

第五章 进化计算

- 主要内容
- 进化计算
- 人工生命

生命科学是系统地阐述与生命特性有关的重大课题的科学。

概述

- **学科交叉**是当前研究领域的一个重要特征
- **信息科学与生物科学的相互交叉、相互渗透和相互促进**是现代科学技术发展的一个显著特点。
- **计算智能**是学科交叉研究过程中出现的一个重要研究方向。

什么是计算智能

- 计算智能,也有人称之为“软计算”,虽然至今没有一个统一的定义,但我们可以这样来概括它。

● **计算智能**就是受自然界（生物界）规律的启迪,根据其原理,模仿设计求解问题的算法。

- 目前这方面的内容很多,如:**神经网络、进化计算、人工生命、群集智能技术、人工免疫系统技术**等领域,它的研究和发展正是反映了当代科学技术多学科交叉与集成的重要发展趋势。

计算智能与人工智能的区别和关系

- 计算智能：**生物智能的计算模拟**，是一种智力方式的低层认知，它与人工智能的区别只是认知层次从**中层**下降至**低层**而已。中层系统含有知识，低层系统则没有。

计算智能与人工智能的区别和关系

- 当一个系统只涉及数值（低层）数据，含有模式识别部分，不应用人工智能意义上的知识，而且能够呈现出：
 - （1）计算适应性；
 - （2）计算容错性；
 - （3）接近人的速度；
 - （4）误差率与人相近，
- 则该系统就是计算智能系统。
- 当一个智能计算系统以非数值方式加上知识，即成为人工智能系统。

进化计算

- 进化计算是一类模拟**生物进化过程与机制**求解问题的自组织、自适应技术。
- 进化计算起源于20世纪50年代末，成熟于20世纪80年代，目前主要被应用于工程控制、机器学习、函数优化、任务调度等领域。

进化理论

- **Darwin进化论**最重要的是**适者生存原理**。
 - 它认为每一物种在发展中越来越适应环境。物种每个个体的基本特征由后代所继承，但后代又会产生一些异于父代的新变化。在环境变化时，只有那些能适应环境的个体特征方能保留下来。
- **Mendel遗传学说**最重要的是**基因遗传原理**。
 - 它认为**遗传**以密码方式存在细胞中，并以**基因形式包含在染色体内**。每个基因有特殊的位置并控制某种特殊性质；所以，每个基因产生的个体对环境具有某种适应性。**基因变异**和**基因杂交**可产生更适应于环境的后代。**经过存优去劣的自然淘汰，适应性高的基因结构得以保存下来。**

进化算法

- 依照达尔文的自然选择和孟德尔的遗传变异理论，生物的进化是通过**繁殖、变异、竞争、选择**来实现的，进化算法就是建立在上述生物模型基础上的一种随机搜索技术

进化算法的起源及分类

从上世纪四十年代以来，生物模拟就开始逐步构成了计算科学的一个重要组成部分。几十年来的研究与应用已经清楚地表明：虽然模拟自然进化搜索过程的一些模型还只是自然界生物体的粗糙简化，但是已经可以产生非常鲁棒的计算方法。

进化算法（Evolutionary Algorithm—EA）就是基于这种思想发展起来的，目前研究的进化算法主要有三种典型的算法：**遗传算法、进化规划和进化策略**。这三种算法是彼此独立发展起来的。

进化算法的分类

●遗传算法：

美国Michigan大学的Holland J. H.教授不仅对以前的学者们提出的遗传概念进行了总结与推广，而且给出了简明、清晰的算法描述，由此形成了目前人们一般意义上所说的“**遗传算法**”。

●进化规划：

最早由美国的Fogel L. J., Owens M. J. 和Walsh M. J. 提出，之后由Fogel D. B. 进行了完善。

●进化策略

由德国的Rechenberg I. 和Schwefel H. P. 建立的。

进化算法的特点

种群搜索策略和种群中**个体之间的信息交换**是进化算法的两大特点。它们的优越性主要表现在：

1. **进化算法在搜索过程中不容易陷入局部最优**，即使在所定义的适应度函数是不连续的、非规则的或有噪声的情况下，它们也能以很大的概率找到全局最优解；
2. 由于它们固有的**并行性**，进化算法非常适合于并行机；再次，进化算法采用进化机制来表现复杂的现象，能够快速可靠地解决非常困难的问题。
3. 由于它们**容易介入**到已有的模型中并且**具有可扩展性**，以及**易于同别的技术混合**等因素，因而进化算法目前已经在最优化、机器学习和并行处理等领域得到越来越**广泛的应用**。

遗传算法

- 遗传算法 (Genetic Algorithm, GA) 是一类借鉴生物界的进化规律 (**适者生存, 优胜劣汰**遗传机制) 演化而来的随机化搜索方法。它是由美国的 J. Holland 教授 1975 年首先提出, 已被人们广泛地应用于组合优化、机器学习、信号处理、自适应控制和人工生命等领域。它是现代智能计算中的关键技术之一。

遗传算法发展历史

- **1965年**，**Holland**首次提出了人工遗传操作的重要性，并把这些应用于自然系统和人工系统中。
- 1967年，Bagley在他的论文中首次提出了遗传算法这一术语，并讨论了遗传算法在自动博弈中的应用。
- 1970年，Cavicchio把遗传算法应用于模式识别中。第一个把遗传算法应用于函数优化的是Hollstien。
- **1975年**是遗传算法研究的历史上十分重要的一年。这一年，**Holland**出版了他的著名专著《**自然系统和人工系统的适应性**》该书系统地阐述了遗传算法的基本理论和方法，并提出了对遗传算法的理论研究和发​​展极为重要的模式理论（schemata theory），该理论首次确认了结构重组遗传操作对于获得隐并行性的重要性。

- 同年，DeJong完成了他的重要论文《遗传自适应系统的行为分析》。他在该论文中所做的研究工作可看作是遗传算法发展过程中的一个里程碑，这是因为他把Holland的模式理论与他的计算使用结合起来。
- 1989 Goldberg出版了《搜索、优化和机器学习中的遗传算法》对遗传算法从理论上、方法上和应用上作了系统的总结。



遗传算法的基本思想

● 基本思想

- 遗传算法把问题的解表示成“染色体”，在算法中即是以一定方式编码的串。并且，在执行遗传算法之前，给出一群“染色体”，也即假设解（候选解）。然后，把这些假设解置于问题的“环境”中，并按适者生存的原则，从中选择出较适应环境的“染色体”进行复制，再通过交叉，变异过程产生更适应环境的新一代“染色体”群。这样，一代一代地进化，最后就会收敛到最适应环境的一个“染色体”上，它就是问题的最优解。

遗传算法的框图

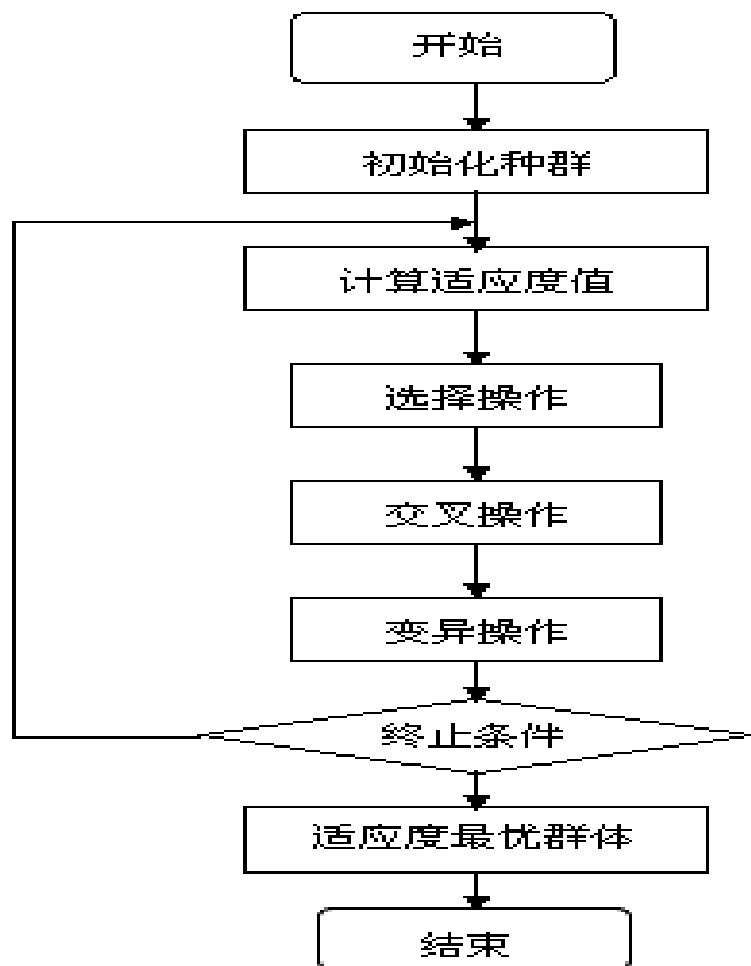


图5.2 简单遗传算法框图

1. 初始化群体;
2. 计算群体上每个个体的适应度值
3. 按由个体适应度值所决定的某个规则选择将进入下一代的个体;
4. 按概率 P_c 进行交叉操作;
5. 按概率 P_c 进行突变操作;
6. 若没有满足某种停止条件, 则转第2步; 否则进入下一步;
7. 输出群体中适应度值最优的染色体作为问题的满意解或最优解。

一些常用的基本概念

- **串**(String)
 - 它是个体(Individual)的形式，在算法中为二进制串或者其它编码方式的串，并且对应于遗传学中的**染色体**(Chromosome)。
- **种群**(Population)
 - 个体的集合称为种群，串是种群的元素
- **种群规模**(Population Size)
 - 在种群中个体的数量称为种群的规模。
- **基因**(Gene)
 - 基因是串中的元素，基因用于表示个体的特征
- **适应度**(Fitness)
 - 表示某一个体对于环境的适应程度

遗传算法的基本机理

- 一般的遗传算法由**四个部分组成**：
编码机制、适应度函数、控制参数、遗传算子
- **编码机制(encoding mechanism)**

用遗传算法解决问题时，首先要对待解决问题的模型结构和参数进行编码，一般用字符串表示。

编码机制是GA的基础

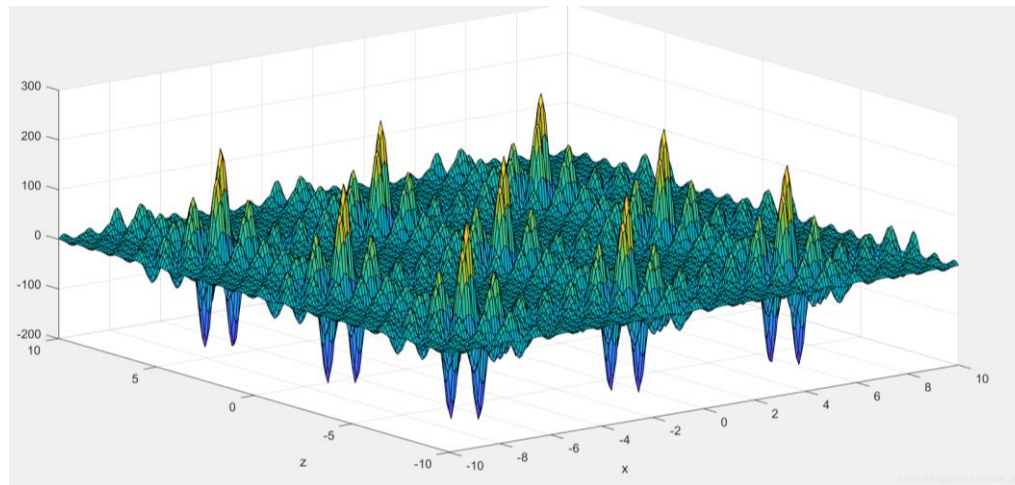
GA不是对研究对象直接进行讨论,而是通过某种编码机制把对象统一赋予由特定符号(字母)按一定顺序排成的串(string)。正如研究生物遗传,是从染色体着手,染色体则是由基因排成的串。

遗传算法的基本机理

● 适应度函数

优胜劣败是自然进化的原则。优、劣要有标准。在GA中，用适应度函数描述每一个体的适应程度。对优化问题，适应度函数与目标函数直接相关。

引进适应度函数的目的在于可根据其适应度对个体进行评估比较，定出优劣程度。



遗传算法的基本机理

- 在遗传算法的执行过程中,每一代有许多不同的个体(染色体)同时存在。这些染色体中哪个保留(生存)、哪个淘汰(死亡),是根据它们对环境的适应能力来决定的,适应性强的有更多的机会保留下来。
- 适应性强弱是通过计算**适应度函数 $f(x)$** 的值来判别的,这个值称为**适应值**。适应度函数 $f(x)$ 的构成与目标函数有密切关系。

遗传算法的基本机理

● 算法参数

在GA的实际操作时,需适当确定某些参数的值以提高选优的效果。

这些参数包含:

- 字符串所含字符的个数,即**串长**。这一长度为常数,即为定长,记为 L 。
- 每一代种群的大小,即所包含字符串的个数,也称**种群规模**,记为 n 。
- **交叉概率** (crossover rate),即施行交叉算子的概率,记为 P_c 。
- **变异概率** (mutation rate),即施行变异算子的概率,记为 P_m 。

遗传算法的基本机理

- 在GA中, **种群规模** n 太小时难以求出最优解, 太大则增长收敛时间。一般 $n=30-160$ 。
- **交叉概率** P_c 太小时难以向前搜索, 太大则容易破坏高适应值的结构。一般取 $P_c=0.25-0.75$ 。
- **变异概率** P_m 太小时难以产生新的基因结构, 太大使遗传算法成了单纯的随机搜索。一般取 $P_m=0.01-0.2$ 。

系统参数对算法的收敛速度及结果有很大的影响, 应视具体问题选取不同的值

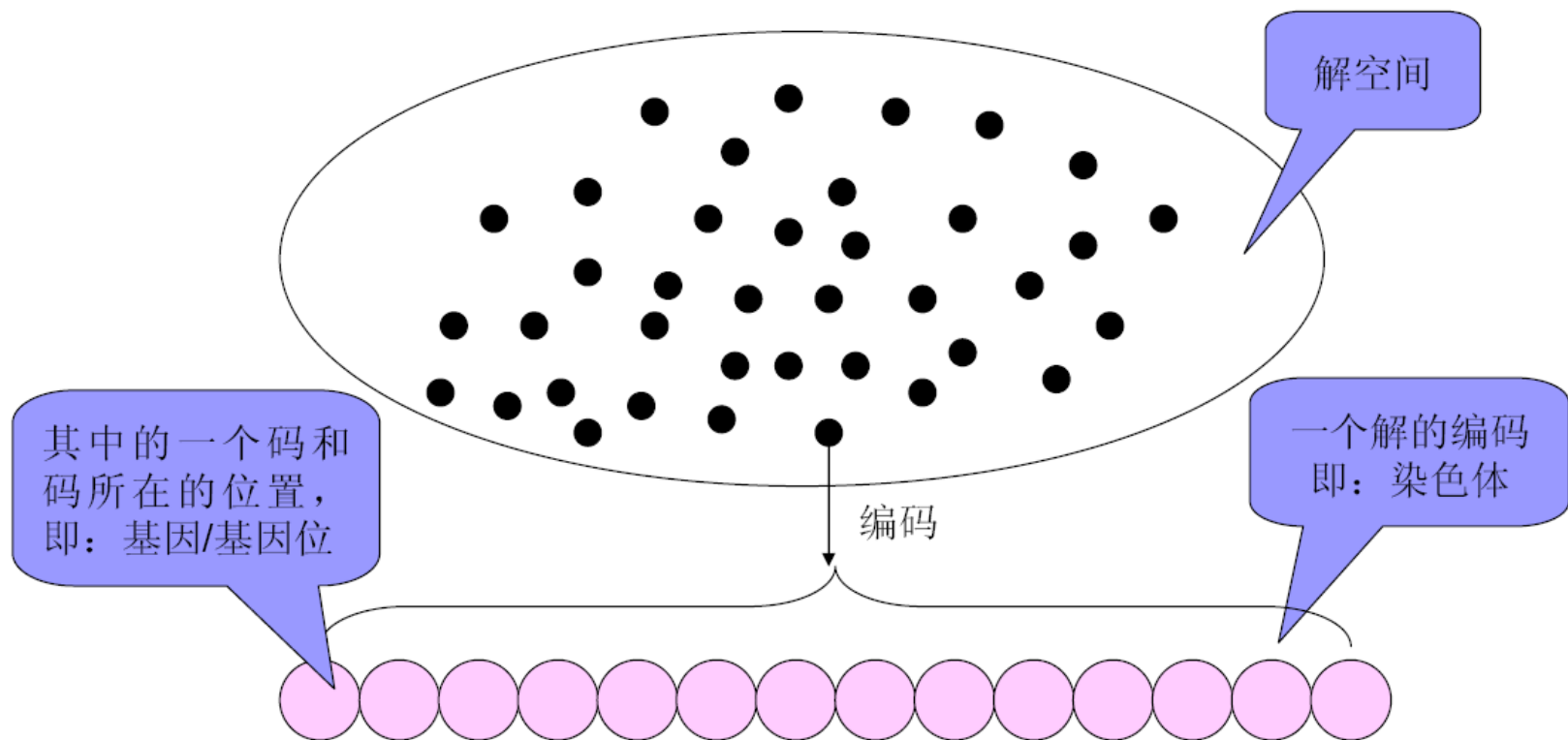
遗传算法的基本机理

● 遗传算子

- **选择算子 (Selection/Reproduction):** 选择算子从种群中按某一概率成对选择个体，某个体 x_i 被选择的概率 P_i 与其适应度值成正比。
- **交叉算子 (Crossover):** 交叉算子将被选中的两个个体的基因链按概率 P_c 进行交叉，生成两个新的个体，交叉位置是随机的。
- **变异算子 (Mutation):** 变异算子将新个体的基因链的各位按概率 P_m 进行变异，对二值基因链(0, 1 编码)来说即是取反。

上述各种算子的实现是多种多样的，而且许多新的算子正在不断地提出，以改进GA的某些性能。

解的编码 - GA的基础



编码与解码

- GA中的**编码方法**可分为三大类：二进制编码方法、浮点数编码方法和符号编码方法。
- **二进制编码方案**
- 是GA中最常用的一种编码方法。它所构成的个体基因型是一个二进制编码符号串。

编码与解码

- 二进制编码符号串的长度与问题所要求的求解精度有关。设某一参数的取值范围是 $[A, B]$, $A < B$ 。则二进制编码的编码精度为：

$$\delta = \frac{B - A}{2^l - 1}$$

- 假设某一个体的编码是：
- $X: b_l b_{l-1} b_{l-2} \dots b_2 b_1$, 则对应的解码公式为：

$$x = A + \frac{B - A}{2^l - 1} \left(\sum_{i=1}^l b_i 2^{i-1} \right)$$

编码与解码

- 例如，对于 $x \in [0, 1023]$ ，若用10位长的二进制编码来表示该参数，则下述符号串：

X: 0 0 1 0 1 0 1 1 1 1

就可表示一个个体。它所对应的参数值是 $x=175$ 。此时的编码精度为 $\delta=1$ 。

其他编码

- 格雷码是这样的一种编码方法，其连续的两个整数所对应的编码值之间仅仅有一个码位是不相同的，其余码位都完全相同。

十进制数	自然二进制数	格雷码	十进制数	自然二进制数	格雷码
0	0000	0000	8	1000	1100
1	0001	0001	9	1001	1101
2	0010	0011	10	1010	1111
3	0011	0010	11	1011	1110
4	0100	0110	12	1100	1010
5	0101	0111	13	1101	1011
6	0110	0101	14	1110	1001
7	0111	0100	15	1111	1000

其他编码

● 浮点编码

所谓浮点数编码方法是指个体染色体编码串中的基因值用某一范围内的一个浮点数来表示，个体的编码长度等于其决策变量的个数。因为这种编码方法使用的是决策变量的真实值，所以浮点数编码方法也叫做真值编码方法。

其他编码

- **符号编码方法**

个体染色体编码串中的基因值取自一个无数值含义、而只有代码含义的符号集。

- **这个符号集可以是一个字母表，如 $\{A,B,C,D,\dots\}$ ；也可以是一个数字序号表，如 $\{1,2,3,\dots\}$ ；还可以是一个代码表，如 $\{A_1,A_2,A_3,\dots\}$ 等等。**

其他编码

- **例如：**对于销售员旅行问题（Traveling Salesman Problem, TSP），按一条回路中城市的次序进行编码。从城市 w_1 开始，依次经过城市 w_2 ，……， w_n ，最后回到城市 w_1 ，我们就有如下编码表示：

$$w_1 \ w_2 \ \dots \ w_n$$

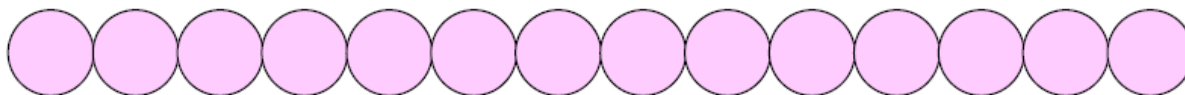
由于是回路，记 $w_{n+1} = w_1$ 。它其实是 $1, \dots, n$ 的一个循环排列。要注意 w_1, w_2, \dots, w_n 是互不相同的。

编码

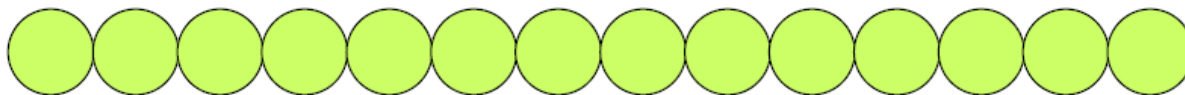
解的编号

解的编码

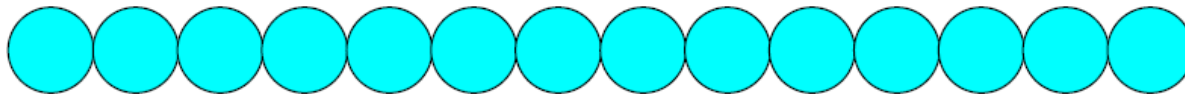
1



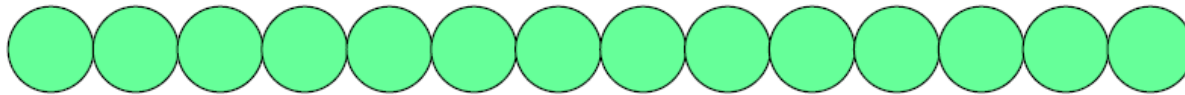
2



3



4



.

.

.

.

.

.

POP

群体规模

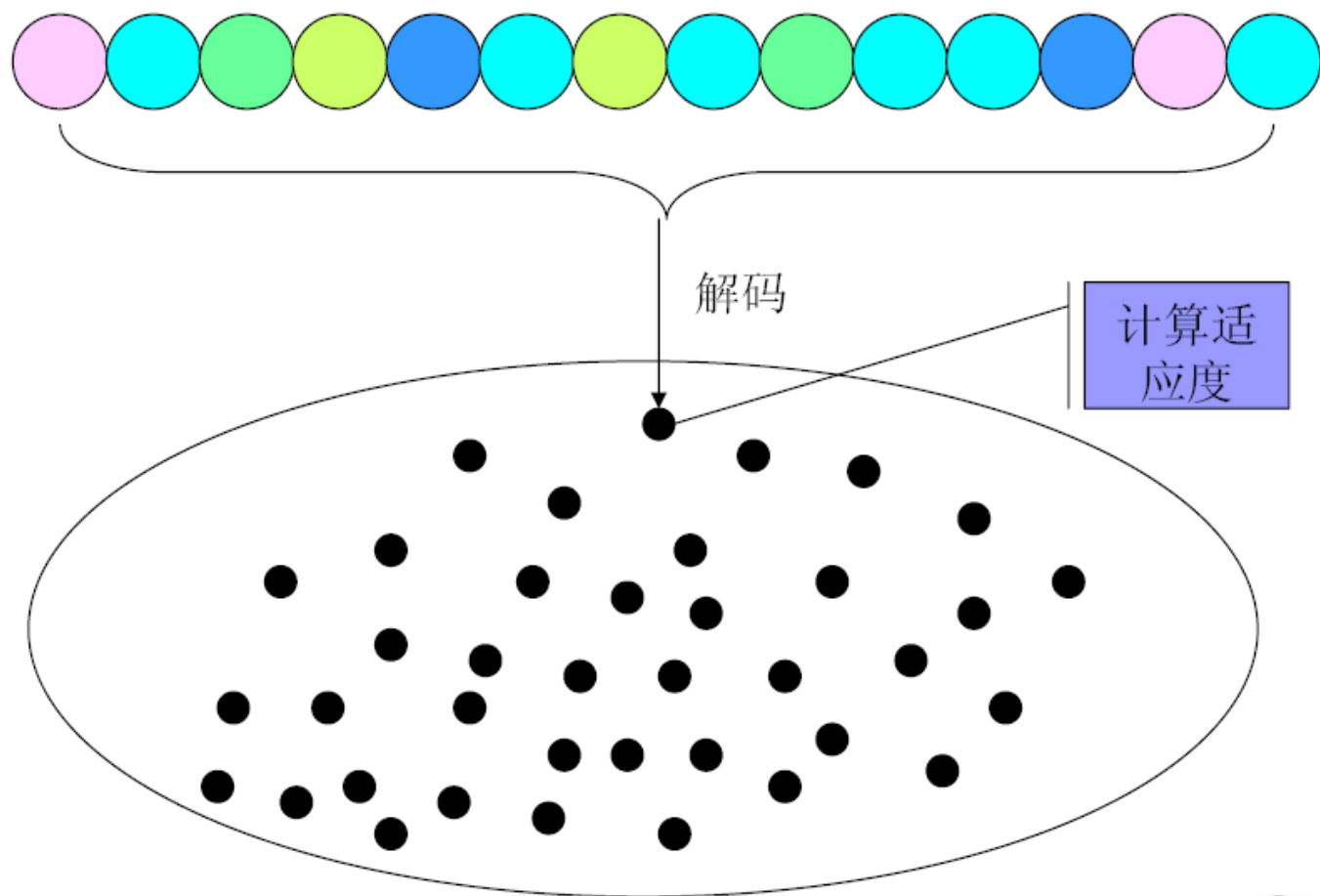


适应度函数

- 为了体现染色体的适应能力，引入了对问题中的每一个染色体都能进行度量的函数，叫**适应度函数**。通过适应度函数来衡量染色体的优、劣程度，它体现了自然进化中的优胜劣汰原则。一般，对于最大化的优化问题，适应度函数就是目标函数。
- 又如TSP问题的目标是路径总长度为最短，路径总长度的倒数就可以为TSP的适应度函数：

$$f(w_1 w_2 \cdots w_n) = \frac{1}{\sum_{j=1}^n d(w_j, w_j + 1)}$$

适应度计算



交叉算子—Crossover Operator

- 对于选中用于繁殖下一代的个体，随机地选择两个个体的相同位置，按交叉概率 P_c 在选中的位置实行交叉。这个过程反映了随机信息交换；
- 目的在于产生新的基因组合，也即产生新的个体。交叉时，可实行单点交叉或多点交叉。
- 例如有个体

P1=10010110

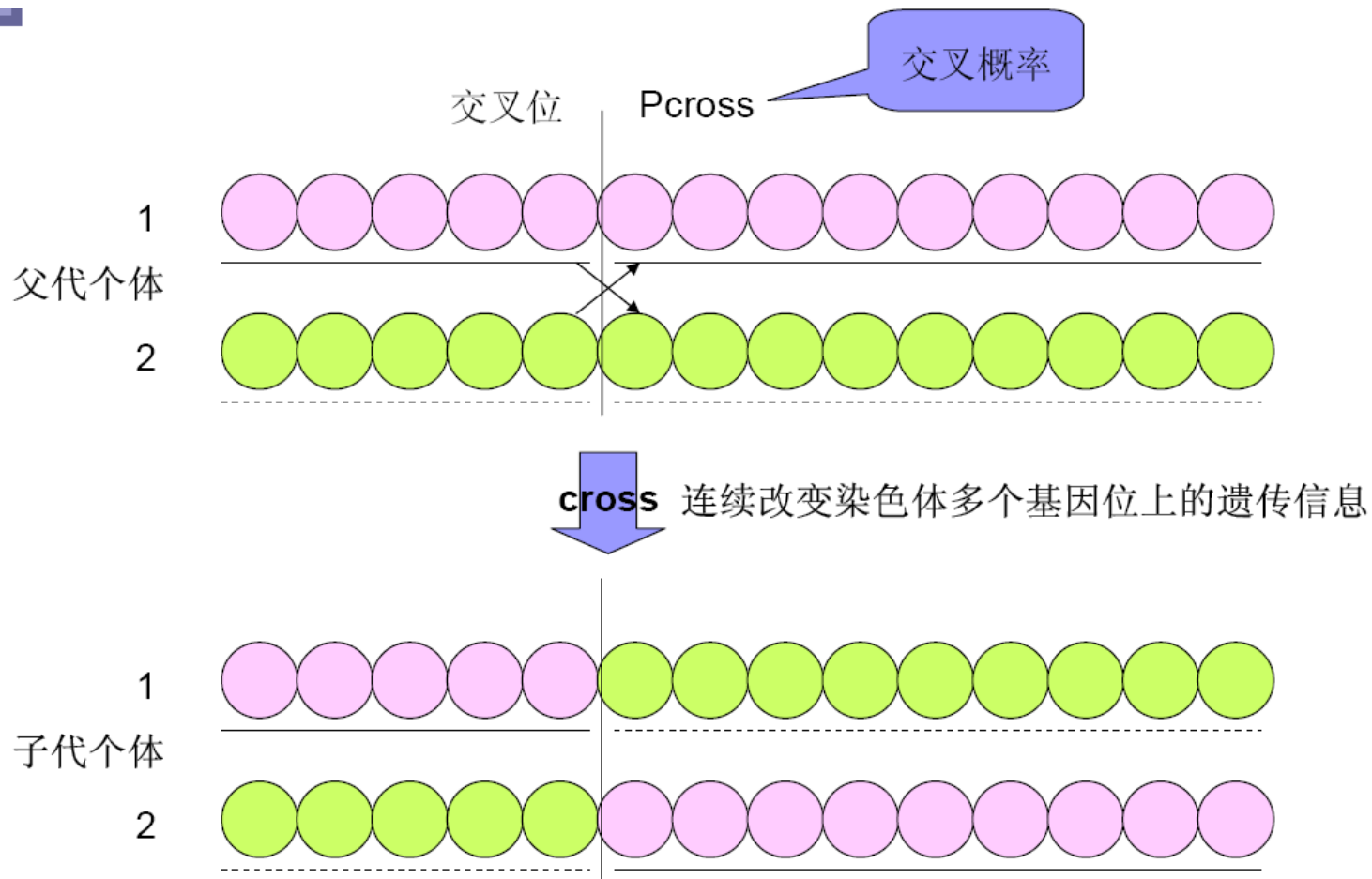
P2=01011110

选择它们的左边3位进行交叉操作，则有

P1=01010110

P2=10011110

交叉算子—Crossover Operator



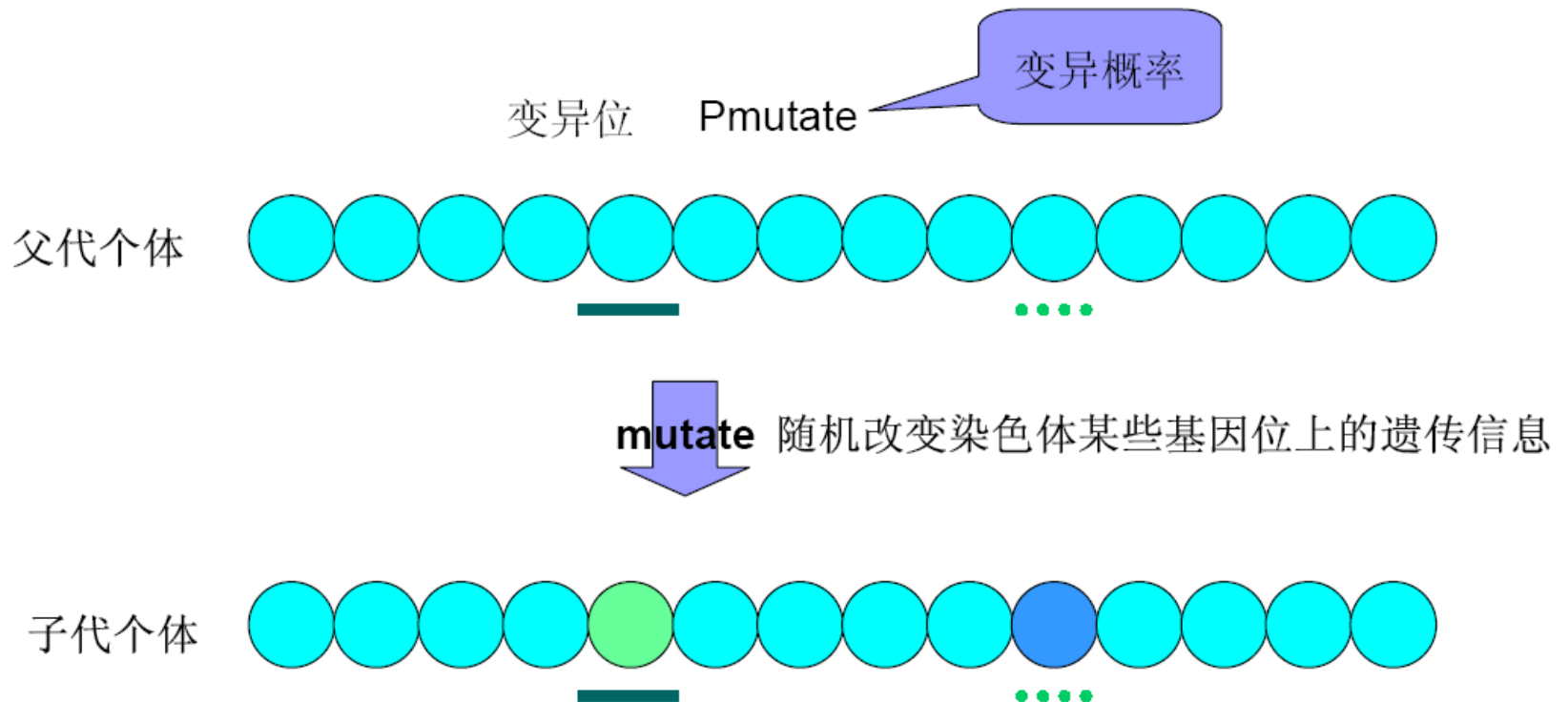
变异算子—Mutate Operator

- 根据生物遗传中基因变异的原理，以变异概率 P_m 对某些个体的某些位执行变异。
- 在变异时，对执行变异的串的对应位求反，即把1变为0，把0变为1。
- 变异概率 P_m 与生物变异极小的情况一致，所以， P_m 的取值较小。
- 例如有个体 $S = 10101110$ 。
对其的第1，4位置的基因进行变异，则有
 $S' = 00111110$

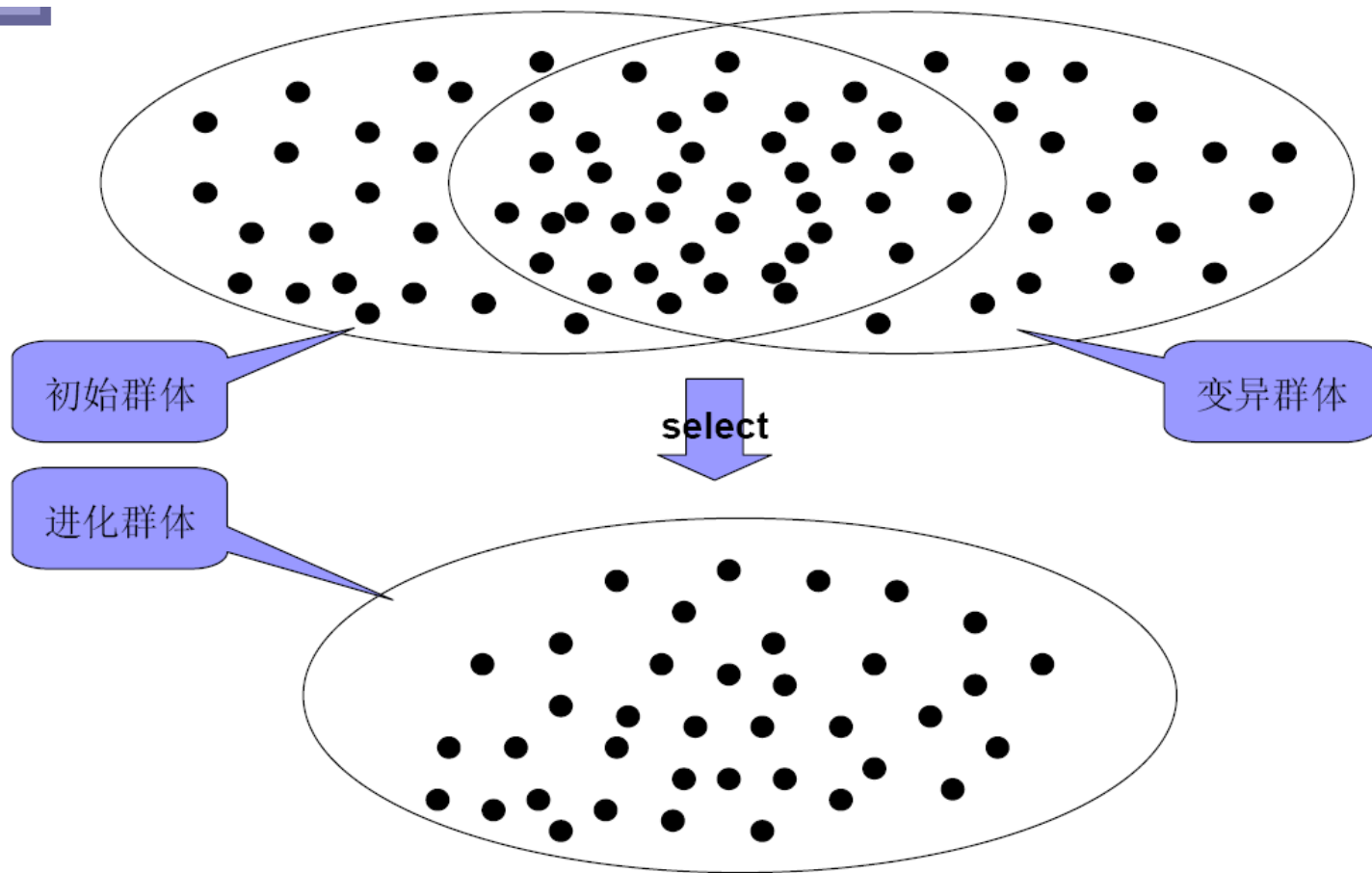
变异算子—Mutate Operator

- **变异能保证算法过程不会产生无法进化的单一种群。因为在所有的个体一样时，交叉是无法产生新的个体的，这时只能靠变异产生新的个体。**

变异算子—Mutate Operator



选择算子—Selection Operator



选择算子—Selection Operator

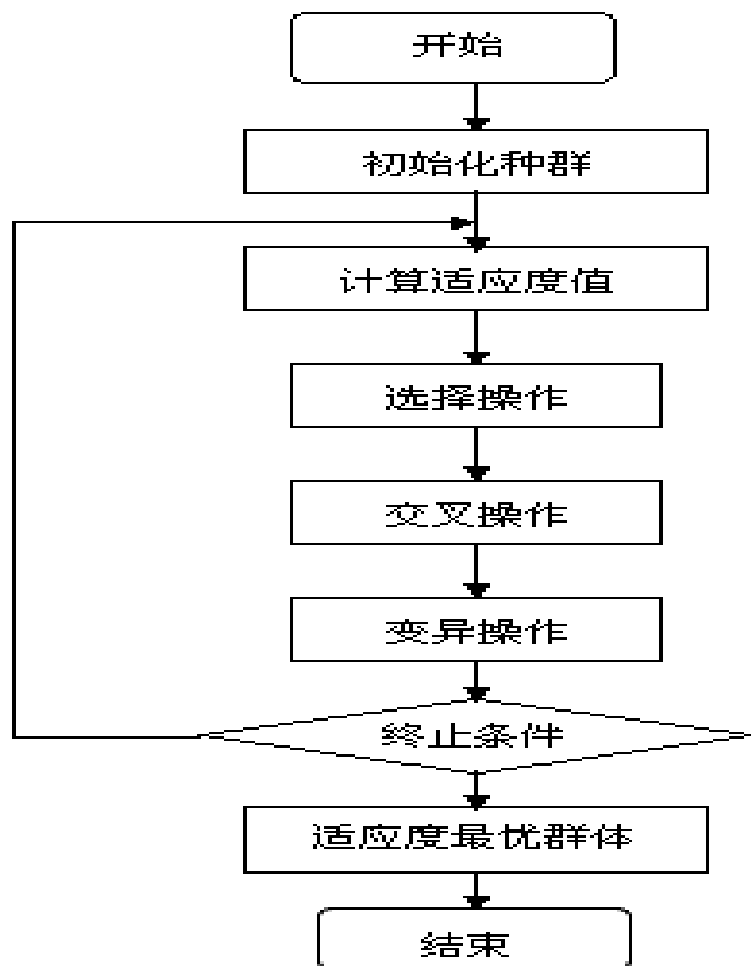
- **选择操作：根据适应度函数值所度量的个体的优、劣程度决定它在下一代是被淘汰还是被保留。**
- **简单遗传算法采用轮盘赌选择机制**

轮盘赌选择机制

令 $\sum f_i$ 表示种群的适应度值之总和， f_i 表示种群中第 i 个染色体的适应度值，则它产生后代的能力正好为其适应度值所占份额 $f_i / \sum f_i$ 。

- 显然，从上式可知：
 - (1) 适应度较高的个体，繁殖下一代的能力就较强。
 - (2) 适应度较小的个体，繁殖下一代的能力就较弱，甚至被淘汰。
- 这样，就产生了对环境适应能力较强的后代。对于问题求解角度来讲，就是选择出和最优解较接近的中间解。

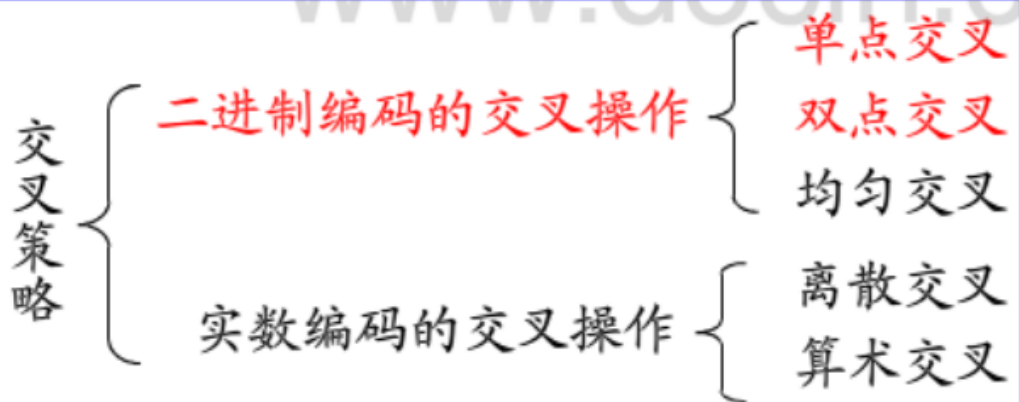
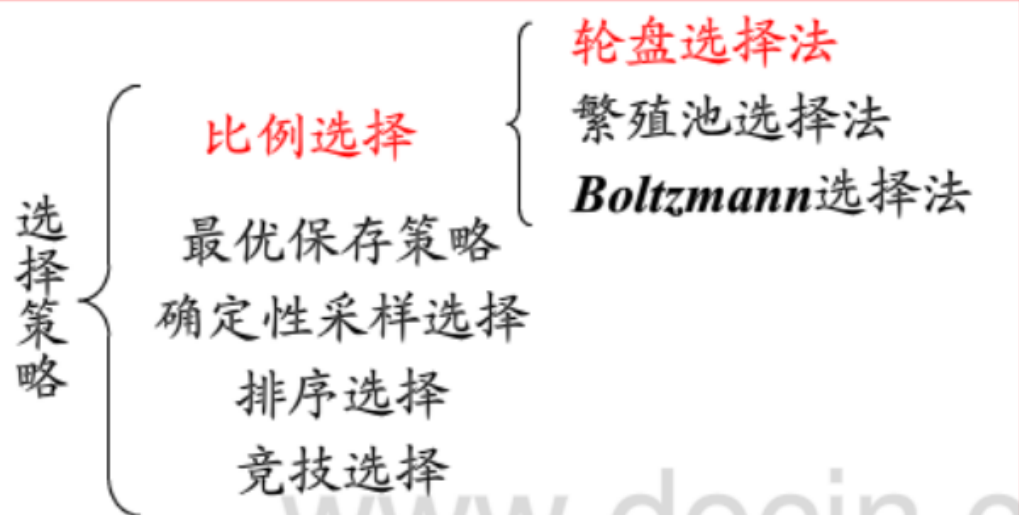
遗传算法的框图



简单遗传算法框图

1. 初始化群体;
2. 计算群体上每个个体的适应度值
3. 按由个体适应度值所决定的某个规则选择将进入下一代的个体;
4. 按概率 P_c 进行交叉操作;
5. 按概率 P_c 进行突变操作;
6. 若没有满足某种停止条件, 则转第2步; 否则进入下一步;
7. 输出群体中适应度值最优的染色体作为问题的满意解或最优解。

遗传算子

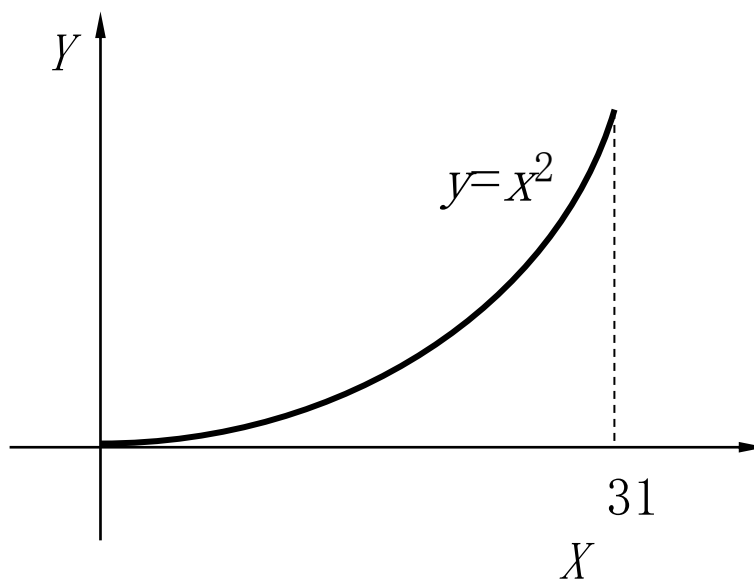


停止条件

- 1) 完成了预先给定的进化代数则停止;
- 2) 种群中的最优个体在连续若干代没有改进
- 3) 平均适应度在连续若干代基本没有改进时停止

遗传算法应用举例

例1 利用遗传算法求解区间 $[0, 31]$ 上的二次函数 $y=x^2$ 的最大值。



分析

原问题可转化为在区间 $[0, 31]$ 中搜索能使 y 取最大值的点 a 的问题。那么, $[0, 31]$ 中的点 x 就是个体, 函数值 $f(x)$ 恰好就可以作为 x 的适应度, 区间 $[0, 31]$ 就是一个(解)空间。这样, 只要能给出个体 x 的适当染色体编码, 该问题就可以用遗传算法来解决。

解

(1) 设定种群规模, 编码染色体, 产生初始种群。

将种群规模设定为4; 用5位二进制数编码染色体; 取下列个体组成初始种群 S_1 :

$$s_1 = 13 (01101), s_2 = 24 (11000)$$

$$s_3 = 8 (01000), s_4 = 19 (10011)$$

(2) 定义适应度函数, 取适应度函数:

$$f(x) = x^2$$

(3) 计算各代种群中的各个体的适应度, 并对其染色体进行遗传操作, 直到算法终止条件满足为止。

首先计算种群 S_1 中各个体

$$s_1 = 13(01101), \quad s_2 = 24(11000)$$

$$s_3 = 8(01000), \quad s_4 = 19(10011)$$

的适应度 $f(s_i)$ 。

容易求得

$$f(s_1) = f(13) = 13^2 = 169$$

$$f(s_2) = f(24) = 24^2 = 576$$

$$f(s_3) = f(8) = 8^2 = 64$$

$$f(s_4) = f(19) = 19^2 = 361$$

再计算种群 S_1 中各个体的选择概率。

选择概率的计算公式为

$$P(x_i) = \frac{f(x_i)}{\sum_{j=1}^N f(x_j)}$$

由此可求得

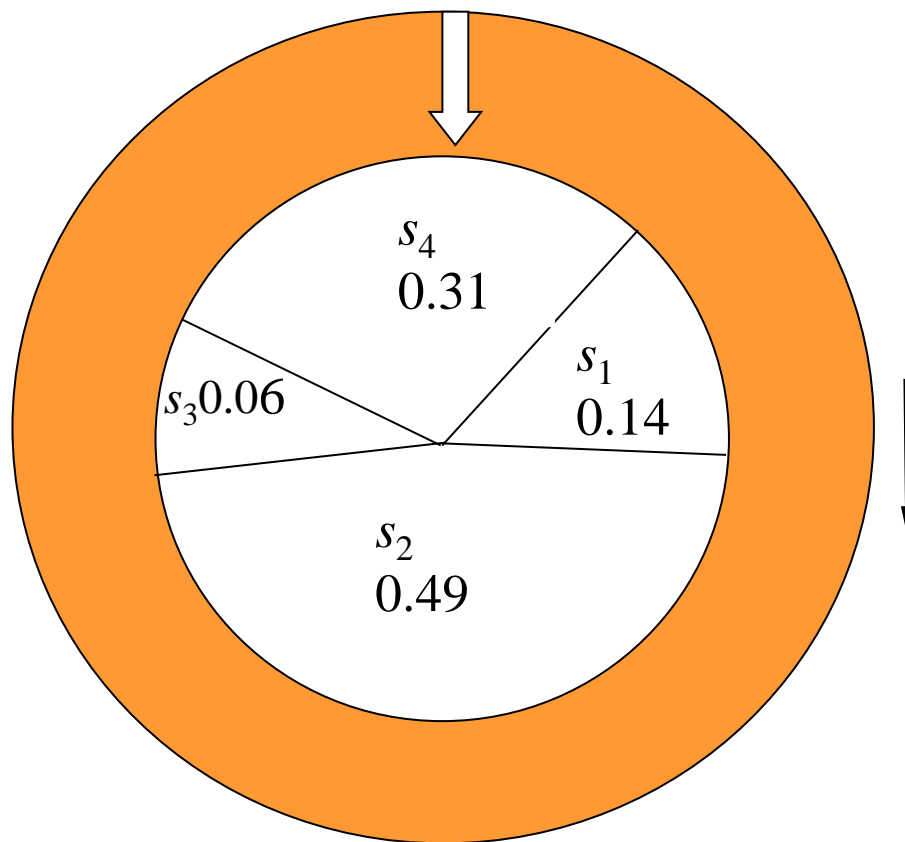
$$P(s_1) = P(13) = 0.14$$

$$P(s_2) = P(24) = 0.49$$

$$P(s_3) = P(8) = 0.06$$

$$P(s_4) = P(19) = 0.31$$

● 轮盘赌选择法



轮盘赌选择

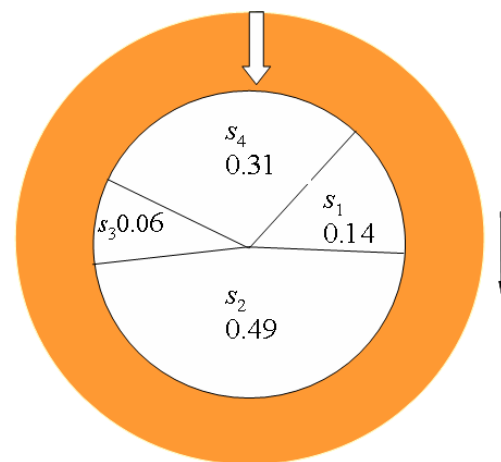
在算法中赌轮选择法可用下面的子过程来模拟:

① 在 $[0, 1]$ 区间内产生一个均匀分布的随机数 r 。

② 若 $r \leq q_1$, 则染色体 x_1 被选中。

③ 若 $q_{k-1} < r \leq q_k$ ($2 \leq k \leq N$), 则染色体 x_k 被选中。其中的 q_i 称为染色体 x_i ($i=1, 2, \dots, n$) 的积累概率, 其计算公式为

$$q_i = \sum_{j=1}^i P(x_j)$$



选择

设从区间 $[0, 1]$ 中产生4个随机数如下：

$$r_1 = 0.450126, \quad r_2 = 0.110347$$

$$r_3 = 0.572496, \quad r_4 = 0.98503$$

染色体	适应度	选择概率	积累概率	选中次数
$s_1=01101$	169	0.14	0.14	1
$s_2=11000$	576	0.49	0.63	2
$s_3=01000$	64	0.06	0.69	0
$s_4=10011$	361	0.31	1.00	1

于是，经选择得种群：

$$s_1' = 11000 \ (24), \ s_2' = 01101 \ (13)$$

$$s_3' = 11000 \ (24), \ s_4' = 10011 \ (19)$$

交叉

设交叉率 $p_c=100\%$ ，即 S_1 中的全体染色体都参加交叉运算。

设 s_1' 与 s_2' 配对， s_3' 与 s_4' 配对。分别交换后两位基因，得新染色体：

$$s_1''=11001 \quad (25), \quad s_2''=01100 \quad (12)$$

$$s_3''=11011 \quad (27), \quad s_4''=10000 \quad (16)$$

变异

设变异率 $p_m=0.001$ 。

这样，种群 S_1 中共有

$$5 \times 4 \times 0.001 = 0.02$$

位基因可以变异。

0.02位显然不足1位，所以本轮遗传操作不做变异。

于是，得到第二代种群 S_2 ：

$$s_1=11001 \quad (25) \quad , \quad s_2=01100 \quad (12)$$

$$s_3=11011 \quad (27) \quad , \quad s_4=10000 \quad (16)$$

第二代种群 S_2 中各染色体的情况

染色体	适应度	选择概率	积累概率	估计的选中次数
$s_1=11001$	625	0.36	0.36	1
$s_2=01100$	144	0.08	0.44	0
$s_3=11011$	729	0.41	0.85	2
$s_4=10000$	256	0.15	1.00	1

假设这一轮选择操作中，种群 S_2 中的
4个染色体都被选中，则得到种群：

$$s_1' = 11001 \quad (25), \quad s_2' = 01100 \quad (12)$$

$$s_3' = 11011 \quad (27), \quad s_4' = 10000 \quad (16)$$

做交叉运算，让 s_1' 与 s_2' ， s_3' 与 s_4' 分别交换后
三位基因，得

$$s_1'' = 11100 \quad (28), \quad s_2'' = 01001 \quad (9)$$

$$s_3'' = 11000 \quad (24), \quad s_4'' = 10011 \quad (19)$$

这一轮仍然不会发生变异。

于是，得第三代种群 S_3 ：

$$s_1=11100 \ (28) , s_2=01001 \ (9)$$

$$s_3=11000 \ (24) , s_4=10011 \ (19)$$

第三代种群 S_3 中各染色体的情况

染色体	适应度	选择概率	积累概率	估计的选中次数
$s_1=11100$	784	0.44	0.44	2
$s_2=01001$	81	0.04	0.48	0
$s_3=11000$	576	0.32	0.80	1
$s_4=10011$	361	0.20	1.00	1

设这一轮的选择结果为：

$$s_1' = 11100 \quad (28), \quad s_2' = 11100 \quad (28)$$

$$s_3' = 11000 \quad (24), \quad s_4' = 10011 \quad (19)$$

做交叉运算，让 s_1' 与 s_4' ， s_2' 与 s_3' 分别交换后两位基因，得

$$s_1'' = 11111 \quad (31), \quad s_2'' = 11100 \quad (28)$$

$$s_3'' = 11000 \quad (24), \quad s_4'' = 10000 \quad (16)$$

这一轮仍然不会发生变异。

于是，得第四代种群 S_4 ：

$$s_1=11111 \quad (31) , \quad s_2=11100 \quad (28)$$

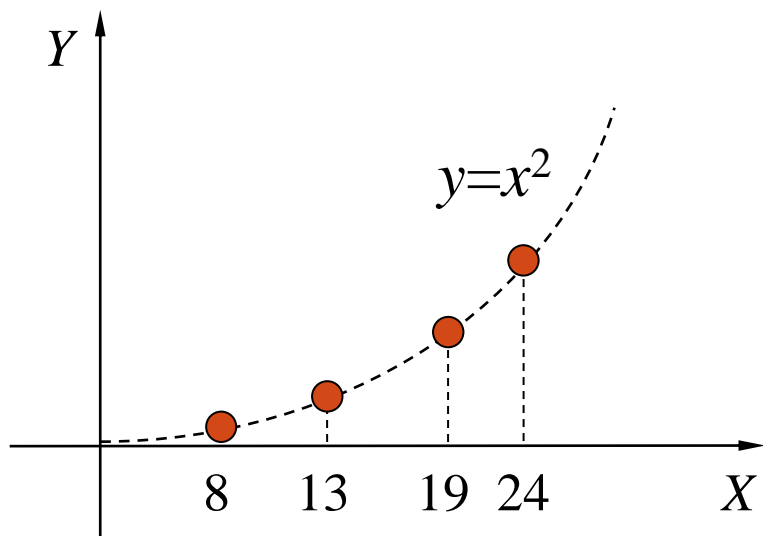
$$s_3=11000 \quad (24) , \quad s_4=10000 \quad (16)$$

显然，在这一代种群中已经出现了适应度最高的染色体 $s_1=11111$ 。

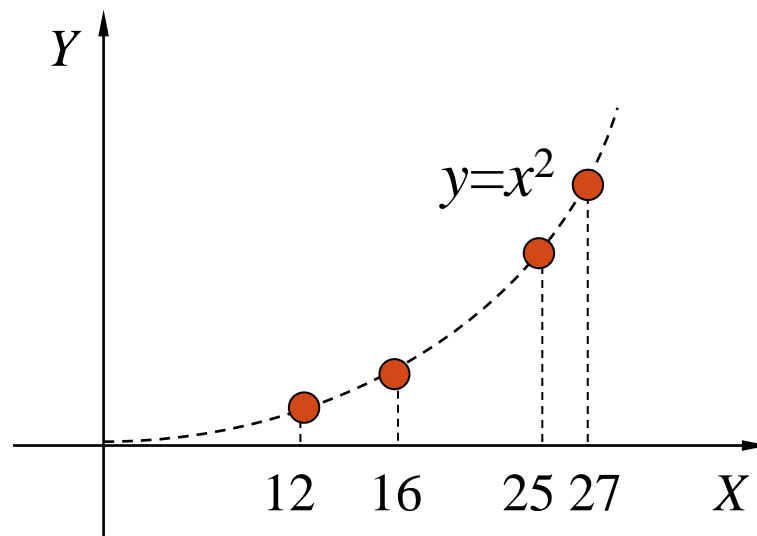
算法终止时将染色体“11111”作为最终结果输出。

然后，将染色体“11111”解码为表现型，即得所求的最优解：31。

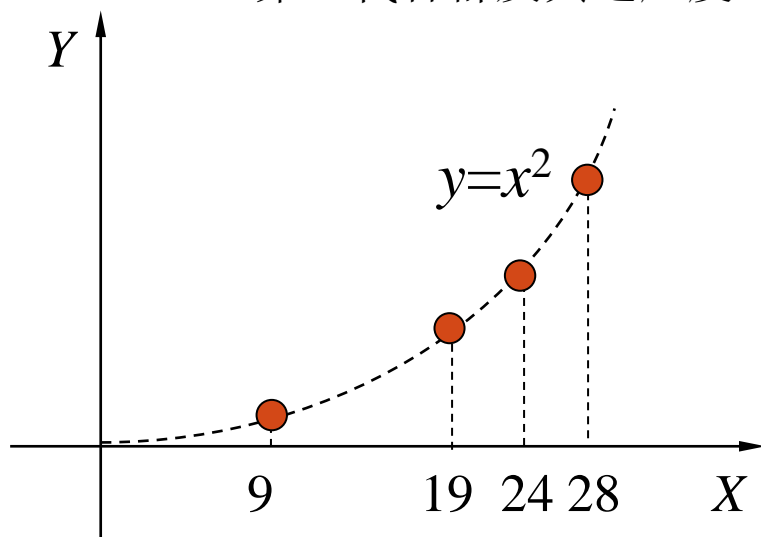
将31代入函数 $y=x^2$ 中，即得原问题的解，即函数 $y=x^2$ 的最大值为961。



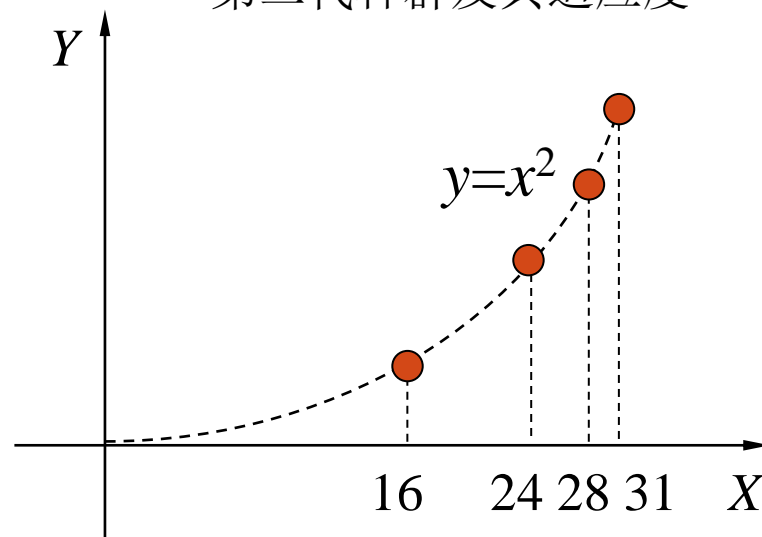
第一代种群及其适应度



第二代种群及其适应度



第三代种群及其适应度



第四代种群及其适应度

遗传算法的特点

- ◇ 遗传算法是从问题解的编码组开始而非从单个解开始搜索；
- ◇ 遗传算法利用目标函数的适应度这一信息而非利用导数或其它辅助信息来指导搜索；
- ◇ 遗传算法利用选择、交叉、变异等算子而不是利用确定性规则进行随机操作。

遗传算法存在的不足

- 遗传算法可以在多种领域应用，展示了它的潜力和宽广前景；但是，遗传算法还有大量的问题需要研究，目前也还有各种不足。
- 首先，在变量多，取值范围大或无给定范围时，收敛速度下降；
- 其次，可找到最优解附近，但无法精确确定最优解位置；
- 最后，遗传算法的参数选择尚未有定量方法。
- 对遗传算法，还需要进一步研究其数学基础理论；还需要在理论上证明它与其它优化技术的优劣及原因；还需研究硬件化的遗传算法；以及遗传算法的通用编程和形式等。

其它进化算法

进化算法 (Evolutionary Algorithm—EA)

目前研究的进化算法主要有三种典型的算法：
遗传算法、进化规划和进化策略。

这三种算法是彼此独立发展起来的。

进化策略

- **进化策略 (Evolution Strategies, ES) 是一类模仿自然进化原理以求解参数优化问题的算法。**
- **它是由雷切伯格 (Rechenberg)、施韦费尔 (Schwefel) 和彼得·比纳特 (Peter Bienert) 于 1964 年提出的，并在德国共同建立的。**

进化策略与遗传算法的结构类似，只是在算法的具体策略上存在差异。

进化策略

- 染色体编码：浮点数编码
- 交叉：离散重组、中值重组、混杂重组
- 变异：在每个分量上面加上零均值、某一方差的高斯分布的变化产生新的个体
- 选择： $(\mu+\lambda)$ -ES、 (μ, λ) -ES

$(\mu+\lambda)$ -ES

搜索开始时，建立一个初始种群 POP_0 ，包含 μ 个个体。从初始种群开始，迭代计算一系列种群。在每一次迭代 $iter$ 中，从当前代 POP_{iter} 产生 λ 个子代：

- (1) 从当前代 POP_{iter} 中选择两个个体，作为父代用于重组。父代的选择是无偏的。
- (2) 通过所选父代的重组，产生一个新个体。
- (3) 对新个体施行变异和评估。

从 λ 个子代与 μ 个 POP_{iter} 代个体组成的集合中，选择 μ 个个体形成新一代种群 POP_{iter+1} 。现在，将解的质量作为选择的标准：即，选择 μ 个具有最高质量的个体来代替当前代。

(μ, λ) -ES

从 λ 个子代个体组成的集合中，选择 μ 个个体形成新一代种群 POP_{iter+1}

进化规划

进化规划 (Evolutionary Programming) 又称为进化编程，是由福格尔 (Fogel) 在1962年提出的一种模仿人类智能的方法。进化规划根据正确预测的符号数来度量适应值。通过变异，为父代群体中的每个机器状态产生一个子代。父代和子代中最好的部分被选择生存下来。

起初，为求解预测问题而提出了一种有限状态机进化模型。机器的状态基于均匀随机分布的规律进行变异。

后来，进化规划的思想被拓展到实数空间，用来求解实数空间中的优化问题。

在变异运算中引入了正态分布的技术。

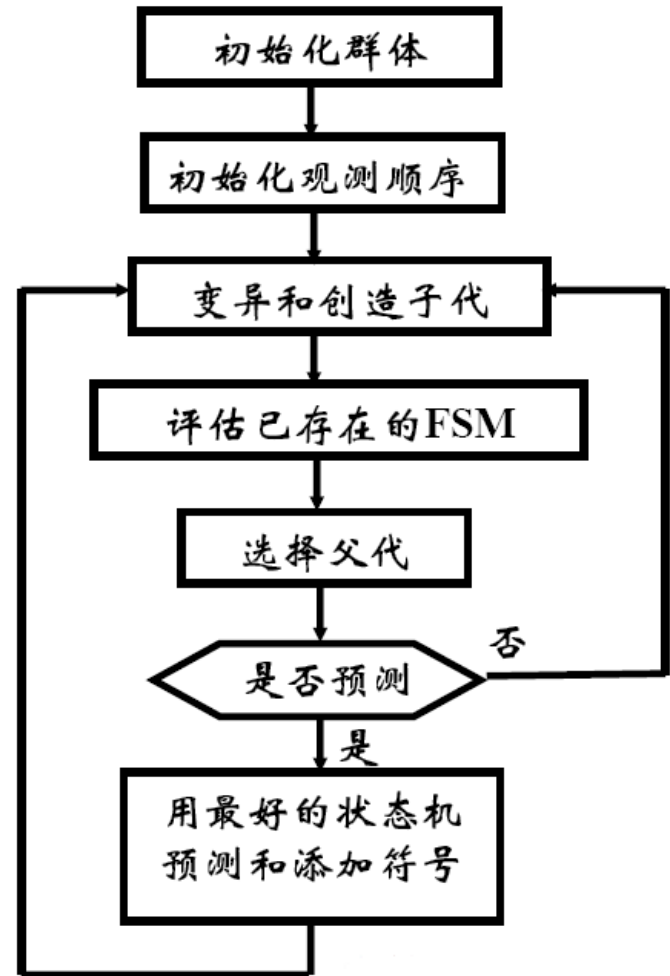
进化规划演变成一种优化搜索的算法。

进化规划的机理与表示

- **进化规划的过程**，可理解为从所有可能的计算机程序形成的空间中，搜索具有高的适应度的计算机程序个体。在进化编程中，可能有几百或几千个计算机程序参与遗传进化。
- **进化规划设计强调群体行为的变化**。进化规划系统的表示自然地面向任务级。一旦选定一种适应性表示，就可以定义依赖于该表示的变异操作，在具体的父辈行为上创建后代。

进化规划流程

- 1. 生成初始种群
- 2. While Not-End Do
 - 1) 变异
 - 2) 计算个体适应度
 - 3) 选择
 - 4) 组成新种群
- 3. End While



遗传算法、进化策略和进化规划

- 进化计算的三种算法——**遗传算法、进化策略和进化规划**都是模拟生物界自然进化过程而建立的鲁棒性计算机算法。在统一框架下对三种算法进行比较，可以发现它们有许多相似之处，同时也存在较大的差别。
- **进化策略和进化规划**都把**变异**作为主要搜索算子，而在**标准的遗传算法**中，变异只处于次要位置。**交叉**在遗传算法中起着重要作用，而在**进化规划**中却被完全省去，在进化策略中与自适应结合使用，起了很重要的作用。**标准遗传算法和进化规划**都强调随机选择机制的重要性，而从进化策略的角度看，选择是完全确定的。

遗传算法、进化策略和进化规划

比较项目	遗传算法	进化策略	进化规划
个体表现形式	离散值	连续值	连续值
参数调整方法	无	标准偏差、协方差	方差
适应度评价方法	变换目标函数值	直接采用目标函数值	变换目标函数值
个体变异算子	辅助搜索方法	主要搜索方法	唯一搜索方法
个体交叉算子	主要搜索方法	辅助搜索方法	不使用
选择算子	概率选择	确定性选择	概率选择

进化计算应用

- (1) **复杂的非线性最优化问题**：对具有多个局部极值的非线性最优化问题。普通的优化方法一般难以找到全局最优解，而进化计算方法可以克服这一缺点，找到全局最优解。
- (2) **复杂的组合规划或整数规划问题**：大多数组合规划或整数规划问题属于NP问题，难以找到有效的求解方法；进化计算广泛用于求解这类问题，在可以承受的时间内求得满意解，如旅行商问题、装箱问题等。
- (3) **生物学**：进化计算起源于生物现象的模拟，现在反过来用于生物学的研究，如利用进化算法研究小生境 (Niche) 理论和生物物种的形成等。

进化计算应用

- (4) **计算机科学**：进化计算广泛用于计算机科学的研究，如用于图象处理和自动识别以及文档自动处理等。
- (5) **工程应用**：进化计算越来越多地用于工程实际，如通讯网络的优化设计、超大规模集成电路的布线、飞机外形的设计等。
- (6) **社会科学**：进化计算在社会科学的许多领域也有广泛应用，如人类行为规范进化过程的模拟、人口迁移模型的建立等

人工生命

- ◇自然界是生命之源。自然生命千千万万，千姿百态，千差万别，巧夺天工，奇妙无穷。
- ◇人工生命（**试图通过人工方法建造具有自然生命特征的人造系统。**人工生命（Artificial Life , AL）
- ◇人工生命是生命科学、信息科学和系统科学等学科交叉研究的产物，其研究成果必将促进人工智能的发展。



人工生命研究的起源和发展

- 人类长期以来一直力图用科学技术方法模拟自然界，包括人脑本身。
- 人工生命的许多早期研究工作也源于人工智能。
- 20世纪**60年代**罗森布拉特研究感知机，斯塔尔（Stahl）建立细胞活动模型，林登迈耶（Lindenmayer）提出了生长发育中的细胞交互作用数学模型。这些模型支持细胞间的通信和差异。

20世纪**70年代**以来，康拉德（Conrad）等研究人工仿生系统中的自适应、进化和种群动力学，提出不断完善的“人工世界”模型。
- **80年代**，人工神经网络再度兴起促进人工生命的发展。在1987年第一次人工生命研讨会上，美国圣塔菲研究所（Santa Fe Institute, SFI）非线性研究组的兰顿（Langton）正式提出人工生命的概念，建立起人工生命新学科。此后，人工生命研究进入一个蓬勃发展的新时期

人工生命的定义和研究意义

- 人工生命是一项抽象地提取控制生物现象的基本动态原理，并且通过物理媒介（如计算机）来模拟生命系统动态发展过程的研究工作。
- 人工生命的定义

通俗地讲，人工生命即人造的生命，非自然的生命。然而，要对人工生命做出严格的定义，要对问题进行深入研究。

人工生命的定义

- 1987年兰德提出的人工生命定义为：“人工生命是研究能够演示出自然生命系统特征行为的人造系统”。通过计算机或其它机器对类似生命的行为进行综合研究，以便对传统生物学起互补作用。
- 兰德在计算机上演示了他们研制的具有生命特征的软件系统，并把这类具有生命现象和特征的人造系统称为人工生命系统。

自然生命的共同特征和现象

从各种不同的自然生命的特征和现象中，可以归纳和抽象出**自然生命的共同特征和现象**，包括但不限于

- (1) 自繁殖、自进化、自寻优。
- (2) 自成长、自学习、自组织。
- (3) 自稳定、自适应、自协调。
- (4) 物质构造。
- (5) 能量转换。
- (6) 信息处理。

如果把人工生命定义为具有自然生命现象和（或）特征的人造系统，那么，凡是具有上述自然生命现象和（或）特征的人造系统，都可称为人工生命。

研究人工生命的意义

人工生命是自然生命的模拟、延伸与扩展，其研究开发有重大的科学意义和广泛的应用价值。

(1) 开发基于人工生命的工程技术新方法、新系统、新产品。

(2) 为自然生命的研究提供新模型、新工具、新环境。人工生命的研究开发可以为自然生命的研究探索提供新模型、新工具、新环境。

(3) 延伸人类寿命、减缓衰老、防治疾病。

(4) 扩展自然生命，实现人工进化和优生优育。

(5) 促进生命科学、信息科学、系统科学的交叉与发展。

人工生命的研究内容及方法

- 人工生命的研究对象包括人工动物、人工植物和人工人等，而人工人的研究又涉及人工脑和其它人工器官。

1、人工生命的研究内容

(1) 构成生物体的内部系统，包括脑、神经系统、内分泌系统、免疫系统、遗传系统、酶系统、代谢系统等。

(2) 生物体及其种群的外部系统，包括环境适应系统和遗传进化系统等。

人工生命的科学框架

(3) 生命现象仿生系统

- 生命现象的建模与仿真
- 进化动力学
- 人工生命的计算理论和工具
- 进化机器人
- 进化和学习等方面的结合
- 人工生命的应用

人工生命的研究方法

- 从生物体内部和外部系统的各种信息出发，可得到人工生命的不同研究方法，主要可分为两类：
 - (1) 信息模型法
 - (2) 工作原理法
- 人工生命的研究技术途径也可分为两种：
 - (1) 工程技术途径
 - (2) 生物科学途径

人工生命的研究技术途径

➤ 工程技术途径

利用计算机、自动化、微电子、精密机械、光电通信、人工智能、神经网络等有关工程技术方法和途径，研究开发、设计制造人工生命。通过计算机屏幕，以三维动画，虚拟现实的软件方法或采用光机电一体化的硬件装置来演示和体现人工生命。

➤ 生物科学途径

利用生物科学方法和技术，通过人工合成、基因控制，无性繁殖过程，培育生成人工生命。

由于伦理学、社会学、人类学等方面的问题，通过生物科学途径生成的人工生命，如克隆人引起了不少争论。需要研究和制订相应的社会监督、国家法律和国际公约。

人工生命的实例

- 人工脑
- 计算机病毒
- 计算机进程
- 细胞自动机
- 人工核苷酸

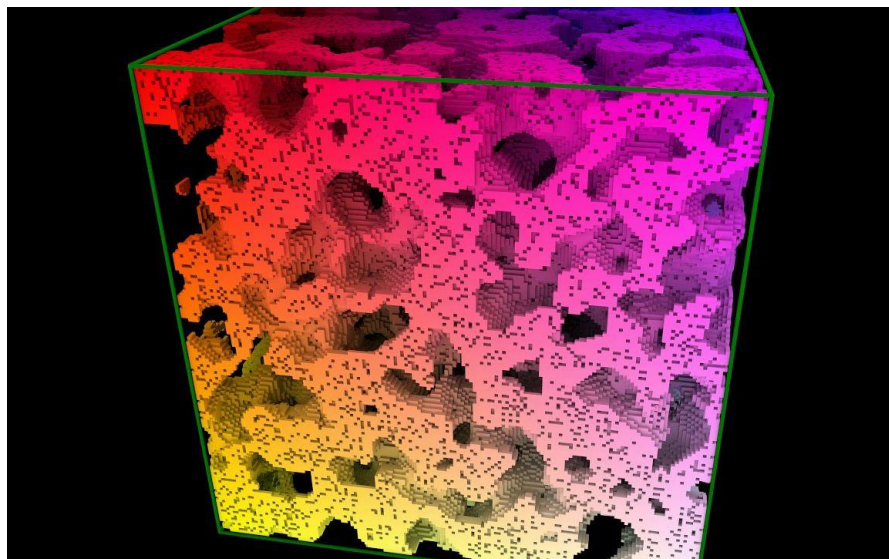
人工脑

波兰人工智能和心理学教授安奇·布勒及一些日本学者在1996年第四届国际人工生命会议上作了题为“针对脑通信的进化系统——走向人工脑”的专题报告。

- **研究方法**：将进化计算、非平衡动力学、林登迈耶(lindenmayer)系统（简称L系统）的产生语法、细胞自动机方式的复制器、神经学习等加以集成和融合。
 - 研究手段：硬件、软件、纳米技术
 - 相关概念：达尔文芯片和达尔文机器
 - 脑计划的目标：建造一个人工脑

细胞自动机

细胞自动机是一种人工细胞陈列，每个细胞具有离散结构，按照预先规定的规则，这些细胞的状态可随时间变化，通过陈列传递规则，计算每个细胞的当前状态及其近邻细胞状态，所有细胞均自发地更新状态

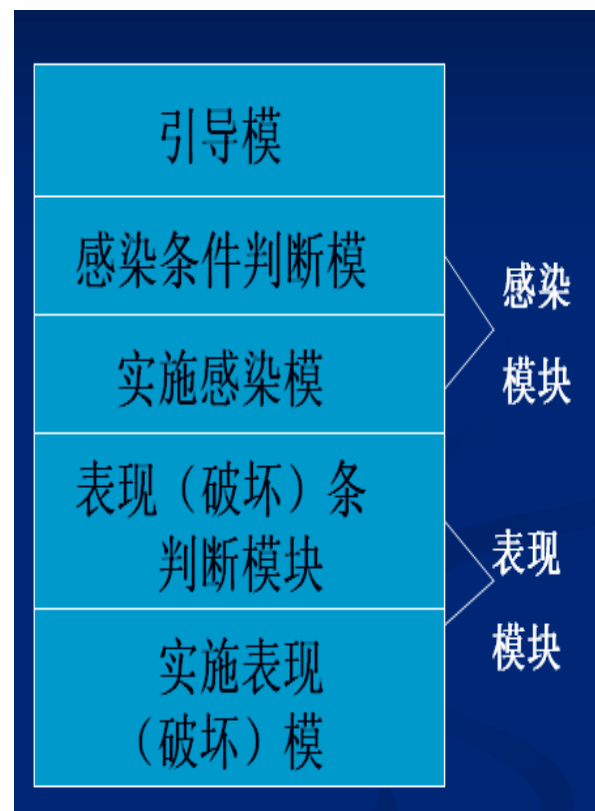


计算机病毒

- 计算机领域引入“病毒”的用法，只是对生物学病毒的一种借用，用以形象地刻画这些“特殊程序”的特征。
- 一般地说，**计算机病毒是一种特殊的危害计算机系统的程序**。就像生物病毒一样，计算机病毒有独特的复制能力。计算机病毒可以很快地蔓延，又常常难以根除。它们能把自身附着在各种类型的文件上。当文件被复制或从一个用户传送到另一个用户时，它们就随同文件一起蔓延开来。
- 除复制能力外，某些计算机病毒还有其它一些共同特性：一个被污染的程序能够传送病毒载体。当你看到病毒载体似乎仅仅表现在文字和图象上时，它们可能也已毁坏了文件、再格式化了你的硬盘驱动或引发了其它类型的灾害。若是病毒并不寄生于一个污染程序，它仍然能通过占据存储空间给你带来麻烦，并降低你的计算机的全部性能。

计算机病毒

- 计算机病毒之所以具有寄生能力和破坏能力，与病毒程序的结构有关。病毒的一般逻辑程序结构如右图所示：
- 各模块的功能的功能分别是：
 - **引导模块**：将病毒引入内存并使其后面的两个模块处于激活状态。
 - **感染模块**：在感染条件满足时把病毒感染到的所攻击的对象上。
 - **表现（破坏）模块**：在病毒发作条件（表现、破坏条件）满足时，实施对系统的干扰和破坏活动



计算机进程

计算机进程类似于计算机病毒，把进程当作生命体，可在时空中繁殖，从环境中汲取信息，修改所在的环境。该进程与物质媒体交互作用以支持这些物质媒体（如处理器、内存等）。进程具有生命的特征，可以在内存某个地方之外活着，等待适当的条件重新出现以恢复它们的活动状态。

人工核苷酸

1960年索尔·施皮格尔 (Sol Spiegelhe) 和他的同事结合当时已知的分子的最小集合，允许在一个试管中进行核糖核酸 (RNA) 的自复制，产生核苷酸前体、无机分子，能源，复制酶以及来自 α β 细菌噬菌体的RNA雏形。经过一系列的转移，RNA分子数目快速增长，同时分子本身变小，并达到最小尺寸。这些复制和进化的RNA分子是同原是人工生命形式相类似的。

本章内容小结

✓ 进化计算

遗传算法

进化策略

进化规划

✓ 人工生命

编码与解码

适应度函数

交叉算子

变异算子

选择算子