



人工智能导论

主讲教师：朱艳菊
信息科学与技术学院
电子信息工程系



- 如何**模仿生物**建立功能强大的算法，并运用于复杂的优化问题成为研究热点。
- 进化计算包括：

遗传算法

遗传编程

进化策略

进化编程

- 4.1 遗传算法概述
- 4.2 遗传算法的基本机理
- 4.3 遗传算法的求解步骤
- 4.4 遗传算法的求解举例

4.1 遗传算法概述

- **遗传算法**（genetic algorithms, GA）：一类借鉴生物界自然选择和自然遗传机制的随机搜索算法,非常适用于处理传统搜索方法难以解决的复杂和非线性优化问题。
- 遗传算法可广泛应用于组合优化、机器学习、自适应控制、规划设计和人工生命等领域。

4.1 遗传算法概述

- 生物种群的生存过程普遍遵循达尔文的进化论。
- 适者生存，优胜劣汰



4.1 遗传算法概述

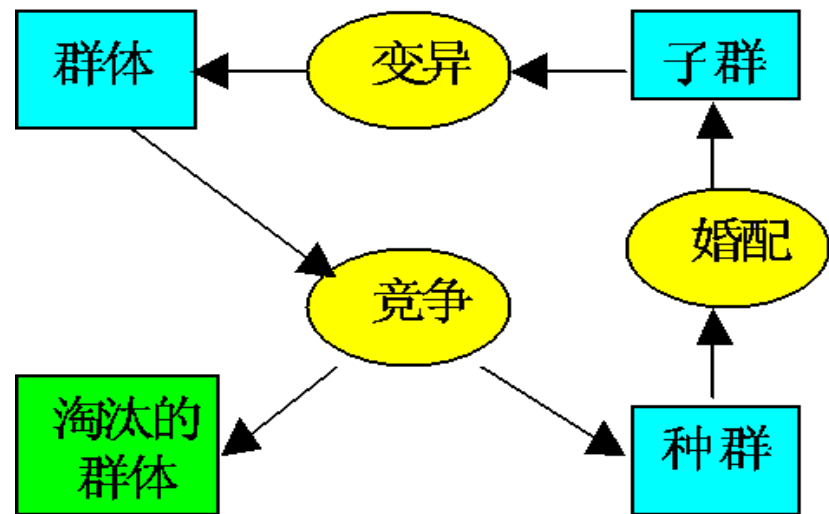
- 每一物种在所处环境中不断进化，适应环境得以生存。
- 物种个体基本特征被后代继承，后代产生异于父代的新变化，适应环境个体特征得以保留。

染色体(chromosome)：生物的遗传物质的主要载体。

基因(gene)：扩展生物性状的遗传物质的功能单元和结构单位。

基因座 (locus)：染色体中基因的位置。

等位基因 (alleles)：基因所取的值。



生物进化的基本过程

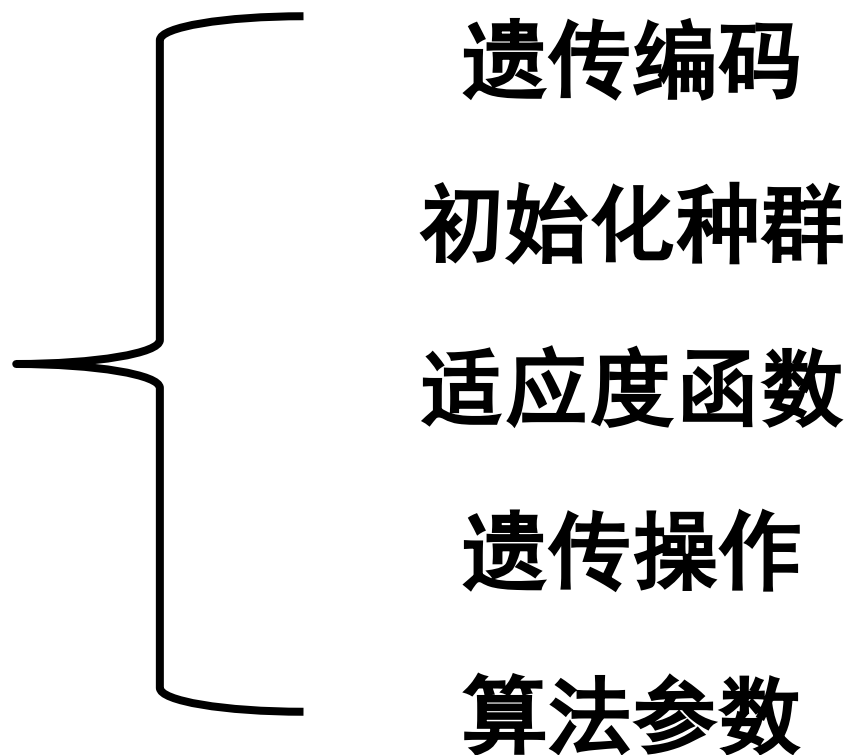
4.1 遗传算法概述

- **遗传算法的基本思想：**

模拟生物遗传学和自然选择机理，用人工方式构造优化搜索算法。

- **遗传算法是一个迭代过程，每次迭代保留一组候选解，按优劣指标排序，从中选出一些解，利用遗传算子进行运算产生新一代。重复该过程，直到满足指定收敛要求为止。**

● 遗传算法基本组成：





遗传算法的基本概念

- **个体**：一个数据结构，用来描述基本遗传结构。
例如，用0、1串表示个体，串为染色体，0或1为等位基因
- **群体**：个体组成的集合。
- **适应性**：个体有对应适应值。优化问题中，适应值来自估计函数。
- **遗传操作**：作用于群体产生新群体。
标准遗传操作：选择、交叉、变异。

遗传算法的基本组成：遗传编码

- 编码：位串编码表示的过程。
- 解码或译码：位串编码表示变换为原问题结构的过程。
- 遗传算法常用编码：
二进制编码、浮点数编码、格雷码、符号编码、多参数编码

4.2 遗传算法的基本机理



遗传编码：二进制编码/

- 参数取值范围 $[A, B]$, $A < B$ 。长度 l 的二进制编码串表示该参数
 - $[A, B]$ 等分成 $2^l - 1$ 个子部分, 等分长度为 δ , 产生 2^l 种不同编码

- 参数**编码**对应关系:

$$\begin{array}{llll} 00000000 \cdots 00000000 = 0 & \rightarrow & A \\ 00000000 \cdots 00000001 = 1 & \rightarrow & A + \delta, \quad \delta = \frac{B-A}{2^l-1} \\ \cdots & \cdots & \cdots & \cdots \\ 11111111 \cdots 11111111 = 2^l - 1 & \rightarrow & B \end{array}$$

- 设某二进制编码 X : $x_1 x_{l-1} x_{l-2} \cdots x_2 x_1$, **解码**公式:

$$x = A + \frac{B-A}{2^l-1} \sum_{i=1}^l x_i 2^{i-1}$$

例如, 0001001解码: $x = A + ((B-A)/(2^l-1)) \times (1 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + \cdots)$

4.2 遗传算法的基本机理



石家庄铁道大学
Shijiazhuang Tiedao University

优点：

类似于生物染色体的组成，算法易于用生物遗传理论解释，遗传操作如交叉、变异等易实现；算法处理的模式数最多。

缺点：

- ① 要先给出求解的精度。
- ② 求解高维优化问题的二进制编码串长，算法的搜索效率低。



遗传算法的基本组成：初始化种群

1. 初始种群的产生

(1) 根据问题固有知识，把握最优解所占空间在整个问题空间中的分布范围，然后，在此分布范围内设定初始群体。

(2) 随机产生一定数目的个体，从中挑选最好的个体加到初始群体中。这种过程不断迭代，直到初始群体中个体数目达到了预先确定的规模。

2. 种群规模的确定

- 群体规模太小，遗传算法的优化性能不太好，易陷入局部最优解。
- 群体规模太大，计算复杂。



遗传算法的基本组成：适应度函数

- 对问题中染色体进行度量的函数，称为适应度函数。

- 如何确定适应度函数？

- (1) 能有效反应染色体与问题最优染色体之间的差距

- (2) 取值大小与求解问题对象的意义有很大关系

- (3) 适应度函数设计原则：

- 单值、连续、非负、最大化；

- 合理、一致性，反应对应解的优劣程度；

- 计算量小，减小计算时间和空间复杂性；

- 通用性强，无须改变适应度函数的参数

将目标函数映射成适应度函数的方法

- 若目标函数为**最大化**问题，则 $Fit(f(x)) = f(x)$
- 若目标函数为**最小化**问题，则 $Fit(f(x)) = \frac{1}{f(x)}$



将目标函数转换为求最大值的形式, 且保证函数值非负!

- 若目标函数为**最大化**问题，则

$$Fit(f(x)) = \begin{cases} f(x) - C_{\min} & f(x) > C_{\min} \\ 0 & \text{其他情况} \end{cases}$$

- 若目标函数为**最小化**问题，则

$$Fit(f(x)) = \begin{cases} C_{\max} - f(x) & f(x) < C_{\max} \\ 0 & \text{其他情况} \end{cases}$$

遗传算法的基本组成：遗传操作

- **选择：**从种群中选择生命力强的染色体产生新种群
- **交叉：**两个个体遗传物质进行交换产生新个体
- **变异：**个体中遗传物质被改变

改进遗传算法大量扩充遗传操作已达到更高效率。

4.2 遗传算法的基本机理



石家庄铁道大学
Shijiazhuang Tiedao University

遗传操作：选择

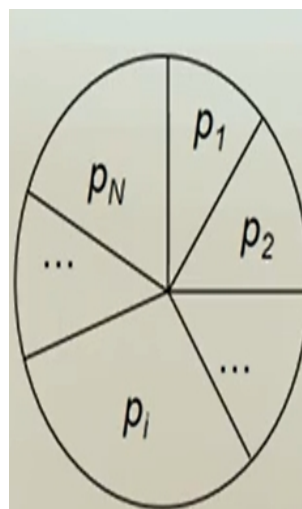
- 选择操作：从种群选择生命力强的染色体
 - 适应度函数值决定下一代被淘汰或被遗传
- 转盘赌轮选择机制

产生后代的能力：适应度值所占份额。

种群第*i*个染色体的适应度值

$$p_{si} = \frac{f_i}{\sum_{i=1}^M f_i}$$

种群适应度值之总和



赌轮选择原理

- 根据 p_i , $i=1\sim N$, 把圆盘分成 N 份
- 假想转动圆盘。若参照点落入第 i 个扇形内, 则选择个体 i 。

选择个体方法

转盘赌选择 (roulette wheel selection)

- ▶ 按个体的选择概率产生一个轮盘，轮盘每个区的角度与个体的选择概率成比例。
- ▶ 产生一个随机数，它落入转盘的哪个区域就选择相应的个体交叉。

个体	1	2	3	4	5	6	7	8	9	10	11
适应度	2.0	1.8	1.6	1.4	1.2	1.0	0.8	0.6	0.4	0.2	0.1
选择概率	0.18	0.16	0.15	0.13	0.11	0.09	0.07	0.06	0.03	0.02	0.0
累积概率	0.18	0.34	0.49	0.62	0.73	0.82	0.89	0.95	0.98	1.00	1.00

第1轮产生一个随机数：**0.81**

第2轮产生一个随机数：**0.32**

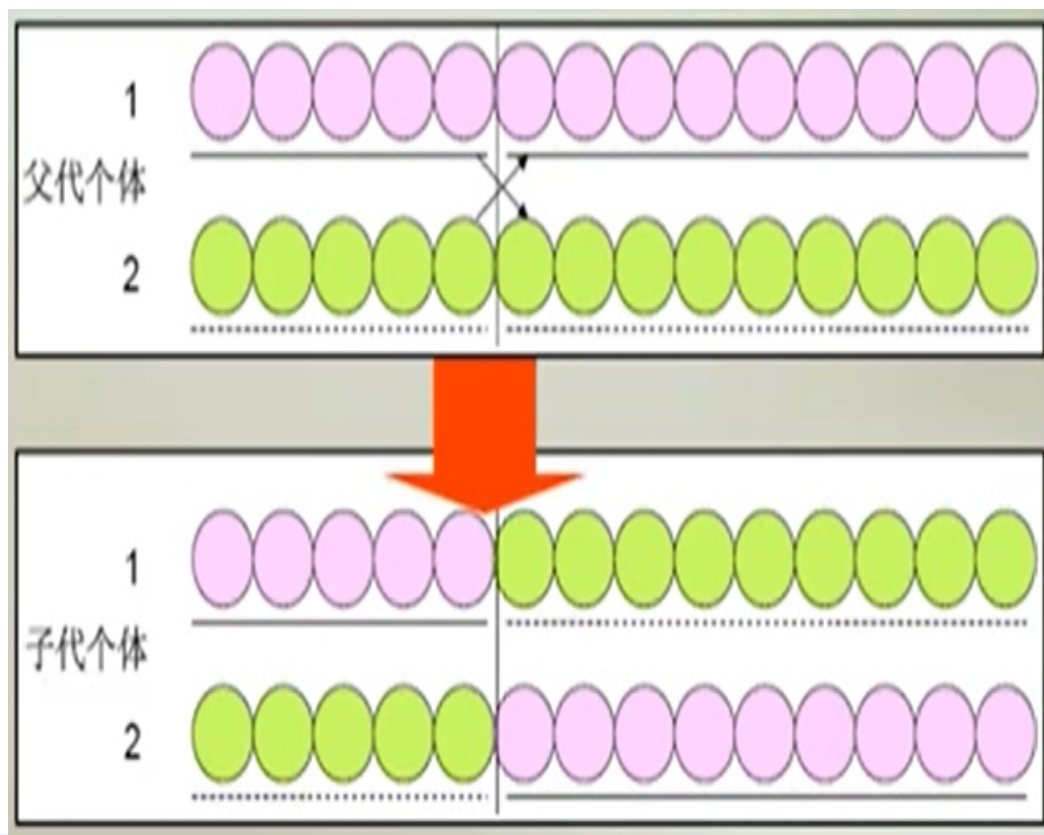
4.2 遗传算法的基本机理



遗传操作：交叉

- 交叉操作：两个个体遗传物进行交换产生新个体
- 交叉后个体性能不佳，则在其后复制中被淘汰
- 交叉操作的具体步骤：

- 1：从交配池中随机取出要交配的一对个体。
- 2：根据位串长度 L ，随机选取 $[1, L-1]$ 中整数 k 作为交叉点。
- 3：根据交叉概率实施交叉，在交叉点处交换各自内容，形成新一对个体。



4.2 遗传算法的基本机理



遗传操作：变异

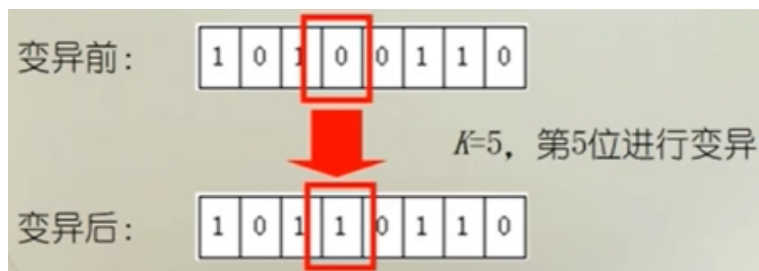
- 变异操作：个体中遗传物质被改变
- 作用

选择和交叉结合，保证算法有效性，并具有局部随机搜索能力；
使算法保持群多样性，防止非成熟收敛

变异基本原则：随机产生一个实数 p , $0 \leq p \leq 1$ 。如果 p 大于变异概率阈值 p_m ，则对该个体进行变异。

变异操作：改变数码串某个位置上的数码。

简单变异操作将0与1互换：0变异为1，1变异为0。



- (1) **位点变异**：群体中的个体码串，随机挑选一个或多个基因座，并对这些基因座的基因值以变异概率作变动。
- (2) **逆转变异**：在个体码串中随机选择两点（逆转点），然后将两点之间的基因值以逆向排序插入到原位置中。
- (3) **插入变异**：在个体码串中随机选择一个码，然后将此码插入随机选择的插入点中间。
- (4) **互换变异**：随机选取染色体的两个基因进行简单互换。
- (5) **移动变异**：随机选取一个基因，向左或者向右移动一个随机位数。

4.3 遗传算法的求解步骤



遗传算法的基本组成：算法参数

主要参数：

- **种群的规模：群体中个体的数量**
- **算法执行的最大代数数目：种群进化迭代次数**

次要参数：

- **交叉概率：两两个体是否需要需要进行交叉的概率**
- **变异概率：任一个体是否需要需要进行变异的规律**
- **影响遗传算法行为和性能的关键参数，直接影响算法收敛性**
- **针对不同优化问题，需要反复试验确定这些概率**

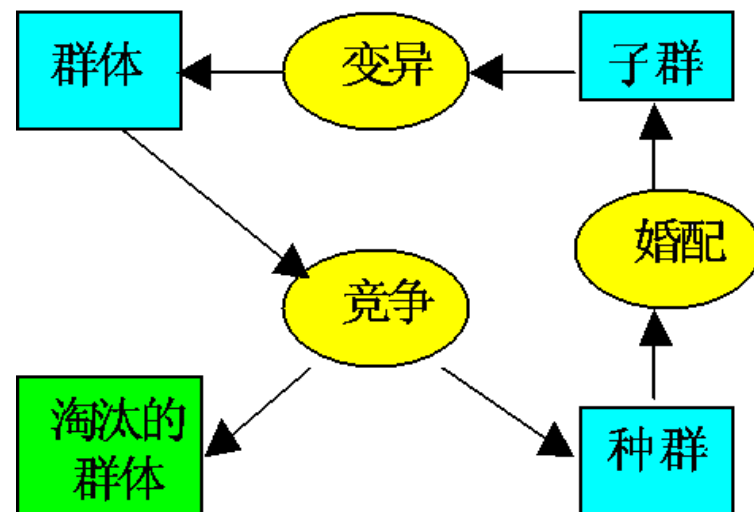
4.3 遗传算法的求解步骤

- 遗传算法类似于自然进化

与自然界相似，是适应性好的染色体有更多繁殖机会

- 求解步骤：

- (1) 随机产生若干求解问题的编码，形成初始群体；
- (2) 用适应度函数评价个体；
- (3) 选择高适应值个体参加遗传操作，形成下一代种群；
- (4) 对新种群进行下一轮进化





遗传算法的特点

- 通过达尔文适者生存理论、自然选择/遗传/变异等操作，模拟自然进化过程寻找所求问题解答。
- 特点：
 - (1) 对参数集合的编码，而非针对参数本身进行进化；
 - (2) 从问题解的编码组开始，而非从单个解开始搜索；
 - (3) 利用目标函数的适应度，而非利用倒数或其他辅助信息指导搜索；
 - (4) 利用选择、交叉、变异等算子，而非利用确定性规则进行随机操作

4.3 遗传算法的求解步骤

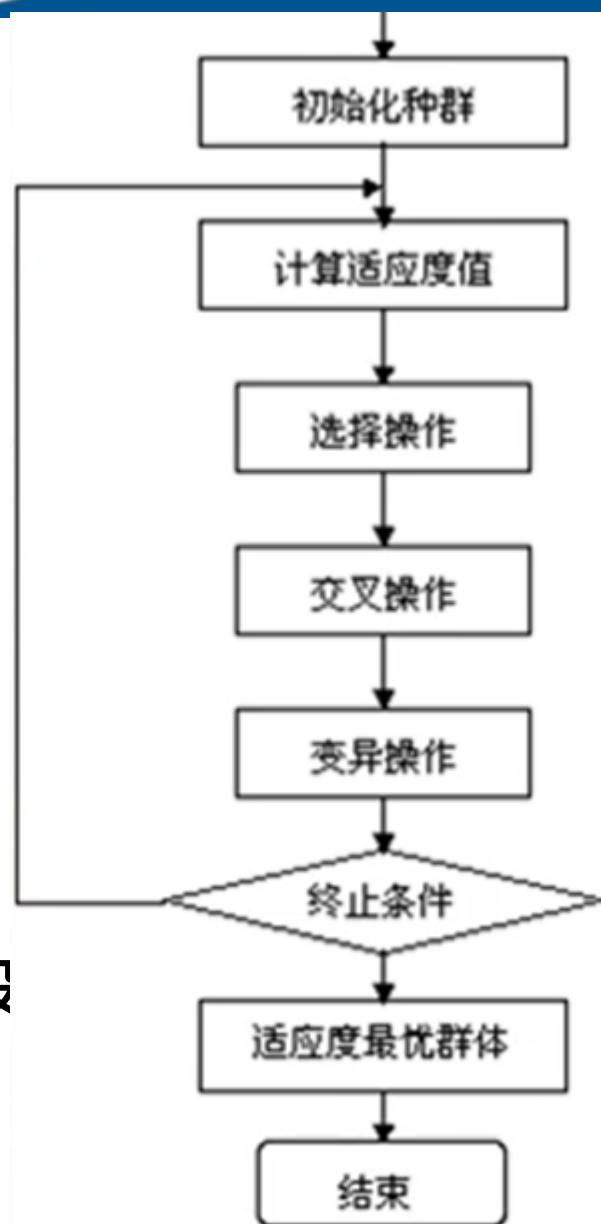


遗传算法的求解步骤

- 1: 初始化群体;
- 2: 计算个体适应度值;
- 3: 按适应度值选择下一代个体;
- 4: 按交叉概率进行交叉操作;
- 5: 按变异概率进行变异操作;
- 6: 未满足停止条件, 则转2;
否则转7;
- 7: 输出最优染色体作为问题最优解

终止条件:

- (1) 完成预先给定进化代数;
- (2) 连续若干代没有改进



4.3 遗传算法的求解步骤



步骤1：随机产生一定长度特征字符串组成初始种群；

步骤2：对字符串种群迭代执行子步骤(1)和(2)；

(1) 计算个体适应值；

(2) 应用复制、交叉和变异等算子产生下一代种群

步骤3：后代最好个体指定为算法执行结果，表示问题解。

4.3 遗传算法的求解步骤



生物遗传概念	遗产算法中的应用
适者生存	目标值比较大的解被选择的可能性大
个体 (Individual)	解
染色体 (Chromosome)	解的编码 (字符串、向量等)
基因 (Gene)	解的编码中每一分量
适应性 (Fitness)	适应度函数值
群体 (Population)	根据适应度值选定的一组解 (解的个数为群体的规模)
婚配 (Marry)	交叉 (Crossover) 选择两个染色体进行交叉产生一组新的染色体的过程
变异 (Mutation)	编码的某一分量发生变化的过程

● 遗传算法的改进算法

- 双倍体遗传算法
 - 采用显性和隐性两个染色体同时进行进化
- 双种群遗传算法
 - 使用多个种群同时进化，交换种群之间最好个体所携带的遗传信息
- 自适应遗传算法
 - 交叉概率和变异概率可以随适应度自动改变
- 免疫遗传算法
 - 生物免疫机制引入标准遗传算法，提高算法全局搜索能力，克服早熟收敛问题
- 小生境遗传算法
 - 小生境技术引入遗传算法原始结构，改善遗传算法处理多峰优化问题的能力
- 混沌遗传算法
 - 混沌理论与遗传算法相结合的优化算法，使用精英预留和混合排序选择策略，设计自适应交叉、变异算子来提高改进算法搜索性

4.3 遗传算法的求解步骤



- 讨论：遗传算法需要解决的问题
- 群体如何设定？
- 算法如何终止？

4.3 遗传算法的求解步骤



1、群体如何设定

关键问题：群体规模，即如何设定群体包括的个体数目。

• 初始群体的设定

- 把握最优解所占空间在整个问题空间中分布范围，在此范围内设定初始群体。
- 随机生成一定数目个体，从中挑出最好个体加入初始群体。过程不断迭代，直到个体数达到预先确定规模。

• 进化中各代规模维持

- **群体规模太大**：从计算效率，群体越大，计算量也增加，**影响算法效能。**
- **群体规模太小**：算法搜索空间分布范围有限，搜索可能停止在未成熟阶段，引起**过早收敛。**

2、算法如何终止

- 算法控制转移规则为**随机**，演化过程无法确定个体在解空间的位置，无法用传统方法判断算法收敛与否来终止算法。
- 常用终止算法：
 - 预先规定最大演化代数；
 - 连续多代后，解适应值无明显改进；
 - 达到明确解目标

4.4 遗传算法的求解举例



石家庄铁道大学
Shijiazhuang Tiedao University

4.4 遗传算法的求解举例



例：用遗传算法求解一元函数最大值的优化问题。

$$f(x)=x\sin(10\pi \cdot x)+1.0, \quad x \in [-1, 2]$$

Step1: 编码

- 二进制编码，设求解精度到6位小数。
- 变量 x 区间长度为3，区间 $[-1, 2]$ 分为 3×10^6 个等长区间。
- $2097152 = 2^{21} < 3 \times 10^6 \leq 2^{22} = 4194304$
二进制串长至少需要22位。

4.4 遗传算法的求解举例



- 区间 $[-1, 2]$ 内实数 x 和二进制串 $(b_{21}b_{20}\cdots b_0)$ 之间对应关系:

- (a) 二进制串 $(b_{21}b_{20}\cdots b_0)$ 转化为十进制数:

$$((b_{21}b_{20}\cdots b_0))_2 = \left(\sum_{i=0}^{21} b_i \cdot 2^i \right)_{10} = x'$$

- (b) 找到对应实数 x :

$$x = -1.0 + x' \cdot \frac{3}{2^{22} - 1}$$

例如, 二进制串 $(100010111011010101000111)$ 表示实数 x =?

因为: $x' = (100010111011010101000111)_2 = 2288967$

$$x = -1.0 + 228867 \times (3/4194303) = 0.637197$$

- $(000000000000000000000000)$ 和 $(111111111111111111111111)$ 分别表示区间边界: -1.0 和 2.0 。

4.4 遗传算法的求解举例



Step2: 产生种群初始化

- 产生一定数目个体组成种群。
- 个体由串长为22、随机产生的二进制串组成染色体

例：设3个初始个体：

$s1 = \langle 1000101110110101000111 \rangle$

$s2 = \langle 0000001110000000100000 \rangle$

$s3 = \langle 1110000000111111000101 \rangle$

4.4 遗传算法的求解举例



Step3: 计算适应度

- 为种群每个字符串指定一个适应值。
- 本例直接引用目标函数作为适应度函数，即
$$f(x) = x \sin(10\pi \cdot x) + 1.0$$

编号	个体值	x	适应值
$s1$	1000101110110101000111	0.637179	1.586345
$s2$	0000001110000000100000	-0.958973	0.078878
$s3$	1110000000111111000101	1.627888	2.25065

4.4 遗传算法的求解举例



Step4:遗传操作

- 转盘赌轮方式选择子个体，生成随机数为0.35，0.72，则选中个体为 s_2 和 s_3 。

$s_2 = \langle 00000 \mid 01110000000100000 \rangle$

$s_3 = \langle 11100 \mid 00000111111000101 \rangle$

- s_2 和 s_3 进行交叉操作，随机选择交叉点。例如第5位与第6位之间位置，交叉后产生新子个体：

$s_2' = \langle 00000 \mid 00000111111000101 \rangle$

$s_3' = \langle 11100 \mid 01110000000100000 \rangle$

- 两个子个体的适应值：

$$f(s_2') = f(-0.998113) = 0.940865$$

$$f(s_3') = f(1.666028) = 2.459245$$

- 交叉后个体 s_3' 的适应值比父个体适应值高。

4.4 遗传算法的求解举例



- s_3 进行变异操作，随机选择变异点

例如，第5位，变异后产生新子个体：

$$s_3' = \langle 1110100000111111000101 \rangle$$

$$\text{子个体适应值: } f(s_3') = f(1.721638) = -0.082257$$

\therefore 变异后个体 s_3' 的适应值比父个体适应值低

例如，第10位，变异后产生新子个体：

$$s_3'' = \langle 1110100001111111000101 \rangle$$

$$\text{子个体适应值: } f(s_3'') = f(1.630818) = 2.343555$$

\therefore 变异后个体 s_3'' 的适应值比父个体适应值高

4.4 遗传算法的求解举例



Step5: 模拟结果

设种群规模=50，交叉概率 $p_c=0.25$ ，变异概率 $p_m=0.01$

代数	个体的二进制串	x	适应值
1	1111 0001 1010 0001 1100 00	1.831 624	2.534 806
4	1111 0010 1000 1101 1000 00	1.842 416	2.790 362
7	1111 0011 1001 1101 0101 10	1.854 860	2.833 280
11	1111 0011 1001 1101 0101 10	1.854 860	2.833 286
17	1111 0010 1111 1101 0101 10	1.847 536	2.842 004
18	1111 0010 1111 1101 1100 00	1.847 554	3.842 102
34	1111 0011 0111 1011 0000 11	1.853 290	2.843 402
40	1111 0011 0001 0001 0010 11	1.848 443	2.846 232
54	1111 0011 0001 0110 1100 00	1.848 699	2.847 155
71	1111 0011 0100 0110 1100 01	1.850 897	2.850 162
89	1111 0011 0011 1111 0010 11	1.850 549	2.850 274
...
150	1111 0011 0011 1111 0010 11	1.850 549	2.850 274

图：模拟世代种群中最佳个体的演变

按照标准遗传算法，运行第150代时获得最佳个体：

$$S_{max} = \langle 1111001100111111001011 \rangle$$

$$x_{max} = 1.850549, \quad f(x_{max}) = 2.850274$$