尔兹曼分布直接代入 Metropolis 抽样即可。

Ising 模型：采用格点模型，自旋设为，自旋有相互作用，系统的Hamilton量为代表邻近的自旋对， J 是交换积分常数。如果一对自旋方向相同，能量为-J，方向相反，能量为 J。因此 J 大于 0 有利于使得所有自旋方向排成一致使得能量最低。这就是铁磁性，如果 J 大于 0，那么自旋对取向相反的时候才可能使得能量最低，宏观不表现磁性。但是加上外磁场之后逼迫自旋取向相同，产生磁化，这就是反铁磁性。温度升高时， 热激发使得某些自旋随机反转，这就是顺磁性。

自由能

一维 Ising 解：。零磁场下，配分函数。每个自旋的自由能为 ，是温度的解析函数，没有相变点。有外磁场时，若采用周期性边界条件， 配分函数； 。，，是温度的解析函数，没有相变。去掉外磁场，，除了点无自发磁化，没有有序相变。

二维 Onsager 解：比热

第一类椭圆积分第二类椭圆积分

在处发散，相变点：

三角格子:

二维 Ising 模拟：减弱有限边界的影响，可以假设周期性边界条件。指数的确定同样可以使用有限尺度标度法。总能图上是拐点。

关联长度𝜉在相变点处，意味着每一个自旋都对其他自旋态特别敏感，因此涨落特别大，稍加一个外磁场就可以极大地改变体系的磁化强度。

在无外磁场时， M 的变化是连续的，为二级相变，在有限磁场下是一级相变。只是在磁场转变为正值即沿 +z 轴方向时，磁化强度突然转向。因此，磁化强度是不连续变化的，为一级相变。注意到在 时， M 和 −M 两个状态是等几率的，但稍加一点外磁场就可使状态取完全不同的几率。在之下，都有这种随磁场的不连续变化。但在Tc之上，由于自发磁化消失，因此当磁场在零值附近变化时磁化强度没有不连续变化行为。 M 由 H < 0 的负磁化方向变为 H = 0 时的零值，再连续变为H > 0 时的正值方向。Tc之下的磁化强度的跳跃是自发磁化强度的两倍，而且在Tc处自发磁化强度消失，因此随磁场变化的一级相变与随温度变化的二级相变之间有紧密的关系。低温下系统有两个相，即 ±M 。在温度轴的Tc点之下，可以通过改变磁场进而不连续地由一个相转变为另一个相。反映在相图上，有一条一级相变线位于温度轴上,终止于临界点。在临界温度处，没有自发磁化，因此两相的区别消失。等于和高于Tc处,可以跨过温度轴而连续地改变 M 。一级相变终止于临界点是一级相变的普遍特征。

XY模型：系统的自旋具有两个任意的取向。，

其中是自旋矢量和轴的夹角。正方格子上的两维模型，对于所有T > 0 的温度，磁化强度 M = 0 ，但在某一温度下将发生 Kosterlitz-Touless 相变，它是一种在无长程有序系统中发现的一种拓扑有序。系统可以存在一种亚稳态，自旋的排列形成涡旋。在相变温度以上，涡旋是自由的。在相变温度之下，自旋涡旋是成对出现的，并且对于的所有温度系统都和 时一样，因此临界点实际上是临界线。计算模拟时，系统的初始构型先选取对应于高温（ ）下的完全无序排列， 即格点上的每个自旋方向是随机选取的，。然后急速降温至 ，即改变 Boltzmann 分布中的指数值时步长间距要大，在每个温度值下按照上述的 Metropolis 抽样法产生系统的各种构型。 由于急速冷却，系统可以形成长寿的亚稳态涡旋结构。相变温度下正涡旋和反涡旋是同时出现的。

Hesienberg 模型：自旋可以取三维空间中的任意方向。注意无论是 X,Y 模型还是Heisenberg模型都可以用 在一维，二维，三维格点上。

在系统最低能量的状态即基态上，所有自旋按照完全有序的平行或反平行排列，在有限的温度下由于热激发出现能量较高的状态。在 Ising 模型中这样的激发态是自旋的反转，在 Heisenberg 模型中可以出现周期性的自旋波激发，它是系统的一种集体行为，其量子称为磁子，由自旋波的 Bose 统计可以推导出磁化强度与温度成的关系。而在一维 XY 模型中可以出现孤立子或孤立波激发，一维链上的自旋发生 的扭转，在反铁磁情况下这种扭转有3种：一种自旋子格子上的自旋扭转，而另一种扭转；两种子格子的自旋各自扭转；一种子格子上的自旋不变，而另一种扭转。对于三维 Heisenberg 模型，其中的一个自旋分量在外加磁场下呈现有序，而另外两个分量出现类似于 Kosterlitz-Touless 相变的束缚拓扑态激发。 随着温度升高， 涡度增加但正负涡旋束缚态开始解离。

q 态 potts 模型：的取值可为 q 个， Hamilton量=是，其中每个格点上的取值为1, 2, …, q 。

时钟模型：自旋是个矢量，它的空间取向限制在二维平面中 q 个离散指定的方向值的话系统的 Hamilton 量是

q > 4 的情形，是 XY 模型退化成 q 重各向异性的极端情形，这时系统有两个Kosterlitz-Touless 相变，其计算结果的分析十分复杂。

自旋玻璃： Ising 模型中，交换常数 J 不再是常数，可以从高斯分布中选取，或者赋予正负值不同几率。磁化强度不再是一个好的序参数，因此用 Edwards-Anderson 参数作为序参数，其中的是指在某单个初始键分布下的各格点的热平均，则是指对不同初始键分布的平均。

模拟退火法：同时模拟不同温度下的系统，在模拟步骤中途交换这些系统的温度，因此可以将低温下冻结的状态转移到高温，以脱离亚稳态。

自旋自相关函数: :特征相关时间(与温度有关)

临界慢化：在临界点附近，由于长程关联和无穷大涨落，采用Metropolis算法的单自旋翻转难以实现大集团涨落下的集体自旋翻转，尤其是集团中心自旋翻转概率，导致达到进入平衡态的计算时间显著延长。

Wolff算法（自旋块翻转）: 1.随机选择一个自旋为自旋块生长的种子，检查它的某近邻自旋是否取向相同；2.如相同，以几率添加到该自旋块中；3.对新添加的块成员自旋，再检查其近邻自旋以确定是否要添加到块中；4.添加块新成员时，以前未成功添加入块的自旋可再被赋予添加新机会；4.全部添加完成后，尝试将该块的自旋集体翻转（翻转几率取决于能量消耗）;

依赖于温度：随温度增加而降低。当选择 ，两种翻转都可全部接受，满足细致平衡条件。

RKKY相互作用势能： 长程震荡

Edwards-Anderson模型: 或 高温:顺磁相 低温:自旋玻璃相

序参量:

Sherrington-Kirkpatrick 模型: 或

副本（replica）方法:

模拟回火法（Simulated Tempering）: 同时模拟两个一样的系统：其中一个是在高温下，不会产生遍历性破缺；另一个在低温下，可能处于亚温态的能量势阱中。先用Mteropolis单自旋翻转方法模拟两个系统各自的演化，再以一定速率。

交换温度的接受概率:

低温系统能量较高

遍历性：高温系统的态直到温度交换时都可遍历，温度交换后另外一个系统也可遍历，两个系统是对称的，因此都具有遍历性。

细致平衡条件：各自系统的Metropolis步进显然是满足的，交换温度时也是满足的。

KAM定理: 一个充分接近可积Halmilton系统的不可积系统，对此系统若把不可积当作可积Halmilton函数的扰动来处理，则在小扰动条件下，系统运动图像与可积系统基本一致，当扰动足够大时，系统图像就发生了性质改变，成了混沌系统。通俗地说就是，经典力学的相空间轨迹既不是完全规则，也不是完全无规，而是十分敏感地依赖于对起始条件的选择。微小的涨落可能引起混沌的发展。揭示了决定论和随机论之间、牛顿力学和统计力学之间没有不可逾越的界限。

Li-Yorke定理: 对闭区间 I 上的连续函数 f(x)，如果存在一个周期为 3 的周期点时，就一定存在任何正整数的周期点，一定出现混沌现象。

一维迭代Logistic方程:

Feigenbaum常数: 前后分岔间距的比值。趋向于一个常数。

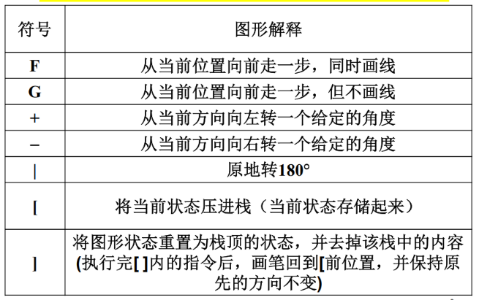
倍周期分岔中的标度行为按以下的几何级数（幂函数）收敛到: ，是依赖于迭代函数的常数，是不依赖于迭代函数的普适常数。倍周期分岔的图上作的直线和放大的分岔曲线相交，得到分岔纵向间距等值。 ；是普适常数。

混沌的特征: 内随机性、分维性质、普适性和Feigenbaum常数

二维迭代Hénon方程:

吸引子: 轨迹在充分长时间之后将渐近地收敛到的极限的状态或状态的集合。简单吸引子的维度是整数,吸引子本身不可分解，吸引子在相空间的体积为 0。

Lorenz方程:

奇异吸引子具有吸引子的性质， 奇异性: 1.对初始条件具有初值敏感性2.具有无限嵌套的自相似几何结构，即具有分数维3.几何结构完全不随迭代方程的参数的变化而连续地变化，运动轨迹永远不自我重复，即具有非周期性。

Rössler吸引子:

物理系统中的吸引子：双摆、蔡氏电路、杜芬振子、Gumowski-Mira model

落叶效应: 当雷诺数达到1000时, 树叶下坠形成混沌运动, 并出现了卡尔曼涡街

Lyapunov(李雅普诺夫)指数: 用来表示初值敏感性是否出现以及敏感的程度。经过迭代，原先很接近的状态之间的差可能变得很大。。Lyapunov指数衡量系统动力学特性的一个重要定量指标,表征了系统在相空间中相邻轨道间收敛或发散的平均指数率,是对混沌态的初值敏感性的定量判据。

正的Lyapunov指数可作为混沌行为的判据。系统相空间中运动轨道在每个局部都不稳定、混沌现象、混沌吸引子。Lyapunov指数小于0 对应于不动点和各个周期的运动状态。轨道在局部是稳定的；对初始条件不敏感。Lyapunov指数等于零：对应稳定边界，初始误差不放大也不缩小。指数由负变正：表明周期轨道向混沌的转变。

Julia 集: （: 和是吸引子，单位圆周上的点组成Julia集）多次迭代后的点 不趋于某固定点，也不趋于无穷值时，则相应的属于该C值下的Julia集。

实际模拟:距离函数：； 每次迭代都计算距离D, 如某初始点，经过次迭代(为设置的迭代上限)后仍有 (为设定的逃离的边界)，表示点未逃离，就可以认为此属于Julia 集，在平面上画出点；若还未达到 次迭代，已经有，则点不属于该Julia集，点不画出。彩色 Julia 集: 可对每一个，计算逃离 所需迭代的次数；按逃离所需迭代的次数的不同，对起始迭代点进行分类，分别给 不同的灰度或颜色，就可以得到灰度或彩色图案。

Manderlbrot 集: 给定 Z0 后，在复参数 C填充的平面上，对参数 C 进行分类得到的图形。通常取进行迭代；参数变化，不变。对每个，以开始，连续计算：如果在时, ，则该 点在 M 集以外, 否则认为 C 点在 M 集内对每个 C 值，计算该 C 值下所需的迭代次数，根据此迭代次数区分颜色。

M集的特性: 自相似性、M 集概括了所有的 J集，是 J 集的缩微字典

广义J 集，广义M 集:

拓扑维数: 欧几里得空间的维数与拓扑维数相等

Hausdorff维数: 是测量单元的尺寸, 是测度得到的规则图形的测量单元数。

Cantor 集:

Cantor 集性质: 自相似性、无穷操作或迭代过程、精细结构、传统几何陷入危机、长度为0、简单与复杂的统一

分形维数均大于拓扑维数: Koch 曲线: ; Levy 曲线: ; Sierpinski三角毯: ; Sierpinski正方毯: ; Vicsek 图形: ; Sierpinski-Menger 海绵:

分形的特点: 图形是“支离破碎的”，从数学上看它处处是奇点，如处处不连续或处处不可微; 分形具有标度不变性，即改变尺度或标度时，图形是相同的或相似的; 分形的豪斯道夫维数一般是分数(不排斥是整数)，并且大于拓扑维数.

布朗运动: 。: 位移的方均值 : 无量纲约化总位移 :图形包含的单元数，

相当于图形尺寸放大的倍数。任何维空间中随机行走的分维都是 2。

自回避随机行走:

粗糙曲线的圆规维数： 用半径尺寸为 1 的圆规从上端开始作圆弧和海岸线相交，其交点为下一个圆弧的中心，这样得到海岸线的总长度为 N(用长度为 1 的尺去丈量，得到 N)，减小尺寸为 s 后丈量，得到更大的 N(s)。如果作 lnN~lns :图后得到斜率为负的直线，这表明存在如下的幂函数关系:N~s^(-D)。 这样测定的分维被称为圆规维数。

周长-面积关系求分维(小岛法)：规则图形的周长P与测量单位尺寸𝜀的一次方成正比，面积 A与𝜀的二次方成正比。。二维不规则分形图形。周长光滑时 ，和形状有关，是测量尺寸。

表面积-体积关系求分维:

盒计数法: 将尺寸分别为 s =1/4,1/8 的网格覆盖在分形图形上，计数网格有像素的方格数目，例如得到 N(1/4)=16 和 N(1/8)=60。为了减少误差，应该使不同尺寸的网格能覆盖相同大小的图形，如 512×512 象素的图形的应当是1, 1/2, 1/4, …直到降到1/512。作 图,

Sandbox 法： 把一个分形生长到不同阶段得到的图形进行分维计算，是将一系列尺寸(r >1)不断增大的方框（也可以是圆）覆盖到分形图形上，计数不同方框(或圆)中象素数: 。适合用于单个生长图形。对单中心图形， Sandbox方法更实用。

面积-回转半径法：把所有分形看成大大小小先后生成的图形

已知分形中心或质心时: 无明确中心或质心时: 对包含许多团簇的图形，用面积-回转半径法能得到更准确一些的分维(比从中选出几个典型的分形计算的分维值更准确)

变换法：(1)固定矩形宽度，矩形高度由最高和最低点的高度差决定，一步一步移动矩形,遍及所有象素点(矩形高度有变化)，将每一象素点处矩形的面积相加，得到总面积(2)改变矩形宽度的大小, 重复以上操作；(3)由求: 覆盖一部分粗糙曲线所需的面积为的盒子数，不过它一般不是整数。愈大，越小；(4)做双对数图

变换法计算粗糙曲面的分维计算。(1)此时测量用的矩形被正方柱代替，正方柱的底面取为。正方柱的高度由正方柱范围内粗糙曲面的最高、低点高度差决定， 一步一步移动正方柱遍及所有点，将所有正方柱的体积加起来得到总体积。(2)改变的大小后重复以上操作，得到一系列体积。(3)计算, 实际上是覆盖一部分粗糙曲面所需的体积为的盒子数(4)作曲线，求线性部分斜率D。

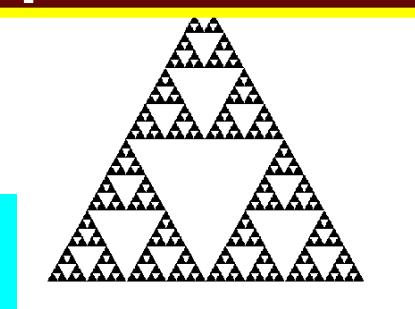
密度-密度相关函数法:

是图形的密度函数，有图形象素处为1，无图形处为 0；N 是总象素数；C(r)的几何意义是: 原始图形和平移 r 后的图形重叠部分的象素数和全部象素数的比值，即在相距 r处发现另一象素的概率; 对于有限的图形，C(r) 随 r 的增大而减小。除了对 N 个不同的 求平均之外，还对不同方向、长度相同的 r 求平均

特例，固定, 并把它取为图形的中心, 即, 此时。表示重心为圆心、为半径的圆周各点上发现另一图形象素的概率。由于分形具有自相似性，可以将表示为幂函数。在回转半径内积分，在足够大时，积分值很接近于和图形总象素数成正比，即。D是分形维数, d 是欧氏空间维数。

标度不变性：标度(测量尺度)改变了倍后，函数具有自相似性或标度不变性

L系统： L 系统由三部分组成。: 模拟事物的最基本结构或初始结构。 : 公理, 迭代过程中遵循的一些基本规则。: 生成规则(或迭代规则)

实际操作: 1、用各种字符串符号表示 L 系统的基本结构，公理和迭代规则。2、按一定的规则和公理，进行多次迭代并画图。迭代过程就是字符串重写(或替换)过程, 多次迭代后将产生一个较长的命令串, 然后可根据预先规定的各字符的图形学解释, 生成图形。作用一次称为一级，一般来说选择的级数不宜太高,通常选 2~8 级,最多 15级。

Sierpinski 三角毯:Sierpin{Angle 60 ;角度增量是 60°Axiom FXF-FF-FF ;初始图形是一个三角形， X 是替换中的中间变量，在作图形解释时跳过它，不做任何操作F=FF;替换规则 1 是将每一线段分为两段X=--FXF++FXF++FXF--﹔替换规则 2 是将中间变量 X 变成一个三角形};结束

迭代函数系统(IFS): 采用确定性算法与随机性算法相结合。“确定性”: 用以迭代的规则是确定性的，它们由一组仿射变换(如,,等等）构成；“随机性”: 每一次迭代用哪一个规则（即选Ri 中哪一个），不是预先定好的，而是随机的。每个迭代规则 Ri 都是一个仿射变换。设最终要生成的图形为 M，它要满足下述几何方程: 。极限图形M应是所有迭代的吸引子，收缩的仿射变换才能保证迭代收敛到 M 上。

设平面上有面积区域为，经仿射变换后变成，则与的面积()关系为:。

, 分别是旋转，扭曲，拉伸。

自仿射性是自相似性的一种拓展和延伸。自相似性可看成是局部到整体在各个方向上的等比例变换的结果。自仿射性可看作局部到整体在不同方向上进行不等比例变换的结果。自相似性变换是自仿射性变换的特例。

图形压缩率: 。若某个 ，则实际操作时可把该设成一个较小的整数，如0.00001。

元胞自动机是定义在一个由具有离散、有限状态的元胞组成的元胞空间上，并按照一定局部规则，在离散的时间维上演化的动力学系统。

Wolfram一维元胞自动机：一维情况，设演化规则的半径为 r ，则包括自身共有2r+1个近邻；每个位置可能有的状态为 k，则 2r+1 个邻座可以取种状态排列；对于每种排列，下一个时刻可以取 k 个不同的值，因此一共有个不同的元胞自动机。

，的 256 种元胞自动机的编码规则： 第一行是对相邻状态的编号(从 0 到 7 共 8 种)；第二行中每种相邻状态编号对应的三个二进位数的值；第三行的八个二进位数换算成十进位数 90（第四行）， 就得到这条规则的编码。

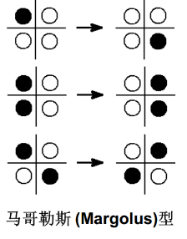
一维元胞自动机的长期行为: ①演化到全部是 0 或全部是 1 的均匀状态(0，4，16，32，36，48，54，60，62)；②演化到不随时间变化的定态或周期性的循环状态(8，24，40，56，58)；③演化到混沌状态，具有初值敏感性(2，6，10，12，14，18，22，26，28，30，34，38，42，44，46，50)；④演化到更复杂的结构，具有局部结构的复杂模式处于“秩序”与“混沌”之间，被称之为混沌的边缘 (20，52)。

边界条件: 1.周期型: 一维空间首尾相接。二维空间上下相接左右相接2.反射型: 边界外邻居的元胞状态是以边界为轴的镜面反射。3.定值型: 所有边界外元胞均取某一固定常量，如0, 1等。

这三种边界类型在实际应用中，可以相互结合。有时在应用中，为更加客观、自然地模拟实际现象，还有可能采用随机型，即在边界实时产生随机值。

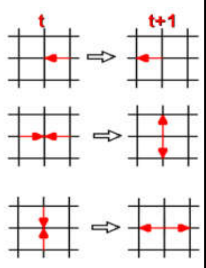
元胞自动机的主要特征: 空间是离散的、时间是离散的、状态取值是离散的、演化的运算规则是局域的。

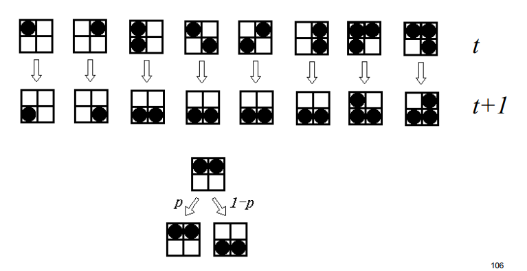
二维元胞自动机： 常用邻居定义: 冯-诺依曼(Von. Neumann)型: 周围四个；摩尔(Moore)型: 周围八个；扩展的摩尔(Moore)型: 比Moore型再大一圈；

表决与退火模型: 网格: 正方形网格; 初始状态: 网格上随机分布一些白点和黑点; 邻居：Moore 型邻居; 生长规则1: 多数决定规则，即 ≥5 ● 得 ●，≥5○得○; 模拟结果：形成了一些相当细碎的畴域；生长规则2：少数决定规则，即≥5个●得○，≥5个○得●;模拟结果: 黑白边界的棱角和曲折抹平了许多，黑白的畴界融合起来，成为更大的畴域，这很像是退火过程，产生了表面张力引起的效果。

Q2 规则—Ising 自旋动力学模型：网格：二维正方网格,每个格点有自旋 S=1 or 0;邻居：Von Neumann邻居;物理依据:自旋对的耦合能量—相同自旋排列能量较低(设为),相反自旋排列能量较高(设为);演化规则:要求保持局部能量守恒,即只有当自旋向上的邻居数和自旋向下的邻居数相同时，自旋才翻转，在时刻变成。因为自旋的这种变化没有引起任何能量交换。上述判定某格点自旋翻转与否是基于邻居不变化的假定，如果邻居也翻转(因为它们遵循相同的规则),那么能量将不守恒。解决这个问题的方法是:把网格划分为奇数子格和偶数子格(如同国际象棋盘上的黑白方格)；状态更新分为两步：首先根据偶数子格上自旋的构形，翻转奇数子格上的自旋；然后再根据奇数子格上自旋的构形，翻转偶数子格上的自旋。

格子气自动机(HPP模型)：目的：模拟流体粒子的运动;网格：二维正方网格;邻居：Margolus 型邻居;限定条件：指定运动方向进入指定格位的粒子数为 1。t 时刻格位信息 s(r,t) 中包括4位数，如 s(r,t) = (1011) 表示有3 个粒子分别沿 1, 3 和 4 方向进入该格位。可以按照能量与动量守恒来制定相应规则。粒子相互作用的特性：碰撞过程满足动量守恒和粒子数守恒定律;交互作用的微观性质：时间逆转过程中的不变性。HPP 模型采用正衬底，不能模拟各向同性。FHP模型:三角型网格，可模拟出各向同性现象。

沙堆规则： 目的：模拟像沙一样的颗粒的基本堆积和倒塌现象;邻居： Margolus型邻居(2x2的元胞块做统一处理 )

Langton 蚂蚁规则:目的:模拟蚂蚁运动;网格:正方网格,元胞为白色或黑色;初始条件:所有元胞都是白色,蚂蚁集中在图中间;规则:① 白色元胞中，蚂蚁向左转 90 度;② 黑色元胞中，蚂蚁向右转 90 度;③ 蚂蚁移动进入下一个元胞时，原来的元胞颜色反转(原来白色的变为黑色，原来黑色的变为白色);④ 每个元胞可以被不同数量的蚂蚁占据。多蚂蚁情况:格位的颜色依据蚂蚁存在的数目进行修改:原来白色的:有 1 个或 3 个蚂蚁变为黑色,有 2 个或 4 个蚂蚁保持白色;原来黑色的:有 1 个或 3 个蚂蚁变为白色,有 2 个或 4 个蚂蚁保持黑色。多蚂蚁行动可以观察到协同行为和干涉行为:一旦公路建成，其他蚂蚁可利用该公路快速行动;当第二个蚂蚁插手这个公路的建设时，将有破坏性的后果:如进入新的混沌运动，或公路沿其他方向迁移。

森林火灾模型：用二维正方网格来表示森林: 其中每一结点可以是一棵绿树(结点值为 1 ) 、一棵着火的树(结点值由 1 变为 0 ) 、或者是空地(结点值为 0 )。演化规则：下一个时刻 (1) 正在燃烧的树变成空格位；(2)如果绿树格位的最近邻居中有一个树在燃烧，则它变成正在燃烧的树；(3) 以概率 p 随机选取一个结点-如果它为空地-则长出一棵树; (4)没有最近邻为着火状态的树以概率 f 被点燃。整数型森林火灾模型：结点的值可以是大于 1 的整数，表示该结点树木的质量。m:火灾损失(火灾面积或火灾烧掉的树木总质量)N:为森林中质量为 m 的树丛的个数。N 与 m 成幂函数关系

管虫礁模型：模型中用●代表管虫的活动状态，○代表管虫的藏匿状态。设当一只管虫的邻域内活动的邻居多到 n 个，它所受到的刺激足以使它藏匿起来。模拟时用计时器计算藏匿起来的管虫何时该出来，并且采用平面 0 和平面 1 两块板，前者作为管虫的栖息地，后者是各处管虫的计时器。计时器所取数值决定于管虫的藏匿时间，以计时器取 0，1，2，3 四个值为例，对于活动的管虫计时器的状态为0，一旦它的活动邻居数达到 n 个，计时器跳到 3，发出警告。下一步管虫缩回管内，计时器退到 2，然后逐步倒数，经1至0。下一步管虫就重新钻出来了，计时器保持读数为 0，直到再次受到足够强烈的刺激为止。

Born-Oppenheimer近似: 由于(Mn>>me, vn<<ve)，电子的运动和原子核的运动可以相互分离。

分子动力学：只考虑核的运动(其动力学方程由牛顿方程近似) ，电子的贡献由原子间的相互作用势能来描述。

局域电子密度泛函理论：只考虑电子的动力学问题 。

MD 研究对象:大量粒子(N个)集合体系(固体、气体、液体) ；粒子的最小结构单元是原子而不是电子，也可以是分子或更大的粒子的集合。

MD 模拟目的:1.计算体系状态(由 N 个粒子的位置 ri、动量 pi 或速度 vi 标志) 随时间的演化;2.模拟与粒子运动路径相关的基本过程或粒子在相空间中的轨迹;3.进而计算感兴趣的物理量的值 Q;4.MD 是微观过程与宏观性质的桥梁。

MD 模拟方法: 分子动力学是在原子、分子水平上，用数值方法，求解经典牛顿运动方程(微分方程)，模拟体系状态相空间演化轨迹，是一种经典力学方法。

硬球的势能:

硬球碰撞遵循的基本规律(1)碰撞是弹性的;粒子间的能量转移是改变粒子的动能，而不是用于球体的形变或内部自由度；(2)总动能守恒;(3)总动量守恒；

两等质量、等直径硬球的一维运动: 一维运动下，等质量的粒子碰撞后只交换速度，因此粒子的初始速度分布不变；不管模拟时间多长也不能达到各态历经性；其计算得到的平衡态性质不对，除非大量粒子的初始速度是由Maxwell分布中得到。

两等质量硬球的二维或三维运动: 二维、三维或高维下，每次碰撞时的 r12方向(两球的连线方向)的速度分量可发生变化；多次碰撞后可以达到各态历经性；最终，粒子速度分布可取得合理的平衡分布。

周期性边界条件: 实际上只能对给定密度下的有限的粒子数目N 即有限面积或体积作计算; 为了减少有限面积/体积的边缘/表面效应，需要对体系的空间进行周期性扩展; 不仅要考虑第 i 个球和V中的第 j 个球之间的碰撞，还要计算它和V 周边扩展区域（镜像元胞）中的第j 个镜像球之间的碰撞; j 球和其镜像球的速度矢量是完全相同的，只是空间位置不同当 j 球和其镜像球 j’与 i球的碰撞都满足碰撞条件时，选碰撞时间最短的与镜像球 j’的碰撞等同于与 j 球的碰撞，碰撞后j 球也同样改变方向

硬球碰撞的模拟步骤: I、初始化(1) 给定粒子数 、密度或堆积分数；设模拟空间是边长为 的立方体，粒子直径为 (2) 热能 KT 作为能量单位，将时间和速度约化(3)给定初始位置坐标 {ri (0), i=1,…N}由于硬球是不能互相重叠，因此采用随机方法选取其空间坐标将是极为耗时的，除非密度非常小。通常按规则晶格排布，如对单原子分子的模拟，常以面心立方开始；也可用上次不同条件模拟得到的结果为初始位置(4) 给定初始速度{vi(0), i=0,1…N }速度值分布无关紧要，可以取 [−1, 1] 区间的均匀分布，或 Maxwell 分布中随机选取；但其平均速度应对应热运动能；要保证总动量为零。方法是将每个粒子的动量求和并除以粒子数，得到平均总动量，再将每个粒子的动量减去此平均总动量(5) 构造N(N−1) /2个球对的碰撞时间列表II、动力学模拟至平衡态(1)从碰撞时间表中找最小值，决定距离下一次碰撞的时间差、以及碰撞球对 (i ,j)；(2)将所有粒子的坐标前进一步：(3)将周期性边界条件运用于每一个在后离开原胞的粒子，求其在元胞中的空间位置；(4)计算碰撞粒子对的新速度矢量；(5)更新碰撞时间列表 [含N(N−1) /2个球对]；(6)监控位置和动量有序度参数,判断是否达到平衡(7)对上面(1)-(6)步骤执行循环，直到达到平衡。

平衡态判断: 在相空间中看体系趋于平衡的过程，就是使轨迹从任意给定的初始点，运动到最可能的区域－平衡态区域的过程；为了监控这个趋于平衡的过程，需要用两个参数，分别用于跟踪原子位置无序度的变化以及速度的Maxwell分布的演化；对于从晶格起始的模拟，可采用平移序参数作为位置参数；对于速度分布的演化，通常用一个数即Boltzmann H 函数来表征。

(1)位置有序度参数:Verlet平移序参数 (面心立方晶格)

(2)动量有序度参数: Boltzman H 函数

结果统计：相对压力或压缩率: 第二个等号代入了 第三个等号代入了

由此可以求出由于相互作用势的引入带来的非理想气体的压力；它是分子模拟中进行统计分析和待求的主要结果。硬球体系在高密度下呈现固态，有长程有序，自扩散系数很小。而在低密度下为流体特征(液态)，即格子会融解，体系无长程有序，扩散系数值有适当大小。在中间的相变区域，体系处于亚稳态，在固相和液相之间变动，亚稳态的寿命与尺度N 的大小有关。无吸引势相互作用的体系中的固化是由于不可形变的物体堆积在较小的空间造成的几何结果。该硬球体系不能描述气－液相变。

软球的势能、势能的截断。解牛顿运动方程: (1)力的计算(2) 列表技术 (用列表技术减少计算力Fi需要的时间) (a)元胞列表法: 计算量与 N 成正比，实际计算时并不需要考虑与其他 N-1 个粒子的相互作用，只需要考虑一定范围内的粒子体系空间分成若干尺度等于或稍大于 r c (势能的截断半径)的元胞；给定元胞中的一个粒子(红)只与相同元胞和近邻元胞中的粒子发生相互作用。(b)最小映像判据(c)Verlet列表法(d)Bekker法—周期性边界条件下的verlet列表法(3)位置和速度的计算

模拟步骤: (1) 初始化给出系统的状态参数:粒子数N 、密度、截断半径rc，温度T(正则系综)，元胞边长(L>2rc)。软球的初始位置{ri (0), i=1,…N}:固定格子坐标、或随机偏离固定格子、或完全随机分布(因软球可互相重叠)。软球的初始速度{vi (0), i=1,…N}:速度大小和分布无关紧要,可以是[−1, 1]区间的均匀分布或Maxwell分布中随机选取;保证总动量为零。构造verlet或元胞列表(2)动力学模拟至平衡态(1)根据列表技术,求出每个粒子受到的作用

力的 x, y, z 分量：Fx(t)、 Fy(t)、 Fz(t)；(2)增加时间步长，根据运动方程的Verlet算法或其它算法将所有粒子的坐标前进一步，并计算速度;(3)计算体系的瞬时温度、动能、势能和总能(4)计算监控体系是否趋于平衡的物理量, 判断体系在当前时刻是否达到平衡；(5)将周期性边界条件应用于每一个在后离开元胞的粒子空间；(6)更新列表(verlet列表或元胞列表)；(7)对上面(1)-(6)步骤执行循环，直到体系达到平衡态。(3) 结果统计

平衡态判据(1) 因总能量恒定，则动能和势能的涨落的关系(2) 每个速度分量都满足Boltzmann分布，各速度分量平方的平均相等(3)温度、压力等热力学量在一个稳定的平均值附

近涨落, 涨落大小为(4) 如果将体系划分为小等分体系，物理量在每个小体系下的时间平均应该相等。(5) 热力学量对小的微扰是稳定的（对体系加温并迅速撤去（将速度乘以一个放大因子，一定步数后再改为原温度下的速度标定因子），对平衡态，体系状态不受扰动；对亚温态，热力学量会有大变化。）

碰撞失稳性: 解经典力学的确定性方程时，由于：算法的误差，如截断误差;浮点数精度的舍入误差;初始值的任意小的不确定性，导致轨迹发生极大变化。确定性运动方程可能产生随机性运动轨迹。但只要保证总能量守恒，这些轨迹都可以对相空间中的允许运动区域进行随机性抽样，保证长时间平均结果不变。分子动力学模拟中，只要保证各态历经性，长时间的计算时间步下，轨迹基本上是随机性的对相空间进行抽样，这时的统计预测是足够好。分子动力学模拟的目标并不仅仅是预测一个已知初始条件的体系将会发生什么，或知道相空间中的代表点的轨迹。而是对其平衡态的统计平均也很感兴趣。因此碰撞失稳性对分子动力学模拟结果来说，问题不是很严重。

等温分子动力学-约束方法: 微正则系综：系统总能量(动能+势能)恒定 正则系综(更常用)：系统温度恒定

定义粒子的瞬时温度 满足则对速度进行修正可使速度与指定温度一致，保持了动能守恒或温度一定，这样，微正则系综过渡到正则系综。

均方位移判断体系固、液态:液态：与时间成正比;固态：常数。也可用平移序参数和关联分布函数作判据。

径向分布函数判断晶态特征:

Monte Carlo算法的缺点: 对于基础风险因素仍然有一定的假设，存在一定的模型风险、如产生的数据序列是伪随机数，可能导致错误结果、计算量很大，且准确性的提高速度较慢

理论双精度浮点计算能力(峰值)=处理器主频×处理器每个时钟周期执行双精度浮点计算

摩尔定律: 当价格不变时，集成电路上可容纳的晶体管数目，约每隔18个月便会增加一倍，性能也将提升一倍。

近几年CPU在主频方面无大提升，主要向多核发展。

国产CPU问题:国内工艺不够，部分受美国限制，即使设计出来，也无法生产;性能有些还不错，但应用生态（兼容性、效率等）短时间内跟不上。

高性能计算:High Performance Computing, HPC.超级计算:Supercomputing并行计:Parallel Computing

通常指使用很多处理器（作为单个机器的一部分）或某一集群中组织的几台计算机(作为单个计算资源操作)的计算系统和环境。

影响高性能计算性能的主要因素:1.硬件：CPU：主频、并发数、 Cache;内存：主频、 CL延迟（CAS Latency，内存存取数据所需的时间）、容量IO能力：缓存、转速、接口速率网络：带宽、延迟2.软件：编译器、数值函数库、并行库3.设置：硬件、操作系统、软件

超算系统CPU特点:1.单核CPU性能并不高:串行程序运行速度主要依赖于CPU主频;单个串行程序速度不会提高，有可能比自己的计算机运行还慢2.超算系统CPU核数非常多

并行计算: 可以加快速度、可以加大规模，更加宏大或微小的尺度、完成单颗CPU无法完成的任务

并行计算设计的分类:1.独享内存：各处理器使用各自内存2.共享内存：多个处理器共享同一内存3.分布式独享内存：每个处理器都有自己的内存处理器之间通过网络传递消息交换信息4. 分布式共享内存：节点内部CPU核心共享节点内内存节点之间分布式内存， CPU之间通过传递消息交换信息。

并行化分解方法:1.任务分解：多任务并发执行2.功能分解：分解被执行的计算3.区域分解：分解被执行的数据

MPI(Message Passing Interface)是一种消息传递接口标准,并不是一种编程语言,实际上是一个消息传递函数库的标准说明,以语言独立的形式来定义这个接口库，并提供了与C/C++和Fortran语言的绑定.

通讯因子定义了进程组内或组间通讯的上下文（具体就是指明通讯链路的数据结构指针）.MPI通过指定通信因子和组来对进程进行一种逻辑上的划分.MPI\_COMM\_WORLD通信因子在MPI环境初始化过程中创建.

在一个通信因子中，每个进程都有一个唯一的整数标识符，称作“进程号”进程号是从0开始的连续整数，编号为:0～进程数-1.

MPI消息包括信封和数据两部分。信封:指出了发送或接收消息的对象及相关信息,含: <源/目进程号,标识,通信域>

数据：是本消息将要传递的内容，含： <起始地址，数据个数，数据类型>

用进程号可以控制程序的不同部分在不同的进程中并行运行

CALL MPI\_COMM\_RANK(MPI\_COMM\_WORLD, MYID, IERR)

IF (MYID == 0) THEN \n 0号进程运行的程序段

ELSE IF (MYID == 1) THEN \n 1号进程运行的程序段

ENDIF

程序相同部分在不同进程中变量值不一样:

CALL MPI\_COMM\_RANK(MPI\_COMM\_WORLD, MYID, IERR) \n A=A+MYID

MPI消息包括信封和数据两部分.信封:指出了发送或接收消息的对象及相关信息，含：<源/目进程号，标识，通信域>数据：是本消息将要传递的内容，含： <起始地址，数据个数，数据类型>

MPI函数的命名规则:函数名形式为MPI\_Class\_action\_subset、 MPI\_Class\_action、MPI\_Action\_subset或MPI\_Action

C语言:MPI和第一个\_后的首字符为大写，其余为小写;Fortran语言：函数名不区分大小写.Fortran函数除了MPI\_Wtime、MPI\_Wtick外，比C函数多一个参数IERROR

在MPI标准中MPI函数采用语言独立的方式说明，参数用IN、 OUT或INOUT标记:IN：变量为输入给函数的，调用后其值不变;OUT：变量不是输入给函数的，其值被函数调用后更新;INOUT：变量既是输入给函数的，调用后也会被更新

MPI\_Init：初始化MPI环境,必须调用;首先调用;调用一次;每个进程都有一个参数表int MPI\_Init( int \*argc, char \*\*\*argv )

MPI\_Comm\_size：返回与该组通信因子相关的进程数，存在SIZE变量中通讯因子必须是组内通讯因子int MPI\_Comm\_size ( MPI\_Comm comm, int \*size )

MPI\_Comm\_rank：返回该进程在指定通信因子中的进程号（0～进程数-1），存在RANK变量中一个进程在不同通信因子中的进程号可能不同int MPI\_Comm\_rank( MPI\_Comm comm, int \*rank )

MPI\_Send：发送buf缓冲区中的count个datatype数据类型的数据发送到dest目的进程，且带有tag标记int MPI\_Send(void\* buf, int count, MPI\_Datatype datatype, int dest, int tag, MPI\_Comm comm)

MPI\_Recv：从指定的source进程接收带有tag标记消息，按datatype数据类型存到buf缓冲区,收到的消息所包含的数据元素的个数最多不能超过count。int MPI\_Recv( void\* buf, int count, MPI\_Datatype datatype, int source, int tag, MPI\_Comm comm, MPI\_Status \*status )

MPI\_Finalize：结束MPI执行环境。该函数一旦被应用程序调用时，就不能调用MPI的其它例行函数(包括MPI\_Init)必须保证在进程调用MPI\_Finalize之前完成与进程有关的所有通信int MPI\_Finalize(void)

目的地（dest）：发送进程指定的接收该消息的目的进程，也就是接收进程的进程号

源（source）：接收进程指定的发送该消息的源进程，也就是发送进程的进程号如该值为MPI\_ANY\_SOURCE表示接收任意源进程发来的消息

标识符（tag）：由程序员指定的为标识一个消息的唯一非负整数值（0-32767）发送操作和接收操作的标识符一定要匹配。对于接收操作来说，如tag指定为MPI\_ANY\_TAG则可与任何发送操作的tag相匹配

通信因子（comm）：包含源与目的进程的一组上下文相关的进程集合。除非用户自己定义（创建）了新的通信因子，否则一般使用系统预先定义的全局通信因子MPI\_COMM\_WORLD

状态（status）:对接收操作，包含接收消息的源进程（source）和标识符（tag）。在C程序中，是个包含三个成员的结构体：status.MPI\_SOURCE：发送数据的进程标识；status.MPI\_TAG：发送数据使用的tag标识；status.MPI\_ERROR：接收操作返回的错误代码。相当于一种接收方对消息的监测机制，并且以其为依据对消息作出不同的处理（当用通配符接受消息时）

阻塞发送:

缓存模式MPI\_BSEND(buf, count, datatype, dest, tag, comm): 只要一可能，消息就被发送。

同步模式MPI\_SSEND(buf, count, datatype, dest, tag, comm): 发送者发一个“请求发送”的消息。接收者存储这个请求。当一个匹配接收登入时，接收者发回一个“允许发送”的消息，这时接收者发送消息;

准备好模式MPI\_RSEND(buf, count, datatype, dest, tag, comm): 发送者把消息拷贝到一个缓存，然后以非阻塞方式发送它(使用与标准发送同样的协议)

非阻塞通信:

通过重叠通信和计算在许多系统能提高性能。当数据已被从发送缓存拷出时，发送完成调用返回

MPI\_ISEND(buf, count, datatype, dest, tag, comm, request)：标准模式

MPI\_IRECV(buf, count, datatype, source, tag, comm, request)：接收，只有一种标准模式

MPI\_IBSEND(buf, count, datatype, dest, tag, comm, request)：缓存模式

MPI\_ISSEND(buf, count, datatype, dest, tag, comm, request)：同步模式

MPI\_IRSEND(buf, count, datatype, dest, tag, comm, request)：准备好模式注：多了个请求句柄（request参数），在C语言中为MPI\_Request类型。

等待通信完成和检测通信是否完成

MPI\_WAIT(request, status)：等待检测的通信完成

MPI\_WAITANY(count, array\_of\_requests, index, status)：等待任意一个检测的通信完成

MPI\_WAITALL(count, array\_of\_requests, array\_of\_statuses)：等待所有检测的通信都完成

MPI\_WAITSOME(incount, array\_of\_requests, outcount, array\_of\_indices,array\_of\_statuses)：等待有些检测的通信完成

MPI\_TEST(request, flag, status)：测试检测的通信是否完成

MPI\_TESTANY(count, array\_of\_requests, index, ag, status)：测试是否任意一个检测的通信完成

MPI\_TESTALL(count, array\_of\_requests, ag, array\_of\_statuses)：测试是否所有要检测的通信都完成

MPI\_TESTSOME(incount, array\_of\_requests, outcount, array\_of\_indices,array\_of\_statuses)：测试是否有些检测的通信完成

MPI\_REQUEST\_GET\_STATUS(request, ag, status):获取检测的通信的状态，与WAIT和TEST不同，并不释放句柄

MPI\_REQUEST\_FREE(request)：释放某个通信

WAIT是要等待满足某个条件后完成， TEST是检测以后即完成

探测和取消

MPI\_IPROBE(source, tag, comm, flag, status)：非阻塞探测进程source是否发标记为tag的消息来，状态存储在flag

MPI\_PROBE(source, tag, comm, status)：阻塞探测进程source是否发标记为tag的消息来

MPI\_CANCEL(request)：取消通信

MPI\_TEST\_CANCELLED(status, flag)：探测通信是否已取消

坚持式通信请求:当重复发送接收同样参数的信息时，可以利用下面坚持式通信方式得到更高的通信性能

初始化发送和接收

MPI\_SEND\_INIT(buf, count, datatype, dest, tag, comm, request)

MPI\_BSEND\_INIT(buf, count, datatype, dest, tag, comm, request)

MPI\_SSEND\_INIT(buf, count, datatype, dest, tag, comm, request)

MPI\_RSEND\_INIT(buf, count, datatype, dest, tag, comm, request)

MPI\_RECV\_INIT(buf, count, datatype, dest, tag, comm, request)

MPI\_START(request)：开始某个通信

MPI\_STARTALL(count, array\_of\_requests)：开始所有通信

MPI\_REQUEST\_FREE(request)：释放某个通信

发送接收和空进程

MPI\_SENDRECV(sendbuf, sendcount, sendtype, dest, sendtag, recvbuf, recvcount,recvtype, source, recvtag, comm, status)：发送接收使用不同的缓冲区

MPI\_SENDRECV\_REPLACE(buf, count, datatype, dest, sendtag, source, recvtag,comm, status):发送接收使用同样缓冲区

MPI\_PROC\_NULL：空进程（虚拟进程），对此进程的发送或接收立即返回，在某些情况下编程采用空进程会更加方便

注意：目的地与源、数据个数、数据类型等都可以不一样，可以用普通的MPI\_SEND与MPI\_RECV来接收，只要匹配集合，没必要非也得用MPI\_SEDNRECV等

集合通信:所有组成员间的栅障同步（barrier synchronization）、一个成员到组内所有成员的广播（broadcast）通信

一个成员从所有组成员收集（gather）数据、一个成员向组内所有成员分散（scatter）数据、组内所有成员都接收结果（allgather）、组内所有成员到所有成员间的分散/收集数据操作（alltoall）、全局归约（global reduction）操作如求和（sum），求极大值（max），或用户自定义的操作，结、果返回给所有的组成员或仅返回给其中的一个成员、组合归约（combined reduction）和分散操作、组内所有成员上的搜索（scan）操作（也称前置操作）

注意：对所有集合通讯，每个进程都需要执行到这里才能执行，千万不要用进程号或其他方式限制到某个进程来执行，也不需要再采用 MPI\_SEND和 MPI\_RECV等来发送接收，否则会导致死锁

数据的打包与解包:一些通信库为发送不连续数据提供打包/解包函数，用户在发送前显式地把数据包装到一个连续的缓冲区，在接收之后从连续缓冲区中解包。在多数情况下允许有一个派生数据类型不显式打包和解包，用户指明要发送的和接收的数据的分布，通信库直接访问一个不连续的缓冲区。

MPI\_PACK(inbuf, incount, datatype, outbuf, outcount, position, comm)：打包

MPI\_UNPACK(inbuf, insize, position, outbuf, outcount, datatype, comm)：解包

MPI\_PACK\_SIZE(incount, datatype, comm, size)：获取打包数据占用空间的上界

一个打包单元可用MPI\_PACKED类型发送。任何点到点或收集式通信功能可被用来从一个进程移动构成打包单元的字节序列到另一个进程中。该打包单元这时可用任何接收操作使用任意数据类型来接收：类型匹配的规定对于以MPI\_PACKED类型发送的消息不再起作用。以任何类型发送的消息（包括MPI\_PACKED类型）都可用MPI\_PACKED类型接收后被调用MPI\_UNPACK来解包。

通信子：将所有这些观点都封装起来以便为MPI中的所有操作提供适当的机会。通信子可分为两种：组内通信子用于一组进程内的操作；组间通信子用于两组进程间的点对点通信。

缓冲区：通信子提供了一个缓冲机制允许人们将与MPI固有特征等价的新的属性联系到通信子上。高级用户可利用这一机制进一步修饰通信子并通过MPI实现一些通信子函数。

组：组定义了一个进程的有序集合，每一进程具有一个序列号，而且为组间进程通信定义低级名字也是由组完成的。这样组在点对点通信中为进程名字定义了一个范围。另外组还定义了集合操作的范围。在MPI中可从通信子中对组进行分别维护，但是只有通信子才能用于通信操作。

上下文：为MPI提供了拥有分离安全的消息传送空间的能力。

拓扑是加在内部通信子上的额外、可选的属性，它不能被加在组间通信子上。对于一组进程（通信子内部），拓扑能够提供一种方便的命名机制，另外，可以辅助运行时间系统，将进程映射到硬件上。MPI中的进程组是n个进程的集合，组中的每一进程被赋予一个从0到n-1的标识数。在许多并行应用程序中，进程的线性排列不能充分地反映进程间在逻辑上的通信模型（通常由基本问题几何和所用的数字算法所决定），进程经常被排列成二维或三维网格形式的拓扑模型，而且，通常用一个图来描述逻辑进程排列，我们指这种逻辑进程排列为“虚拟拓扑”。在虚拟进程拓扑和底层的物理硬件拓扑之间有一个清晰的差别。进程分配到物理处理器上时，虚拟拓扑可以由系统开发，假设这会帮助提高所给机器的通信性能，然而，这种映射是如何来完成的，已超出了MPI的范围。另外，虚拟拓扑的描述仅依赖于应用，并且独立于机器。

MPI环境管理:用于获取和在合适的地方设置相关于MPI实现及执行环境（例如错误处理）的不同参数的例程。这里描述了用于进入和离开MPI执行环境的过程。

MPI派生数据类型:到此为止，所有的点对点通信只牵涉含有相同数据类型的相邻缓冲区，这对两种用户限制太大。一种是经常想传送含有不同数据类型值的消息的用户；另一种是经常发送非连续数据的用户。一种解决的办法是在发送端把非连续的数据打包到一个连续的缓冲区，在接收端再解包。这样做的缺点在于在两端都需要额外的内存到内存拷贝操作，甚至当通信子系统具有收集分散数据功能的时候也是如此。而MPI提供说明更通用的，混合的非连续通信缓冲区的机制。直到执行（implementation）时再决定数据应该在发送之前打包到连续缓冲中，还是直接从数据存储区收集。这里提供的通用机制允许不需拷贝，而是直接传送各种形式和大小的目标。我们并没有假设MPI库是用本地语言描述的连续目标。因此，如用户想要传送一个结构或一个数组部分，则需要向MPI提供一个通信缓冲区的定义，该定义用问题模仿那个结构和数组部分的定义。这些工具可以用于使库设计者定义能够传送用本地语言定义的目标的通信函数：通过对可获得的符号表或虚拟向量（dopevector）的定义解码即可。这种高级通信功能不是MPI的部分。

MPI动态进程管理:组间通信域、动态创建新的MPI进程、独立进程间的通信、基于socket的通信

MPI远程存储访问:远程存储访问即直接对非本地的存储空间进行访问、一个进程对另外一个进程的存储区域进行直接访问

MPI并行I/O:显式偏移的并行文件读写、多视口的并行文件并行读写、共享文件读写、分布式数组文件的存取

影响MPI程序效率的主要因素:1.并行程序由于进程/线程之间有时候需要相互依赖，某个进程/线程需要等另外的进程/线程完成才可以进行下一步计算，那么会产生等待，导致效率无法100%。2.MPI利用通信进行数据交换，影响MPI程序的主要因素为：网络延迟、网络带宽。

解决措施：1.根据不同的网络情况决定发送接收的频率及通信包的大小等，比如降低通讯频率，将几次发送接收合并为一次等2.改进并行模型及算法，有时候比从所用函数等代码方面单纯优化程序对效率影响更大

OpenMP是通过在源代码中添加一些OpenMP编译指令、调用OpenMP库函数来实现在共享内存的系统上并行运行的一种标准

OpenMP格式: C/C++中利用pragma预处理指令做为OpenMP的指令，语法格式如下：#pragma omp directive−name [clause[ [,] clause]…] new−line

每个指令以#pragma omp开始。指令其余部分按照C/C++编译指令标准，并区分大小写。#之前和之后可有空白，且有时必须用空白来分隔指令中的文字。#pragma omp之后的预处理目标在编译时将被宏替换。每个OpenMP执行指令必须应用于至少一个随后的语句，并且必须是一个结构块。每行只能有一个OpenMP指令。参数的位置无前后之分，可重复。注：[ ]表示内部的参数是可选的

内在控制变量（ICV）:内置的用于控制OpenMP程序行为的变量，如存储线程数、线程号等。影响并行块的内在控制变量：dyn-var：控制影响的并行区域是否允许动态调整线程数，每任务一个ICV副本nest-var：控制影响的并行区域是否允许嵌套并行，每任务一个ICV副本nthreads-var：控制影响的并行区域的线程数，每任务一个ICV副本thread-limit-var：控制程序最大允许参与的线程数，整个任务共用一个ICV副本max-active-levels-var：控制嵌套的激活的并行区域的最大数，整个任务共用一个ICV副本。影响循环区域操作的内在控制变量：run-sched-var：控制循环区域的实时调度策略，每任务一个ICV副本def-sched-var：控制循环区域的默认实时调度策略，整个任务共用一个ICV副本。影响程序执行的内在控制变量：stacksize-var：控制OpenMP实现产生的线程的堆栈（stack）的大小，整个任务共用一个ICV副本wait-policy-var：控制等待线程的期望行为，整个任务共用一个ICV 副本

并行结构:指明随后区域中的代码将并行执行,并且并行是可以嵌套的

#pragma omp parallel [clause[ [, ]clause] ...] new−line \n structured−block

工作共享构造(Worksharing Constructs)会在执行到此结构的线程中分配与之相关区域的到各线程执行。线程会执行在每个正在执行任务的情况下隐含的部分区域。在每个工作共享构造的入口处无需进行同步，但在工作共享区域的结束处，除非有nowait参数，否则默认会进行同步。如有nowait，那么执行到此处时将会忽略此工作共享区域的同步，早执行完此结构的线程会直接执行下面的指令而不需等待其它进程执行到此,也不需要进行刷新(flush)操作

限制：1.工作共享区域必须为所有组内线程都执行到或都不执行到2.一组内每个线程执行到工作共享区域和同步区域的顺序必须一致

循环结构指明一个或多个与之相关的循环迭代将被一组线程执行

#pragma omp for [clause[[,] clause] ... ] new−line \n for−loops

分块结构指明不同的线程执行不同区域内的代码

#pragma omp sections [clause[[,] clause] ...] new−line{[#pragma omp section new−line] \n structured−block \n [#pragma omp section new−line \n structured−block ] \n ...}

单执行结构指明相关代码只能有一个线程（并不必须是雇主线程， 0号线程）执行，一般为先到的线程执行

#pragma omp single [clause[[,] clause] ...] new−line \n structured−block

工作共享结构(workshare Construct)指明相关的代码被分割成不同的单元以供不同线程执行，不同单元只能被某个线程执行一次

C/C++：不支持此OpenMP指令

联合并行工作共享结构实际上是在并行结构中嵌套共享结构的一个快捷方式，这些指令将同时指明一个工作。共享结构具有并行性，而不需其它语句另行指明。允许一些并行结构和工作共享结构都允许的特定参数。如程序含有对并行结构或工作共享结构具有不同行为的参数，那么其行为将不可预测。

并行循环结构指明一个或多个相关的循环迭代将被一组线程并行执行

#pragma omp parallel for [clause[[,] clause] ... ] new−line \n for−loops

并行分块结构指明与之相关的不同区域的代码将被不同线程并行执行

#pragma omp parallel sections [clause[[,] clause] ...] new−line{ [#pragma omp section new−line] \n structured−block \n [#pragma omp section new−line \n structured−block ] \n ...}

并行工作共享结构指明相关的代码被分割成不同的单元以供不同线程并行执行,不同单元只能被某一线程执行一次

C/C++：不支持此OpenMP指令

任务结构定义一个明确的任务

#pragma omp task [clause[[,] clause] ...] new−line \n structured−block

雇主结构指明结构块需要雇主线程（主线程， 0号线程）执行

#pragma omp master new−line \n structured−block

临界结构指明结构块在同一时间只能有一个线程执行，一个线程执行完毕，其它线程才能执行该处代码

#pragma omp critical [(name)] new−line \n structured−block

栅栏结构指明在此处有一个栅栏，需进行显式同步，即需本组内所有线程都要运行到此后才执行后续代码(栅栏结构必须在所有线程的同样顺序中)

#pragma omp barrier new−line

任务等待结构指明需要自当前任务开始就需等待产生的子任务完成

#pragma omp taskwait newline

原子结构用于确保指明的存储区域对某项操作进行原子更新，而不使其暴露给多个同步进行写的线程，避免同时多个线程对这些变量进行更新

#pragma omp atomic new−line \n expression−stmt

刷新结构指明对指定的变量执行内存刷新操作以保证线程此时看到这些变量在内存中的值一致

#pragma omp flush [(list)] new−line

有序结构指明循环结构中结构块的执行按照循环的迭代顺序，将排序此结构块中的执行顺序，并允许结构块之外的代码可以并行执行

#pragma omp ordered new−line \n structured−block

结构内引用的变量的数据共享属性法则:结构中引用变量的数据共享属性可为预先决定的、显示决定的或隐式决定的之一。封闭结构中的firstprivate、 lastprivate或reduction参数声明的变量会导致此结构中变量采用隐式的

引用方式，并遵守以下规则：threadprivate指令指明的变量是线程私有的、结构中某个范围内申明的具有自动存储周期的变量是私有的、具有堆栈分配存储的变量是共享的、静态数据成员是共享的、关联的for循环或 parallel for循环结构中的循环迭代变量是私有的、不具有易变成员的保留常数变量是共享的、结构中某个范围内声明的静态变量是共享的。除了在以下情形中，具有预定义数据共享属性的变量也许未在数据共享属性参数中列出，对这些例外，在数据共享属性参数中允许列出一些预定义的变量，且将取代这些变量的预先定义数据共享属性：for循环或 parallel for循环结构中的循环迭代变量可在private或lastprivate参数中被列出。数据共享法则:具有显式决定或隐式决定的数据共享属性的变量：1.具有显式决定的数据共享属性的变量：在一个给定的结构中被引用并且在此结构的数据共享参数中被列出的变量2.具有隐式决定的数据共享属性的变量：在一个给定的结构中被引用，但没有被预先决定数据共享属性，且没有在此结构的数据共享属性参数中列出的变量。隐式数据共享法则:1.在parallel或task结构中，如存在default参数，那么这些变量的数据共享属性由default参数决定2.在parallel结构中，如没有default参数，那么这些变量是共享的3.对于不是task的结构来说，如没有default参数，那么这些变量将从此封闭上下文中继承数据共享属性4.在task结构中，如没有default参数，那么在所有封闭结构并一直到最内部的封闭parallel结构中变量被决定为共享的5.在task结构中，如没有default参数且数据共享属性没有被上述规则决定，那么变量是firstprivate的。区域而非结构中的数据共享属性规则:1.在被调用子程序中声明的静态变量在此区域中是共享的2.不具有易变成员且在被调用子程序中声明的保留常数变量是共享的3.除非是在threadprivate指令中出现，否则区域中在被调用子程序中引用的文件范围或者名字空间范围变量是共享的4.具有堆栈分配存储的变量是共享的5.除非是在threadprivate指令中出现，否则静态数据成员是共享的6.区域中被调用子程序的通过引用传递的正式参数继承关联的实际参数数据的数据共享属性7.区域中被调用子程序中的其它变量是私有的

threadprivate指令指明变量是线程私有的

#pragma omp threadprivate(list) new−line

default：设置默认的共享方式

default(shared | none)

其中：shared：线程共享

reduction：声明对变量进行规约操作

reduction(operator:list)

数据复制参数copyin(list):在执行parallel结构时,将主线程的线程私有变量的值复制给所有其它线程copyprivate(list):提供一种机制可以将隐式任务数据环境的某个私有变量的值广播给此parallel结构中的其它数据环境

线程嵌套的规则如下1.工作共享区域：可没被嵌套封闭在工作共享、显式task、 critical、 ordered或master区域2.barrier区域：可没被嵌套封闭在工作共享、显式task、 critical、 ordered或master区域3.master区域：可没被嵌套封闭在工作共享或显式task区域4.ordered区域：可没被嵌套封闭在critical或显式task区域5.ordered区域：必须被嵌套封闭在具有ordered参数的循环区域或并行循环区域中6.critical区域：可没被嵌套（封闭或其它）在具有同样名字的critical 区域中，此限制并不充分能防止一个死锁

简单锁子程序

omp\_init\_(nest\_)lock分别初始化简单锁和嵌套锁void omp\_init\_(nest\_)lock(omp\_(nest\_)lock\_t \*lock);

omp\_destroy\_(nest\_)lock分别销毁简单锁和嵌套锁void omp\_destroy\_(nest\_)lock(omp\_nest\_lock\_t \*lock);

omp\_set\_(nest\_)lock分别设置简单锁和嵌套锁void omp\_set\_nest\_lock(omp\_(nest\_)lock\_t \*lock);

omp\_unset\_(nest\_)lock分别复位简单锁和嵌套锁void omp\_unset\_(nest\_)lock(omp\_(nest\_)lock\_t \*lock);

omp\_test\_(nest\_)lock分别测试简单锁和嵌套锁，如存在则设置int omp\_test\_(nest\_)lock(omp\_(nest\_)lock\_t \*lock);

获取计时器相关变量：

omp\_get\_wtime获取以秒为单位的当前时间double omp\_get\_wtime(void);

omp\_get\_wtick获取计时器的精度double omp\_get\_wtick(void);

利用时间程序可以计算程序中某段计算花费的时间

double start, end;

start = omp\_get\_wtime();

... work to be timed ...

end = omp\_get\_wtime();printf(”Work␣took␣%f␣seconds\n”, end − start);

MPI:进程级、分布式内存、显式、可扩展性好

OpenMP：线程级（并行粒度）、共享内存、隐式（数据分配方式）、可扩展性差

OpenMP采用共享内存，只适应SMP、 DSM架构，不适合集群架构

MPI适合于广泛架构，但编程模型复杂：1.需要分析及划分应用程序问题，并将问题映射到分布式进程集合2.需要解决通信延迟大和负载不平衡两个主要问题3.调试麻烦4.程序可靠性差，一个进程出问题，整个程序将错误

影响OpenMP并行程序的因素:OpenMP程序为共享内存的，各线程可以直接读写其它线程的内存，无需通过网络来进行数据交换：1.避免频繁设置内存私有或共享2.降低写同样内存的操作，为了保证正确，各线程写同样内存时，需要排队，导致效率降低

常见数值函数库：

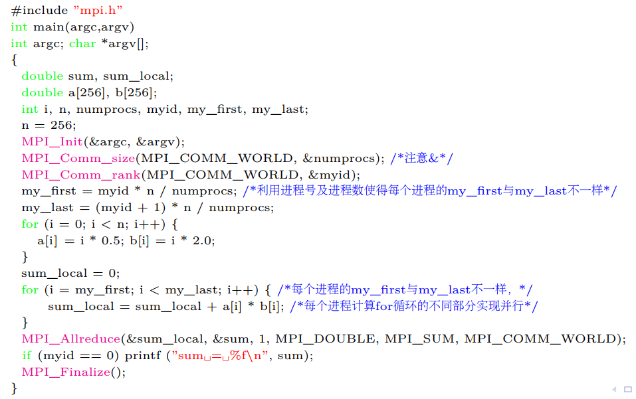
一般数值函数库:基本线性代数库(BLAS)、线性代数库(LAPACK)、可扩展性线性代数库(ScaLAPACK)、傅立叶变换程序(Fourier Transform functions, FFT)、Netlib(http://www.netlib.org)：收集了大量常见的数学函数库

数值函数库的集合产品:MKL: Intel Math Kernel Library、AMD CPU库、IMSL: Rogue Wave International Mathematics and Statistics Library

数据处理、计算可视化软件:Origin、Mathematica、Maple、MATLAB、IDL、GNUPLOT、Python

作业调度系统:在一个大型系统内部，通常需要处理一些自动化运行的任务，通常会采用系统自带的crontable的定时任务完成。但是，很多情况下，是多个作业，彼此先后执行，共同完成任务。在这样的情况下，定时任务存在两个明显的问题：1.浪费了大量的系统等待时间2.假设两个作业，第一个作业必须在第二个作业前运行，如第二个作业先运行，就会有灾难性的后果，对于定时任务而言，解决任务这样两个作业优先级的问题是只能把任务一的运行时间安排在二之前，不能完全满足前面的假设，但是对于作业调度器而言，安排作业的优先级，是最基本的功能，简直是小Case。

常见作业调度系统:LSF(IBM Platform Load Sharing Facility)、SLURM(Simple Linux Utility for Resource Management)、PBS(PBS Pro、 OpenPBS、 TORQUE)、Maui、Condor、SGE(Oracle Grid Engine)、LoadLeveler(IBM Tivoli Workload Scheduler)

TEX/LATEX:优点:1.排版的效果非常整齐漂亮2.排版的效率非常高3.非常稳定，从1995年到现在， TEX系统只发现过一个bug4.排版科技文献，尤其是含有很多数学公式的文献特别方便、高效。现今没有一个排版软件在排版数学公式上面能和TEX/LATEX相媲美5.纯文本，可用任何编辑器编辑；缺点：不是WYSIWYG，需编译后才知道效果

