

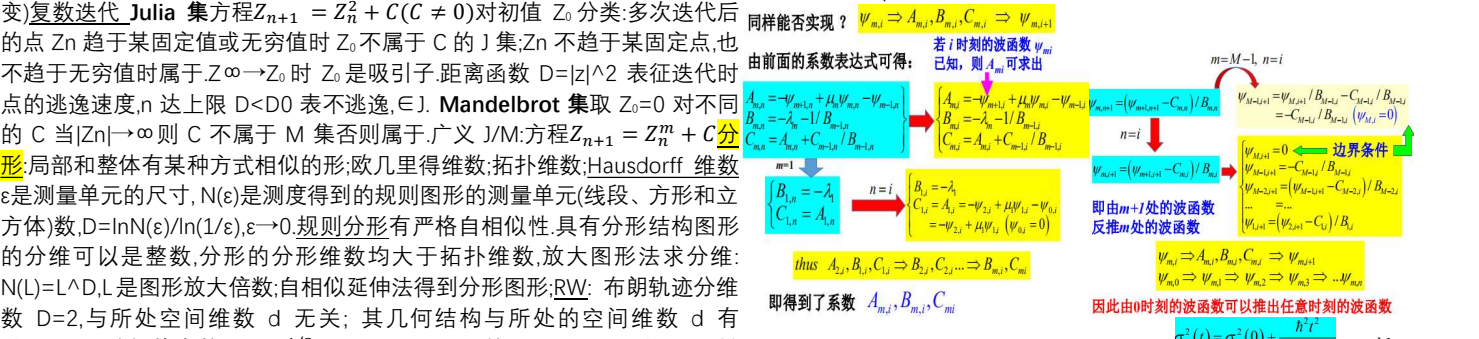
混合输入迭代法 $x_{n+1} = \alpha x_n + (1-\alpha)x_{n-1}$ 混沌是一种确定性系统中出现的有限状态的元胞组成的元胞空间上,按照一定局部规则在离散的时间维上演(或决定论规律所产生的)类似随机的过程;具有初值敏感性(由系统本身确定).化的动力学系统.Wolfram 一维元:演化规则半径 r 则 $2r+1$ 个近邻,k 种状态则一维迭代 Logistic $x_{n+1} = \lambda x_n(1-x_n)$ ($0 \leq x \leq 1, 0 \leq \lambda \leq 4$) (是一维非线性迭代/映射)终态: $k \wedge (2r+1)$ 种条件;对应 k 个次态有 $k \wedge (k \wedge (2r+1))$ 种元胞自动机.(2d 近邻数换含绝灭(所有初值得到 0)/定态(除 0 任意初值结果一样)/倍周期分岔(除 0、不动自己的邻域格座数 z)).编码:第一行相邻状态编号第二行其二进制第三行单格点最终均是 $x_1 \times x_2 \times \dots$ /混沌(除 0、不动点最终互不重复)logistic[0,1]绝灭[1,3]定态次态.演化:结合边界条件(周期首尾相接,反射,定值,随机)推导.长期行为:均匀(3,3.57]倍周期分岔(3.57,4]混沌和周期窗口.倍周期分岔 $T=2^n$ 混沌后的周期(全是 0 或 1).定态或周期循环,混沌,混沌的边缘.邻居数:冯·5,摩尔 9 扩展摩'25 窗口奇 $\times 2^n$;Feigenbaum 常数:倍周期前后分岔间距比值趋于常数马格勒斯 2×2 .|表决与退火:正方/9 邻/多少少数决定;Ising 自旋(Q2):正方 5 邻局 $\lambda_\omega \cdot \lambda_m = 48^m$ ($m \gg 1$) (λ_m 是 m 变 $2m$ 分岔点)A 依赖于迭代函数, δ 不依赖于迭

代函数. $d_m/d_{m+1} \rightarrow \alpha$ ($m \rightarrow \infty$) 普适常数 α , δ 即 F' 常数.混沌定性特征:内随机性信息 $s(r,t)$ 表有无粒入射,HPP 正对反射;FHP 正对等概率左右偏 60° ,3 粒夹 120° 分维 (维数描述系统运动轨道在相空间的行为特征),普适性(不依赖具体系数反弹,余不变.沙堆:M 邻/绝对/一定概率)受重力运动.蚂蚁:邻定向一格多蚁;森林火灾:燃烧/蔓延/生长/自燃;管虫:栖息状态/藏匿计时.有限差分方法:方 (x_n, y_n) 的数值都不重复,即系统处于混沌状态,是一种奇异吸引子.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.简单吸引子维数是整数;吸引子本身不可分解,在相空间体积为 0.奇异吸引子具吸引子性质,奇异性(初值敏感/分数维/非周期性)三维常微分 Lorenz: $x \propto$ 对流运动的强度 $y \propto$ 水平方向温度的变化 $z \propto$ 竖直方向温度的变化;螺旋线永远被限制在有限的空间内却永不相交、永无止境,是一种奇异吸引子 Rössler 吸引子: $a, b, c > 0$

$$\begin{cases} x_{n+1} = 1 - \alpha x_n^2 + y_n \\ y_{n+1} = b x_n \end{cases} \quad \begin{cases} \frac{dx}{dt} = -x - y \\ \frac{dy}{dt} = x + \alpha y \end{cases}$$

 $a=1.4, b=0.3$ $\frac{dx}{dt} = -x - y$ $\frac{dy}{dt} = x + \alpha y$

Lyapunov 指数:用来表示初值敏感性是否出现以及敏感的程度,定量刻画混沌运动. $\lambda' = \lim_{n \rightarrow \infty} \frac{\ln |dx_n/dx_0|}{n} > 0$ 混沌, < 0 稳定(周期和不动点), $= 0$ (稳定边界,初始误差不变)复数迭代 Julia 集方程 $Z_{n+1} = Z_n^2 + C$ ($C \neq 0$) 对初值 Z_0 分类:多次迭代后的点 Z_n 趋于某固定值或无穷值时 Z_0 不属于 C 的 J 集; Z_n 不趋于某固定点,也不趋于无穷值时属于 $Z \rightarrow Z_0$ 时 Z_0 是吸引子.距离函数 $D = |z|^{1/2}$ 表征迭代时点的逃逸速度, n 达上限 $D < D_0$ 表不逃逸, $\in J$. Mandelbrot 集取 $Z_0 = 0$ 对不同的 C 当 $|Z_n| \rightarrow \infty$ 则 C 不属于 M 集否则属于.广义 J/M:方程 $Z_{n+1} = Z_n^m + C$ 分形:局部和整体有某种方式相似的形;欧几里得维数;拓扑维数;Hausdorff 维数 ϵ 是测量单元的尺寸, $N(\epsilon)$ 是测度得到的规则图形的测量单元(线段、方形和立方体)数, $D = \ln N(\epsilon) / \ln(1/\epsilon)$, $\epsilon \rightarrow 0$.规则分形有严格自相似性.具有分形结构图形的分维可以是整数,分形的分形维数均大于拓扑维数,放大图形法求分维: $N(L) = L^D$, L 是图形放大倍数;自相似延伸法得到分形图形;RW: 布朗轨迹分维数 $D=2$,与所处空间维数 d 无关;其几何结构与所处的空间维数 d 有关, $d=1,2,3$ 访问格点数 $N \propto N^{1/2}$, $N/\log N$, N . SAW 的 $D = (d+2)/3$. 2 维 DLA 模拟:取正方形点阵,中心置种子,远处任意位置(可设置一圆或正方边界,边界距中心的距离 $>>$ 欲模拟图形尺寸)随机产生一粒子进行 RW,粒子与中心种子或团簇相接触时被粘住成为团簇的一部分,如此往复.(生长中心少,团簇范围大,生长缓慢的模拟,实验枝杈粗;温度高/原子有能量/多中心/原子不是远处产生)不规则分形维数的测定:粗糙曲线的圆规维数,半径 ϵ , $N \sim \epsilon^{-d}$



MPI_INIT	启动MPI环境
MPI_COMM_SIZE	确定进程数
MPI_COMM_RANK	确定自己的进程标识符
MPI_SEND	发送一条消息
MPI_RECV	接收一条消息
MPI_FINALIZE	结束MPI环境

Monte Carlo 算法缺点:模型风险;伪随机数发来的消息;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(0-32767)发送和接收操作的标识符要匹配;接收 t'如为 MPI_ANY_TAG 则与任何双精度浮点计算能力(峰值)=处理器主频×发送 t'相匹配;通信因子 comm:包含源与目的进'的一组上下文相关的进'集合;处理器每个时钟周期执行双精度浮点运算除非用户创建新通信因子,否则是系统预先定义的全局通信因子的次数×系统中处理器核心数;超算集群结MPI_COMM_WORLD;状态 status: 对接收操作包含接收消息的源进'和标识符;主节点:管理、IO、用户登录;计算节点:执行计算任务,多个节点;各节点间:符;接收方对消息的监测机制,并且以其为依据对消息作不同的处理(当用通配通过网络连接;操作系统:多 Linux 或 Unix;用户使用:远程联网使用,比如通过符接受消息时)C: 结构体 {status.MPI_SOURCE 发送数据的进'标识 SSH,需要通过作业调度系统提交作业,排队运行;性能主要影响:硬件:CPU[主status.MPI_TAG 发送数据使用的 tag 标识 status.MPI_ERROR 接收操作返回频、并发数、Cache]内存[主频、CL 延迟(CAS Latency,内存存取数据所需的时的错误代码)Fortran:含 MPI_STATUS_SIZE 个整数的数组点对点通信:阻塞发间)、容量]IO 能力[缓存、转速、接口速率]网络[带宽、延迟];软件:编译器、数送:标准模式;MPI_BSEND:缓存模式(存着随时发);MPI_SSEND:同步模式(一同值函数库、并行库;设置:硬件,操作系统,软件;超算系统 CPU 特点:单核 CPU 性步发送的完成不表示发送缓存能被再使用,表接收者已开始执行匹配接收,两能不高,CPU 核数多;并行计算设计分类: 独享内存:各处理器使用各自内存;阻塞通信没聚会均不完成;S 发请求发 R 存,匹'登入后 R 发允许发);共享内存:多个处理器共享同一内存,OpenMP、MPI; 分布式独享内存:每个处MPI_RSEND:准备好模式(S 把消息拷到缓存后以非阻塞方式发送它(与标准发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;MPI; 分布式送同样的协议)仅 R 要匹配接收登入,参数一样;非阻塞通信 MPI_ISEND 标准共享内存:节点内部 CPU 核心共享节点内内存;节点之间分布式内存,CPU 间模式;MPI_IRECV 接收(仅一种标准模式);MPI_BSEND 缓存模式;MPI_ISSEND 通过传递消息交换信息 OpenMP+MPI;并行化分解方法: 任务分解:多任务并同步模式;MPI_IRSEND 准备好模式;最后多个请求句柄 request 参数发执行;功能分解:分解被执行的计算;区域分解:分解被执行的数据.MPI 是消(MPI_Request 类型)等待通信完成和检测通信是否完成 WAIT 等满足某条件息传递接口标准不是编程语言.MPI 程序一般结构:程序开始-包含 MPI 头文后完成,TEST 检测后即完成;MPI_WAIT 等待检测的通信完成;MPI_WAITANY 件-初始化 MPI 环境-消息交换处理及计算等-退出 MPI 环境-程序结束 等待任意一个检测的通信完成;MPI_WAITALL 等待所有检测的通信都完头文件:MPI 程序要求所有包含 MPI 调用的程序应加入 MPI 文头文件;MPI_WAITSSOME 等待有些检测的通信完成;MPI_TEST 测试检测的通信是 C:#include "mpi.h"mpif.h 是文件名,linux 必须小写.通信因子:定义进'组内或组否完成;MPI_TESTANY 测试是否任意一个检测的通信完成 MPI_TESTALL 测试间通讯的上下文(指明通讯链路的数据结构指针);MPI 通过指定通信因子和组是否所有要检测的通信都完成 MPI_TESTSOME 测试是否有些检测的通信完对进'进行逻辑上的划分;MPI_COMM_WORLD 通信因子在 MPI 环境初始化过成 MPI_REQUEST_GET_STATUS 获取检测的通信的状态,与 WAIT 和 TEST 不程中创建;进'号(rank):一个通信因子中对各进'的唯一整数标识符,0~进'数-1;同,不释放句柄 MPI_REQUEST_FREE(request): 释放某个通信;探测和取消作用:控制程序的不同部分在不同的进'中并行运行\使程序相同部分在不同IPROBE 非阻塞探测进'source 是否发标记 tag 的消息来状态存储在进'中变量值不一样

```
CALL MPI_COMM_RANK(MPI_COMM_WORLD, MYID, IERR)
IF (MYID == 0) THEN
  0号进程运行的程序段
ELSE IF (MYID == 1) THEN
  1号进程运行的程序段
ENDIF

CALL MPI_COMM_RANK(MPI_COMM_WORLD, MYID, IERR)
A=A+MYID
```

MPI 消息:信封(发送/接收消息的对象及相关信息<源/目进'号,标识,通信域>)+数据(本消息将要传递的内容<起始地址,数据个数,数据类型>).隐含约定:MPI 函数命名规则:函数名形式 MPI_Class_action_subset、MPI_Class_action、MPI_Action_subset、MPI_Action;C 语言:MPI 和第一个_型限制;影响 MPI 程序的主要因素网络延迟&网络带宽.的首字符大写,其余小写;Fortran:函数名不区分大小写,除 MPI_Wtime、OpenMP 编译指令+库函数+环境变量;每行只能有一个 OpenMP 指令;各任 MPI_Wtick 外,比 C 函数多一个参数 IERROR;MPI 函数采用语言独立的方式说务的内在控制变量的工作方式:子任务继承,函数只更新相关;当执行到指定明,参数:IN:变量输入给函数,调用后值不变;OUT:变量不是输入给函数的,其值schedule(runtime)的循环工作共享区域时,所有组成绑定 parallel 区域的任务被函数调用后更新;INOUT:变量既是输入给函数的,调用后更新;MPI 基本函数区域的 run-sched-var 必须有同样的值,否则行为将无法预料.(启动结束 MPI 环境,识别进',发送接收消息;)是变量地址的参记得加&!!! 工作共享构造:在执行到此结构的线'中分配与之相关区域的到各线'执行;在 MPI_Init:初始化 MPI 环境,必须调用、首先调用、调用一次;各进'都有一参数每个工'的入口处无需进行同步但在工'结束处,除非有 nowait 参数,否则默认表 C: int MPI_Init(int *argc, char ***argv)Fortan(行为函数+变量声会进行同步,如有,执行到此处时将会忽略此工作共享区域的同步,早执行完此明);MPI_Comm_size:取得进'数(返回与该组通信因子相关的进'数存在变量结构中的线'会直接执行下面的指令而不需等待其它进'执行到此,也不需要进行 SIZE 中,通讯因子必须是组内通讯因子)int MPI_Comm_size(MPI_Comm刷新. 一组内每个线'执行到工作共享区域和同步区域的顺序必须一 comm, int *size)MPI_Comm_rank:取得进'号返回该进'在指定通信因子中的致.OpenMP 工作共享构造: 循环结构 loop structure 分块结构 sections s'(指明进'号存在 RANK 变量中(一个进'在不同通信因子中的进'号可能不同)int明不同的线'执行不同区域内的代码)单执行结构 single(指明相关代码只能有 MPI_Comm_rank(MPI_Comm comm, int *rank)MPI_Send:发送消息(发送一个线'(不必须是雇主即 0 号线)执行,一般为先到的线'执行)工作共享结构 buf 缓冲区中的 count 个 datatype 数据类型的数据发送到 dest 目的进',且带workshare s'(指明相关的代码被分割成不同的单元以供不同线'执行,不同单有 tag 标记) MPI_SEND(buf, count, datatype, dest, tag, comm) IN buf 发元只能被某个线'执行一次)Fortan 独有,写成 structure-block,critical 结构中封送缓存的起始地址(选择型)IN count 发送缓存的元素的个数(非负整数)IN入的语句受此限制 parallel 不受. 联合并行工作共享结构:在并行结构中嵌套 datatype 每个发送缓存元素的数据类型(句柄)IN dest 目的地进'号(整型)IN共享结构快捷方式,这些指令同时指明一工'具有并行性,而不需其它语句,包 tag 消息标志(整型)IN comm 通信子(句柄)int MPI_Send(void* buf, int含并行循环/分块/工作共享结构;任务结构;雇主结构和同步结构;雇主结构指明结构块需要雇主线'执行;临界结构:同一时间唯一线';栅栏结构:强行同步(需在所有线'的同样顺序中);任务等待结构:指明需自当前任务开始就需等待产明的子任务完成;原子结构: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',避免多个更新.刷新结构:对指定的变量执行内存刷新操作以保证线'此时看到这些变量在内存中的值一致;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,允许结构块之外的代码可以并行执行;数据共享属性法则结构内引用的变量:预先,显式,隐式决定的之一;区域内:threadprivate 指明变量是线'私有;数据共享属性参':default 设置默认共享方式 reduction 声明对变量进行规约操作;数据复制参数:copyin 执行 p 结构时将主线'的线'私有变量的值复制给所有其它线';copyprivate 将隐式任务数据环境的某私'的值广播给此 p 结构中其它数据环境;线'嵌套:critical 防死锁;锁子程序(同步用)简单锁':初始化 omp_init_lock(换中间词);destroy 销毁 set 设置 unset 复位 test 测试(如存在则