

遗传算法

提纲

- 一、生物遗传思想和基本遗传算法
- 二、遗传算法原理
- 三、遗传算法的应用

一、生物遗传思想和基本遗传算法

- 1、智能优化算法
- 2、基本遗传算法
- 3、遗传算法的特点

1、智能优化算法

智能优化算法又称为现代启发式算法，是一种具有全局优化性能、通用性强、且适合于并行处理的算法。这种算法一般具有严密的理论依据，而不是单纯凭借专家经验，理论上可以在一定的时间内找到最优解或近似最优解。

常用的智能优化算法

(1) 遗传算法

(Genetic Algorithm ， 简称 GA)

(2) 模拟退火算法

(Simulated Annealing ， 简称 SA)

(3) 禁忌搜索算法

(Tabu Search ， 简称 TS)

.....

智能优化算法的特点

它们的共同特点：都是从任一解出发，按照某种机制，以一定的概率在整个求解空间中探索最优解。由于它们可以把搜索空间扩展到整个问题空间，因而具有**全局优化性能**。

遗传算法起源

遗传算法是由美国的 J. Holland 教授于 1975 年在他的专著《自然界和人工系统的适应性》中首先提出的，它是一类**借鉴生物界自然选择和自然遗传机制**的随机化搜索算法。

遗传算法的搜索机制

遗传算法模拟自然选择和自然遗传过程中发生的**繁殖、交叉和基因突变**现象，在每次迭代中都保留一组候选解，并按某种指标从解群中选取较优的个体，利用遗传算子（**选择、交叉和变异**）对这些个体进行组合，产生新一代的候选解群，重复此过程，直到满足某种收敛指标为止。

2、基本遗传算法

基本遗传算法（Simple Genetic Algorithms，简称SGA，又称简单遗传算法或标准遗传算法），是由Goldberg总结出的一种最基本的遗传算法，其遗传进化操作过程简单，容易理解，是其它一些遗传算法的雏形和基础。

基本遗传算法的组成

- (1) 编码 (产生初始种群)
- (2) 适应度函数
- (3) 遗传算子 (选择、交叉、变异)
- (4) 运行参数

编码

GA 是通过某种编码机制把对象抽象为由特定符号按一定顺序排成的串。正如研究生物遗传是从染色体着手，而染色体则是由基因排成的串。SGA 使用二进制串进行编码。

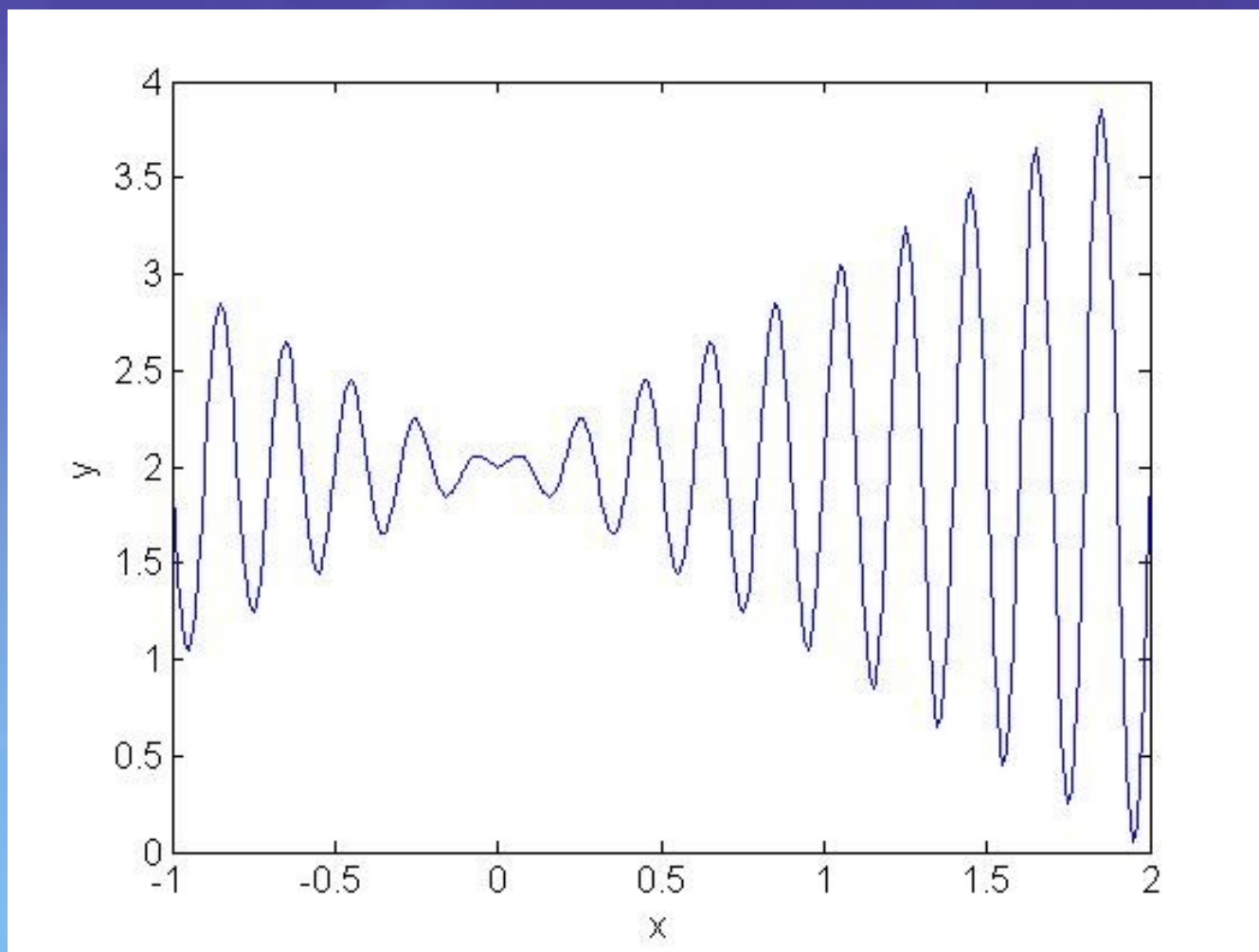
函数优化示例

求下列一元函数的最大值：

$$f(x) = x \cdot \sin(10\pi \cdot x) + 2.0$$

$x \in [-1, 2]$ ，求解结果精确到 6 位小数。

多极值函数



SGA 对于本例的编码

由于区间长度为 3，求解结果精确到 6 位小数，因此可将自变量定义区间划分为 3×10^6 等份。又因为 $2^{21} < 3 \times 10^6 < 2^{22}$ ，所以本例的二进制编码长度至少需要 22 位，本例的编码过程实质上是将区间 $[-1, 2]$ 内对应的实数值转化为一个二进制串（ $b_{21}b_{20} \dots b_0$ ）。

几个术语

个体（染色体）

基因型：1000101110110101000111

基因

解码

编码

表现型：0.637197

初始种群

SGA 采用随机方法生成若干个个体的集合，该集合称为初始种群。初始种群中个体的数量称为种群规模。

适应度函数

遗传算法对一个个体（解）的好坏用适应度函数值来评价，适应度函数值越大，解的质量越好。适应度函数是遗传算法进化过程的驱动力，也是进行自然选择的唯一标准，它的设计应结合求解问题本身的要求而定。

选择算子

遗传算法使用选择运算来实现对群体中的个体进行优胜劣汰操作：适应度高的个体被遗传到下一代群体中的概率大；适应度低的个体，被遗传到下一代群体中的概率小。选择操作的任务就是按某种方法从父代群体中选取一些个体，遗传到下一代群体。SGA 中选择算子采用轮盘赌选择方法。

轮盘赌选择方法

轮盘赌选择又称比例选择算子，它的基本思想是：各个个体被选中的概率与其适应度函数值大小成正比。设群体大小为 n ，个体 i 的适应度为 F_i ，则个体 i 被选中遗传到下一代群体的概率为：

$$P_i = F_i / \sum_{i=1}^n F_i$$

轮盘赌选择方法的实现步骤

- (1) 计算群体中所有个体的适应度函数值（需要解码）；
- (2) 利用比例选择算子的公式，计算每个个体被选中遗传到下一代群体的概率；
- (3) 采用模拟赌盘操作（即生成 0 到 1 之间的随机数与每个个体遗传到下一代群体的概率进行匹配）来确定各个个体是否遗传到下一代群体中。

交叉算子

所谓交叉运算，是指对两个相互配对的染色体依据交叉概率 P_c 按某种方式相互交换其部分基因，从而形成两个新的个体。交叉运算是遗传算法区别于其他进化算法的重要特征，它在遗传算法中起关键作用，是产生新个体的主要方法。SGA 中交叉算子采用单点交叉算子。

单点交叉运算

交叉点

交叉前：

00000|01110000000010000

11100|00000111111000101

交叉后：

00000|00000111111000101

11100|01110000000010000

变异算子

所谓变异运算，是指依据变异概率 P_m 将个体编码串中的某些基因值用其它基因值来替换，从而形成一个新的个体。遗传算法中的变异运算是产生新个体的辅助方法，它决定了遗传算法的局部搜索能力，同时保持种群的多样性。交叉运算和变异运算的相互配合，共同完成对搜索空间的全局搜索和局部搜索。SGA中变异算子采用基本位变异算子。

基本位变异算子

基本位变异算子是指对个体编码串随机指定的某一位或某几位基因作变异运算。对于基本遗传算法中用二进制编码符号串所表示的个体，若需要进行变异操作的某一基因座上的原有基因值为 0，则变异操作将其变为 1；反之，若原有基因值为 1，则变异操作将其变为 0。

基本位变异算子的执行过程

变异点

变异前：

000001110000000010000

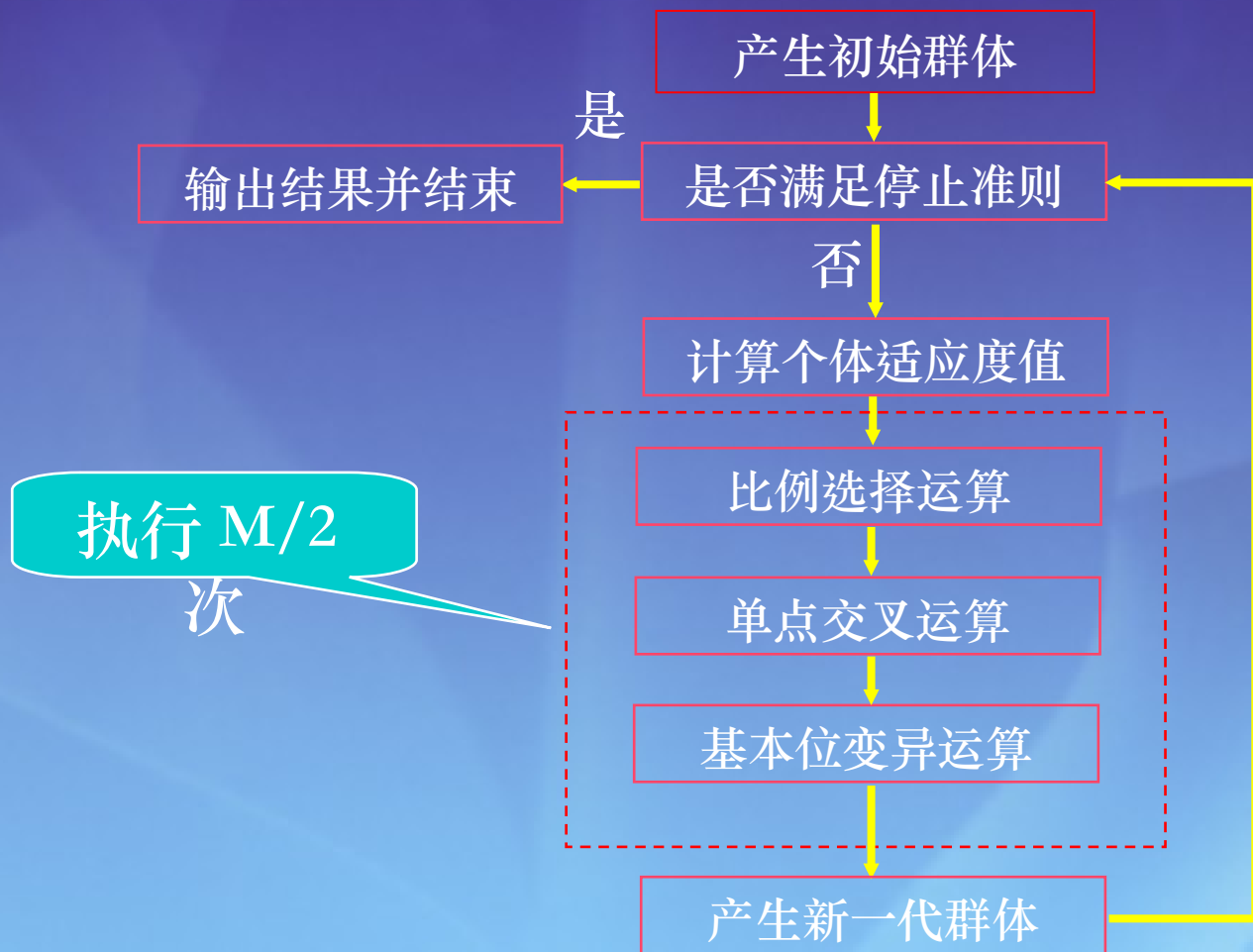
变异后：

000001110001000010000

运行参数

- (1) M : 种群规模
- (2) T : 遗传运算的终止进化代数
- (3) P_c : 交叉概率
- (4) P_m : 变异概率

SGA 的框图



3、遗传算法的特点

- (1) 群体搜索，易于并行化处理；
- (2) 不是盲目穷举，而是启发式搜索；
- (3) 适应度函数不受连续、可微等条件的约束，适用范围很广。

二、遗传算法原理

- 1、遗传算法的数学基础
- 2、遗传算法的收敛性分析
- 3、遗传算法的改进

1、遗传算法的数学基础

(1) 模式定理

(2) 积木块假设

模式

模式是指种群个体基因串中的相似样板，它用来描述基因串中某些特征位相同的结构。在二进制编码中，模式是基于三个字符集 (0,1,*) 的字符串，符号 * 代表任意字符，即 0 或者 1。

模式示例：101**

两个定义

定义 1：模式 H 中确定位置的个数称为模式 H 的阶，记作 $O(H)$ 。例如

$$O(10^{**}1)=3 \quad \circ$$

定义 2：模式 H 中第一个确定位置和最后一个确定位置之间的距离称为模式 H 的定义距，记作 $\delta(H)$ 。例如 $\delta(10^{**}1)=4 \quad \circ$

模式的阶和定义距的含义

模式阶用来反映不同模式间确定性的差异，模式阶数越高，模式的确定性就越高，所匹配的样本数就越少。在遗传操作中，即使阶数相同的模式，也会有不同的性质，而模式的定义距就反映了这种性质的差异。

模式定理

模式定理：具有低阶、短定义距以及平均适应度高于种群平均适应度的模式在子代中呈指数增长。

模式定理保证了较优的模式（遗传算法的较优解）的数目呈指数增长，为解释遗传算法机理提供了数学基础。

模式定理

从模式定理可看出，有高平均适应度、短定义距、低阶的模式，在连续的后代里获得至少以指数增长的串数目，这主要是因为选择使最好的模式有更多的复制，交叉算子不容易破坏高频率出现的、短定义长的模式，而一般突变概率又相当小，因而它对这些重要的模式几乎没有影响。

积木块假设

积木块假设：遗传算法通过短定义距、低阶以及高平均适应度的模式（积木块），在遗传操作下相互结合，最终接近全局最优解。

模式定理保证了较优模式的样本数呈指数增长，从而使遗传算法找到全局最优解的可能性存在；而积木块假设则指出了在遗传算子的作用下，能生成全局最优解。

2、遗传算法的收敛性分析

遗传算法要实现全局收敛，首先要求任意初始种群经有限步都能到达全局最优解，其次算法必须由保优操作来防止最优解的遗失。与算法收敛性有关的因素主要包括种群规模、选择操作、交叉概率和变异概率。

种群规模对收敛性的影响

通常，种群太小则不能提供足够的采样点，以致算法性能很差；种群太大，尽管可以增加优化信息，阻止早熟收敛的发生，但无疑会增加计算量，造成收敛时间太长，表现为收敛速度缓慢。

选择操作对收敛性的影响

选择操作使高适应度个体能够以更大的概率生存，从而提高了遗传算法的全局收敛性。如果在算法中采用最优保存策略，即将父代群体中最佳个体保留下来，不参加交叉和变异操作，使之直接进入下一代，最终可使遗传算法以概率 1 收敛于全局最优解。

交叉概率对收敛性的影响

交叉操作作用于个体对，产生新的个体，实质上是在解空间中进行有效搜索。交叉概率太大时，种群中个体更新很快，会造成高适应度值的个体很快被破坏掉；概率太小时，交叉操作很少进行，从而会使搜索停滞不前，造成算法的不收敛。

变异概率对收敛性的影响

变异操作是对种群模式的扰动，有利于增加种群的多样性。但是，变异概率太小则很难产生新模式，变异概率太大则会使遗传算法成为随机搜索算法。

遗传算法的本质

遗传算法本质上是对染色体模式所进行的一系列运算，即通过选择算子将当前种群中的优良模式遗传到下一代种群中，利用交叉算子进行模式重组，利用变异算子进行模式突变。通过这些遗传操作，模式逐步向较好的方向进化，最终得到问题的最优解。

3、遗传算法的改进

遗传欺骗问题：在遗传算法进化过程中，有时会产生一些超常的个体，这些个体因竞争力太突出而控制了选择运算过程，从而影响算法的全局优化性能，导致算法获得某个局部最优解。

遗传算法的改进途径

- (1) 对编码方式的改进
- (2) 对遗传算子的改进
- (3) 对控制参数的改进
- (4) 对执行策略的改进

对编码方式的改进

二进制编码优点在于编码、解码操作简单，交叉、变异等操作便于实现，缺点在于精度要求较高时，个体编码串较长，使算法的搜索空间急剧扩大，遗传算法的性能降低。格雷编码克服了二进制编码的不连续问题，浮点数编码改善了遗传算法的计算复杂性。

对遗传算子的改进

排序选择

均匀交叉

逆序变异

(1) 对群体中的所有个体按其适应度大小进行降序排序；

(2) 根据具体求解问题，设计一个概率分配表，将各个概率值按上述排列次序分配给各个个体；

(3) 以各个个体所分配到的概率值作为其遗传到下一代的概率，基于这些概率用赌盘选择法来产生下一代群体。

对遗传算子的改进

排序选择

均匀交叉

逆序变异

(1) 随机产生一个与个体编码长度相同的二进制屏蔽字 $P = W_1W_2\dots W_n$ ；

(2) 按下列规则从 A 、 B 两个父代个体中产生两个新个体 X 、 Y ：若 $W_i = 0$ ，则 X 的第 i 个基因继承 A 的对应基因， Y 的第 i 个基因继承 B 的对应基因；若 $W_i = 1$ ，则 A 、 B 的第 i 个基因相互交换，从而生成 X 、 Y 的第 i 个基因。

对遗传算子的改进

排序选择

均匀交叉

逆序变异

变异前：

3 4 8 | 7 9 6 5 | 2 1

变异后：

3 4 8 | 5 6 9 7 | 2 1

对控制参数的改进

Schaffer 建议的最优参数范围是：

$$M = 20-100 \text{ ,}$$

$$T = 100-500 \text{ ,}$$

$$P_c = 0.4-0.9 \text{ ,}$$

$$P_m = 0.001-0.01 \text{ 。}$$

对控制参数的改进

Srinivas 等人提出自适应遗传算法，即 P_c 和 P_m 能够随适应度自动改变，当种群的各个个体适应度趋于一致或趋于局部最优时，使二者增加，而当种群适应度比较分散时，使二者减小，同时对适应值高于群体平均适应值的个体，采用较低的 P_c 和 P_m ，使性能优良的个体进入下一代，而低于平均适应值的个体，采用较高的 P_c 和 P_m ，使性能较差的个体被淘汰。

对执行策略的改进

混合遗传算法

免疫遗传算法

小生境遗传算法

单亲遗传算法

并行遗传算法

混合遗传算法

混合遗传算法是将遗传算法与其他优化算法、启发式方法或领域知识相结合的一种算法，旨在综合不同算法的优势，提高算法的性能和求解质量。遗传算法在搜索过程中可能存在收敛速度慢、容易陷入局部最优等问题。混合遗传算法通过引入其他算法或方法，弥补遗传算法的不足，增强其搜索能力和收敛性。

小生境遗传算法

小生境是指在一个生态环境中，具有相似特征或行为的生物群体所形成的局部环境。在遗传算法中，将种群中的个体按照某种相似性度量方法划分为不同的小生境，每个小生境中的个体具有较高的相似性，而不同小生境中的个体具有较大的差异。通过在选择、交叉和变异等遗传操作中引入小生境技术，使得每个小生境中的优秀个体能够得到充分的进化，同时避免某个小生境中的个体过度繁殖而占据整个种群，从而维持种群的多样性，提高算法的全局搜索能力。

三、遗传算法的应用领域

- (1) 组合优化
- (2) 函数优化
- (3) 自动控制
- (4) 生产调度
- (5) 图像处理
- (6) 机器学习
- (7) 人工生命
- (8) 数据挖掘

遗传算法应用于组合优化

随着问题规模的增大，组合优化问题的搜索空间也急剧扩大，有时在计算机上用枚举法很难甚至不可能求出其最优解。实践证明，遗传算法已经在求解旅行商问题、背包问题、装箱问题、布局优化、网络路由等组合优化问题上取得了成功的应用。

遗传算法应用于旅行商问题

问题：旅行商问题是一个经典的组合优化问题，目标是找到一条遍历所有城市且每个城市仅访问一次，最后回到起始城市的最短路径。

编码：常用的是顺序编码，即直接用城市编号的排列来表示一个可能的路径。例如，有 5 个城市分别编号为 1、2、3、4、5，那么一个染色体可以是 [3, 1, 4, 5, 2]，表示旅行商依次访问城市 3、1、4、5、2。

适应度函数：通常将路径的总长度的倒数作为适应度值。

选择算子：轮盘赌选择或者排序选择

交叉算子：顺序交叉，随机选择两个交叉点，将父代 1 中两个交叉点之间的基因片段直接复制到子代中，然后从父代 2 中按顺序选取子代中尚未出现的基因，依次填入子代的剩余位置。

变异算子：交换变异，随机选择个体中的两个基因位置，交换这两个位置上的基因。

遗传算法应用于背包问题

问题：背包问题是一个经典的组合优化问题，给定一组物品，每个物品都有自己的重量和价值，同时给定一个背包的容量，要求在有限的背包容量下选择物品，使得装入背包的物品总价值最大。

编码：常用的是二进制编码，用 0 和 1 表示物品是否放入背包，例如长度为 n 的二进制串，第 i 位为 1 表示第 i 个物品放入背包，为 0 则表示不放入。

适应度函数：将放入背包的物品总价值作为适应度值，但要注意满足背包容量的限制。对于超出背包容量的染色体，可以采用惩罚函数的方式降低其适应度。

选择算子：轮盘赌选择或者排序选择。

交叉算子：均匀交叉。

变异算子：基本位变异。

遗传算法应用于 PID 控制优化

问题： PID 控制器是工业控制中应用最广泛的一种控制器，它通过比例（P）、积分（I）、微分（D）三个参数的组合来对系统进行控制。遗传算法可以用于优化 PID 控制器的参数，以提高控制系统的性能。

编码： 将 PID 参数 K_p 、 K_i 和 K_d 编码为染色体。常用的编码方式有二进制编码和实数编码。

适应度函数： 度适应度函数用于评估每个个体（PID 参数组合）的优劣。常见的适应度函数设计方法是基于控制系统的性能指标，如积分时间绝对误差指标（ITAE）： $J = \int t|e(t)|dt$ 。

选择算子： 轮盘赌选择或者排序选择。

交叉算子： 均匀交叉。

变异算子： 基本位变异。