

人工智能概论-计算智能

赵亚伟

zhaoyw@ucas.ac.cn

中国科学院大学 大数据分析技术实验室

2018.6.9

智能优化算法

- **智能优化算法**又称**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（AI经典部分），理论上可以在一定的时间内找到最优解或近似最优解。
-

智能优化算法的特点

- 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。
 - 求解的过程通常是一个迭代的过程，即不是一步就完成。
 - 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。
-

常用的智能优化算法

- ☐ 遗传算法
 - ☐ 模拟退火算法
 - ☐ 禁忌搜索算法
 - ☐ 粒子群算法
 - ☐ 蚁群算法
 - ☐
-

目录

- 进化计算
 - 遗传算法
 - 人工生命
 - 粒子群优化算法
 - 蚁群算法
 - 小结
-

□ 进化计算包括:

- 遗传算法(genetic algorithms, GA)
- 进化策略(evolutionary strategies)
- 进化编程(evolutionary programming)
- 遗传编程(genetic programming)

□ 人类不满足于模仿生物进化行为, 希望能够建立具有自然生命特征的人造生命和人造生命系统。

□ 人工生命是人工智能和计算智能的一个新的研究热点。

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 Bieri**）于1964年提出的，并在德国共同建立的。

进化策略的算法模型

- 寻求与函数极值关联的实 n 维矢量 \mathbf{x} 。
- 随机选择父矢量的初始种群。
- 父矢量 $\mathbf{x}_i, i = 1, \dots, p$ 产生子代矢量 \mathbf{x}_i 。
- 对误差 ($i = 1, \dots, p$)排序以选择和决定保持哪些矢量。
- 继续产生新的试验数据以及选择最小误差矢量。

进化策略和遗传算法的区别

- 进化策略和遗传算法有着很强的相似性，它们都是一类模仿自然进化原理的算法。
- 两者也存在着区别，其中最基本的区别是它们的研究领域不同。
 - 进化策略是一种数值优化的方法，它采用一个具有自适应步长和倾角的特定爬山方法。
 - 遗传算法从广义上说是一种自适应搜索技术。

进化编程

- ❑ 进化编程(**Evolutionary Programming, EP**), 又称为进化规划 (**Evolutionary Planning**) , 是由福格尔 (**Fogel**) 在**1962**年提出的一种模仿人类智能的方法。
- ❑ 进化编程根据正确预测的符号数来度量适应值。通过变异, 为父代种群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。
- ❑ 它的提出是受自然生物进化机制的启发。

进化编程的机理与表示

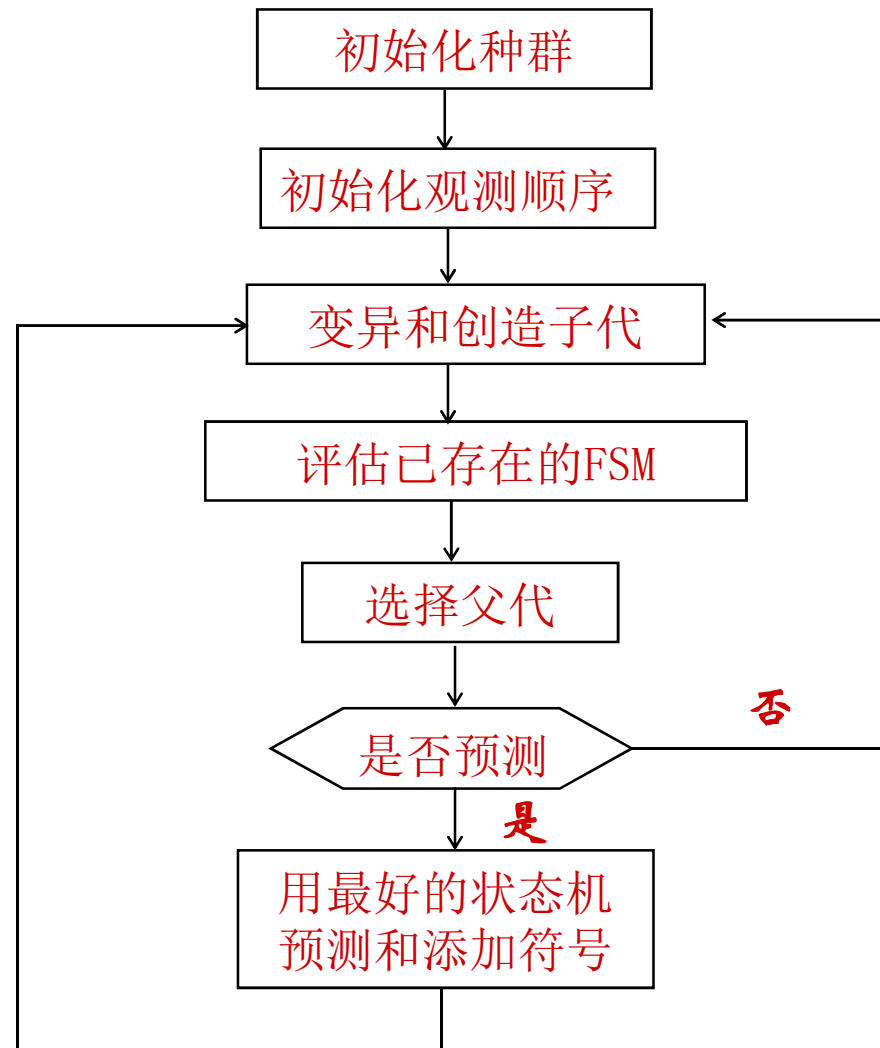
- 进化编程的过程，可理解为从所有可能的计算机程序形成的空间中，搜索具有高的适应度的计算机程序个体。
- 进化编程设计强调种群行为的变化。进化编程系统的表示自然地面向任务级。一旦选定一种适应性表示，就可以定义依赖于该表示的变异操作，在具体的父辈行为上创建后代。

进化编程的步骤

进化编程分为三个步骤：

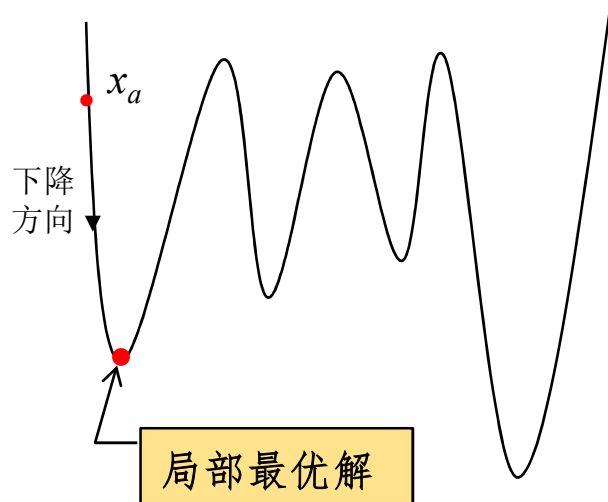
- 产生出初始种群。
- 迭代完成下述子步骤，直至满足选种标准为止：
 - 执行种群中的每个程序。
 - 应用变异等操作创造新程序种群。
- 在后代中适应值最高的计算机程序个体被指定为进化编程的结果。

进化编程的基本过程

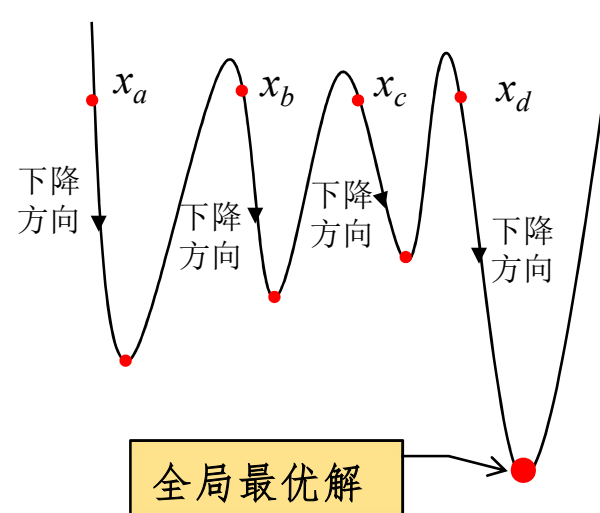


进化算法与全局最优解

□ 进化算法采用多点（群）寻优，更可能求得全局最优解



基于梯度的优化方法



进化算法的优化方法

目录

- 进化计算
 - 遗传算法
 - 人工生命
 - 粒子群优化算法
 - 蚁群算法
 - 小结
-

遗传算法概述

- ❑ 美国J. Holland教授于1975年在专著《自然界和人工系统的适应性》中首先提出。
 - ❑ 借鉴生物界自然选择和自然遗传机制的随机化搜索算法。
 - ❑ 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。
-

遗传算法思想

- 遗传算法是模仿生物遗传学和自然选择机理，通过人工方式所构造的一类优化搜索算法，是对生物进化过程进行的一种数学仿真，是进化计算的最重要的形式。
- 遗传算法为那些难以找到传统数学模型的难题指出了解决方法。
- 进化计算和遗传算法借鉴了生物学中的某些知识，这也体现了人工智能这一交叉学科的特点。

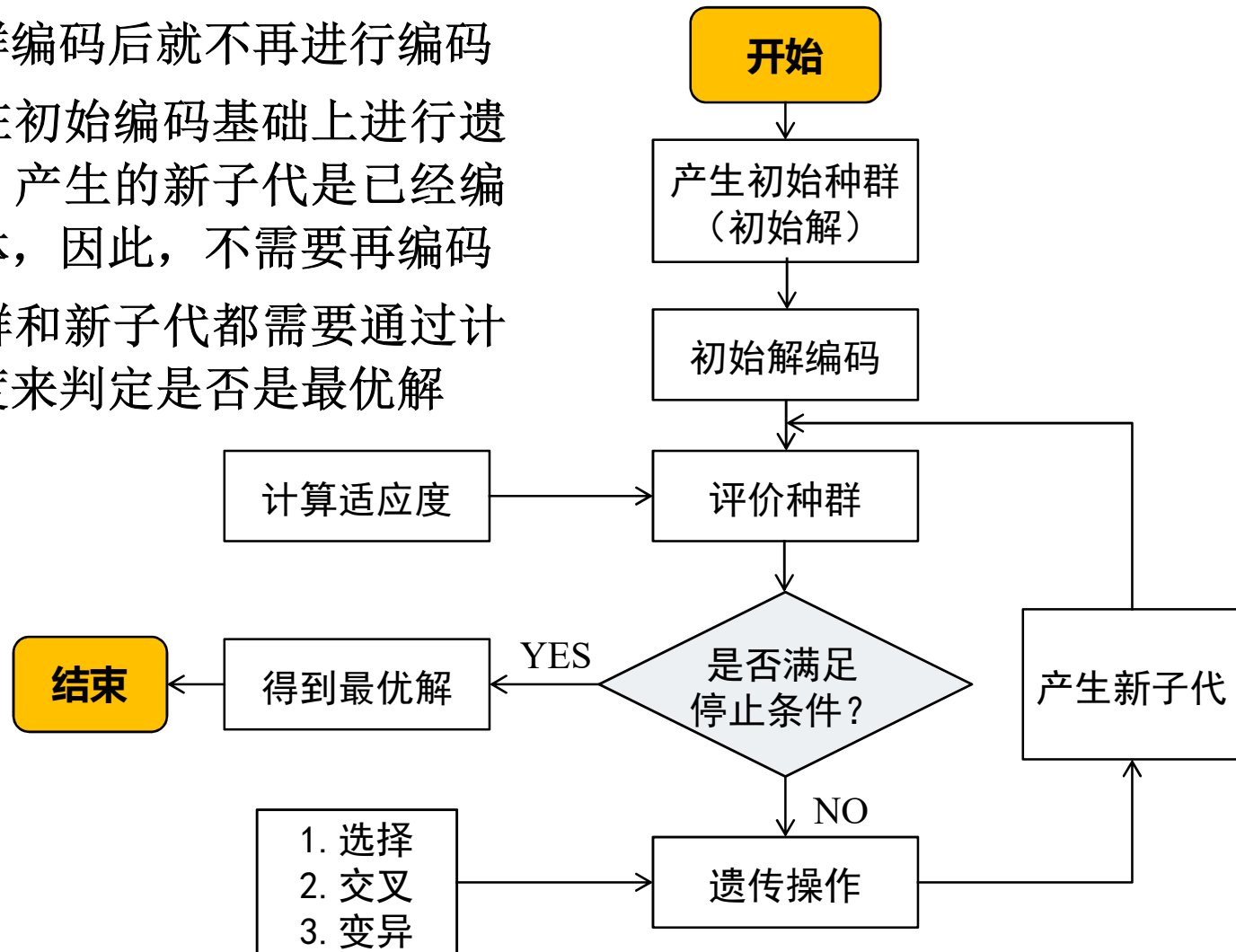
遗传算法的基本机理

- 霍兰德的遗传算法通常称为简单遗传算法（SGA）。现以此作为讨论主要对象，加上适应的改进，来分析遗传算法的结构和机理。
- 在遗传算法中，一个个体一般只包含一条染色体。染色体上包含这一组基因组。

有些情况下，**一个个体 = 一条染色体 = 一个基因**

遗传算法流程

- 初始种群编码后就不再进行编码
- 子代是在初始编码基础上进行遗传操作，产生的新子代是已经编码的个体，因此，不需要再编码
- 初始种群和新子代都需要通过计算适应度来判定是否是最优解



1. 染色体编码与解码

- 将问题结构变换为位串形式编码表示的过程叫编码；而相反将位串形式编码表示变换为原问题结构的过程叫解码或译码。把位串形式编码表示叫染色体。
- 遗传算法的编码方法有
 - 二进制编码
 - 浮点数编码方法
 - 格雷码
 - 符号编码方法
 - 多参数编码方法
 - 等等。

二进制编码

□ 最常用的编码方法是二进制编码

■ 假设优化目标为

$$\max f(x_1, x_2) = x_1^2 + x_2^2$$

$$\text{s.t. } x_1, x_2 \in [1, 2, 3, 4, 5, 6, 7]$$

■ 那么我们将其中一个解 $x_1 = 5$, $x_2 = 6$ 作为一个染色个体，将5的二进制101和6的二进制110连接起来为101110

■ 解码是将二进制编码转化为十进制，即将101110分解为101和110，然后再采用二进制和十进制的转码算法转换为5和6

符号编码

- 二进制编码的最大**缺点**之一是长度较大，对很多问题用其他主编码方法可能更有利
- **符号编码方法**是指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。
- 例如，对于TSP问题，采用符号编码方法，按一条回路中城市的次序进行编码，一般情况是从城市 w_1 开始，依次经过城市 w_2, \dots, 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

2. 适应度函数

- 体现染色体的适应能力，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 (fitness function)
- 对优化问题，适应度函数就是目标函数。TSP的目标是路径总长度为最短，路径总长度倒数可作为TSP问题的适应度函数：

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

路径总长度的倒数，
越大距离越短，越
好！最大化此函数

3. 遗传操作

□ 简单遗传算法的遗传操作主要有三种:

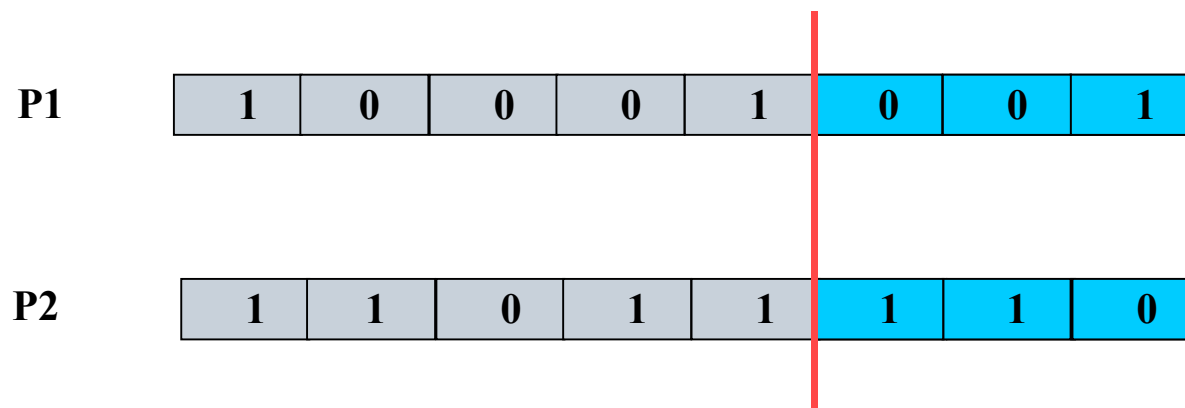
- 选择 (selection)
 - 交叉 (crossover)
 - 变异 (mutation)
-

3. 遗传操作：选择操作

- **选择操作**也叫复制（reproduction）操作，根据个体的适应度函数值所度量的优劣程度决定它在下一代是被淘汰还是被遗传。
- 一般地说，选择将使适应度较大（优良）个体有较大的存在机会，而适应度较小（低劣）的个体继续存在的机会也较小。（**适者生存**）

3. 遗传操作：交叉操作

- **交叉操作**的简单方式是将被选择出的两个个体P1和P2作为父母个体，将两者的部分码值进行交换
- 假设有八位长的二个体，产生一个在1到8之间的随机数 c ，假如现在产生的是3，将P1和P2的低三位交换



3. 遗传操作：变异操作

- **变异操作**的简单方式是改变数码串的某个位置上的数码
- 二进制编码表示的简单变异操作是将0与1互换：0变异为1，1变异为0
- TSP的变异操作：随机产生一个1至 n 之间的数 k ，对回路中的第 k 个城市的代码 w_k 作变异操作，又产生一个1至 n 之间的数 w ，替代 w_k ，并将 w_k 加到尾部，得到：

$$w_1 w_2 \dots w_{k-1} w w_{k+1} \dots w_n w_k$$

举例

- 利用GA算法求解下述函数的最大值

$$\max f(x_1, x_2) = x_1^2 + x_2^2$$

$$\text{s.t. } x_1 \in [1, 2, 3, 4, 5, 6, 7]$$

$$x_2 \in [1, 2, 3, 4, 5, 6, 7]$$

- 显然这是一个最优化问题，适合采用GA算法
 - 采用基本的数学知识可以直接求得上述函数值，即当 x_1 , $x_2=7$ 时, $f(x_1, x_2)$ 最大值为98
 - 这里为了说明问题，采用简单的例子便于分析理解
-

(1) 编码与解码

□ 按照GA算法，首先要对个体进行编码

- 把变量 x_1 , x_2 编码为一种符号串，这里用无符号二进制整数来表示
 - 由于 $1=001$, $7=111$ ，所以用3位无符号二进制整数来表示 x_1 和 x_2 ，并把它们连接起来
 - 例如， $x_1=5$, $x_2=6$ 可表示为基因型 $X=101110$
 - 个体的表现型 x 和基因型 X 之间可通过编码和解码程序相互转换
-

(2) 初始群体的产生

- ❑ GA算法是对群体进行的进化操作，需要给其准备一些表示起始搜索点的初始群体数据。
 - ❑ 本例中，**群体规模**的大小取为4，即群体由4个个体组成
 - ❑ **初始群体**的每个个体可通过**随机方法产生**。
如：011101，101011，011100，111001
 - ❑ 随机产生有很多好处，如可能避免陷入局部最优问题，参考梯度下降法
 - ❑ **遗传算法就是对初始群体进行操作，然后逐步迭代下去，直至找到最优解**
-

(3) 适应度计算

- ❑ 遗传算法中以个体适应度的大小来评定各个个体的优劣程度，从而决定其遗传机会的大小。
 - ❑ 本例中，目标函数总取非负值，并且是以求函数最大值为优化目标，故可直接利用**目标函数值**作为个体的**适应度**。
 - ❑ 适应度函数
$$f(x_1, x_2) = x_1^2 + x_2^2$$
 - ❑ 适应度越大越好，小的可能会被淘汰！
-

(4) 选择运算（又称自然选择）

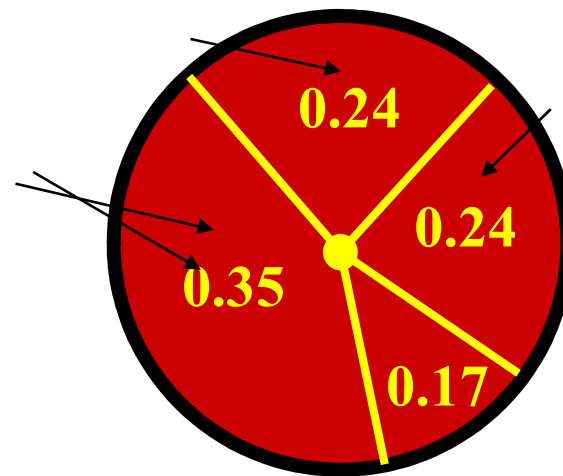
- 一般要求适应度较高的个体将有更多的机会遗传到下一代群体中去
 - 本例中，采用与适应度成正比的概率来确定各个个体复制到下一代群体中的数量。（轮盘赌法）
 - 其具体操作过程是：
 - 先计算出群体中所有个体的适应度的总和 $\sum f_i (i = 1, 2, \dots, M)$;
 - 其次计算出每个个体的相对适应度的大小 $f_i / \sum f_i$ ，它即为每个个体被遗传到下一代群体中的概率
 - 每个概率值组成一个区域，全部概率值之和为1;
 - 最后再产生一个0到1之间的随机数（就是小数），依据该随机数出现在上述哪一个概率区域内来确定各个个体被选中的次数。
-

(4) 选择运算 Cont.

□ 计算过程和结果如下表

个体编号	初始群体	x_1	x_2	适应度值	占比	选择次数	选择结果
1	011101	3	5	34	0.24	1	011101
2	101011	5	3	34	0.24	1	111001
3	011100	3	4	25	0.17	0	101011
4	111001	7	1	50	0.35	2	111001
总和				143			

被淘汰



轮盘赌选择方法

(5) 交叉运算

- 交叉运算（又称繁殖，交配，杂交）是遗传算法中产生新个体的主要操作过程，它以某一概率（交叉概率，人定）相互交换某两个个体之间的部分染色体。
- 其具体操作过程是：
 - 先对群体进行随机配对；
 - 其次随机设置交叉点位置；
 - 最后再相互交换配对染色体之间的部分基因。

个体编号	选择结果	配对情况	交叉点位置	交叉结果
1	01 1101	1, 2	1, 2: 2	011001
2	11 1001			111101
3	1010 11	3, 4	3, 4: 4	111001
4	1110 01			111011

可见，新生成的111101、111011的适应值比原来的两个个体都要高！

(6) 变异运算（又称突变）

- 变异运算是对个体的某一个或某一些基因座上的基因值按某一较小的概率进行改变，它也是产生新个体的一种操作方法。
- 其具体操作过程是：
 - 首先确定出各个个体的基因变异位置，下表所示为随机产生的变异点位置，其中的数字表示变异点设置在该基因座处；
 - 然后依照某一概率（**变异概率**，人定）将变异点的原有基因值取反。

个体编号	交叉结果	变异点	变异结果	子代群体
1	011001	4	011101	011101
2	111101	5	111111	111111
3	111001	2	111001	111001
4	111011	6	111010	111010
已经出现最优子代111111				

遗传算法的特点

□ 遗传算法的特点

- 遗传算法是对参数集合的编码而非针对参数本身进行进化；
- 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。

GA与进化计算的发展

- 进化计算 (Evolutionary computing)
- 灵感计算 (inspired computing)
- 自然计算 (Nature computing)
 - 进化计算
 - 蚁群系统
 - 量子遗传算法
 - 人工免疫系统
 - 人工内分泌系统
 - 复杂自适应系统

目录

- 进化计算
 - 遗传算法
 - 人工生命
 - 粒子群优化算法
 - 蚁群算法
 - 小结
-

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**（Artificial Life, AL）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。**1943**年麦卡络奇和皮茨提出了**M—P**神经学网络模型。
- ❑ 人工生命的许多早期研究工作也源于人工智能。
- ❑ **20**世纪**70**年代以来，康拉德（**Conrad**）等提出不断完善的“人工世界”模型。
- ❑ **80**年代
- ❑ **90**年代

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。
- 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，却需要对问题进行深入研究。

人工生命系统

- 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

- 自繁殖、自进化、自寻优
- 自成长、自学习、自组织
- 自稳定、自适应、自协调
- 物质构造
- 能量转换
- 信息处理

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。**
 - 开发基于人工生命的工程技术新方法、新系统、新产品
 - 为自然生命的研究提供新模型、新工具、新环境
 - 延伸人类寿命、减缓衰老、防治疾病
 - 扩展自然生命，人工进化、优生优育
 - 促进生命、信息、系统科学的交叉与发展

人工生命的研究内容

□ 人工生命的研究内容大致可分为两类：

- （1）研究如何利用计算技术研究生物现象，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）研究如何利用生物技术研究计算问题，包括环境适应系统和遗传进化系统等。

□ 我们关注的是第二点。

- 已有很多源于生物现象的计算技巧，例如神经网络和遗传算法。
 - 基于群体的智能，如粒子群算法、蚂蚁算法等。
-

人工生命的科学框架

- ❑ 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
- ❑ 进化和学习等的结合
- ❑ 人工生命的应用

人工生命的研究方法

□ 信息模型法

- 根据内部和外部系统所表现的生命行为来建造信息模型。

□ 工作原理法

- 生命行为所显示的自律分数和非线性行为，其工作原理是混沌和分形，以此为基础研究人工生命的机理。
-

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

- 波兰人工智能和心理学教授安奇·布勒（**Andrzej Buller**）及一些日本学者在日本现代通讯研究所进化系统研究室对人工脑的研究，已取得重要进展。

□ 计算机病毒

□ 计算机进程

□ 细胞自动机

□ 人工核苷酸

目录

- 进化计算
 - 遗传算法
 - 人工生命
 - 粒子群优化算法
 - 蚁群算法
 - 小结
-

社会系统：群体智能

- 已有很多源于生物现象的计算技巧，例如神经网络和遗传算法。
 - 现在讨论另一种生物系统——**社会系统**：由简单个体组成的群落和环境及个体之间的相互行为。
 - **群体智能**(swarm intelligence)
 - 模拟系统利用**局部信息**从而可以产生不可预测的**群行为**。
 - 成群的鸟、鱼或者浮游生物的聚集行为有利于它们觅食和逃避捕食者。它们的群落动辄以十、百、千甚至万计，并且经常不存在一个统一的指挥者。它们是如何完成聚集、移动这些功能呢？
-

群体智能五原则

□ Millonas在开发人工生命算法时(1994年)，提出群体智能概念并提出五点原则：

- 1、接近性原则：群体应能够实现简单的时空计算；
 - 2、优质性原则：群体能够响应环境要素；
 - 3、变化相应原则：群体不应把自己的活动限制在一狭小范围；
 - 4、稳定性原则：群体不应每次随环境改变自己的模式；
 - 5、适应性原则：群体的模式应在计算代价值得的时候改变。
-

社会组织

- 社会组织的全局群行为是由群内个体行为以**非线性**方式出现的。个体间的交互作用在构建群行为中起到重要的作用。
 - 从不同的群研究得到不同的应用。最引人注目的是对**蚁群**和**鸟群**的研究。
 - 其中**粒群优化方法**就是**模拟鸟群**的社会行为发展而来。
-

鸟群行为模拟

□ 对鸟群行为的模拟：

- Reynolds、Heppner和Grenader提出鸟群行为的模拟。他们发现，鸟群在行进中会突然同步的改变方向，散开或者聚集等。那么一定有某种潜在的能力或规则保证了这些同步的行为。这些科学家都认为上述行为是基于不可预知的鸟类社会行为中的群体动态学。
 - 在这些早期的模型中仅仅依赖个体间距的操作，也就是说，这种**同步**是鸟群中个体之间努力保持**最优的距离的结果**。
-

鱼群行为

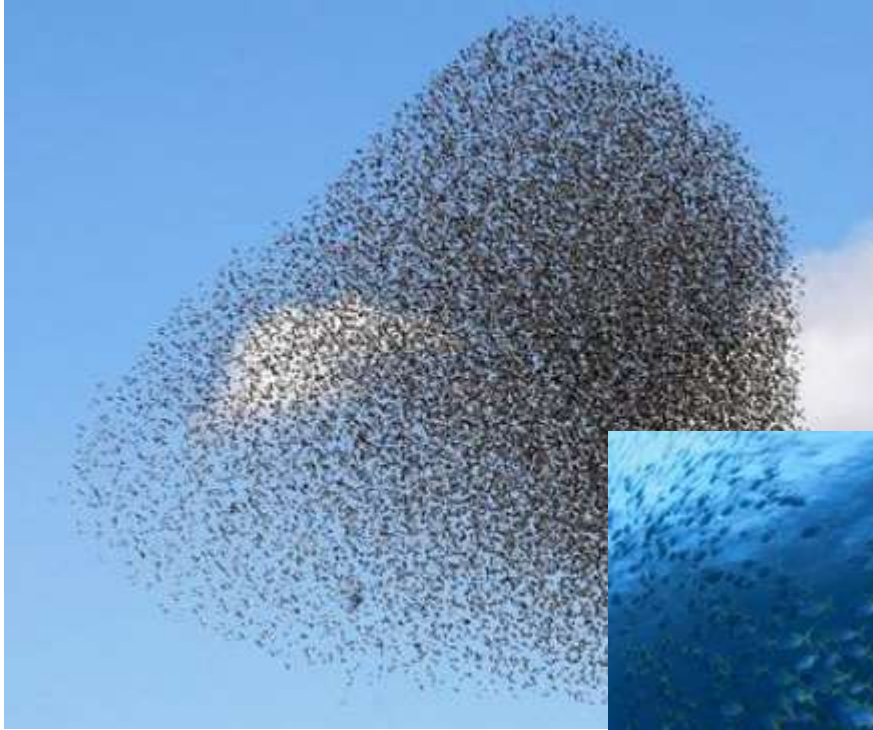
□ 对鱼群行为的研究：

■ 生物社会学家 **E.O.Wilson** 对鱼群进行了研究。提出：

□ “至少在理论上，鱼群的个体成员能够受益于群体中其他个体在寻找食物的过程中的发现和以前的经验，这种受益超过了个体之间的竞争所带来的利益消耗，不管任何时候食物资源不可预知的分散。”

■ 这说明，同种生物之间**信息**的**社会共享**能够带来好处。**这是粒子群算法的基础。**

鸟群与鱼群



进入正题：粒子群算法

□ 粒子群优化算法 (Particle Swarm Optimization, PSO)

■ 是一种**进化计算技术** (evolutionary computation), 由Eberhart博士和Kennedy博士于1995年提出 (Kennedy J, Eberhart R. Particle swarm optimization. Proceedings of the IEEE International Conference on Neural Networks. 1995. 1942~1948.)。源于对鸟群捕食的行为研究。

■ 粒子群优化算法的基本思想:**通过群体中个体之间的协作和信息共享来寻找最优解.**

□ PSO的优势在于简单容易实现并且没有许多参数的调节。目前已被广泛应用于**函数优化、神经网络训练、模糊系统控制**以及其他**遗传算法**的应用领域。

粒子群算法介绍

□ 设想这样一个场景:



- 一群鸟在随机的搜索食物。在这个区域里只有一块食物，所有的鸟都不知道食物在那。但是它们知道自己当前的位置距离食物还有多远。

□ 那么找到食物的最优策略是什么？

- 最简单有效的就是：**搜寻**目前离食物最近的鸟的**周围区域**。
-

算法描述

□ 抽象:

- 鸟被抽象为没有质量和体积的**微粒(点)**，并延伸到N维空间，粒子 i 在N维空间的**位置**表示为矢量 $X_i=(x_1, x_2, \dots, x_N)$ ，飞行**速度**表示为矢量 $V_i=(v_1, v_2, \dots, v_N)$ 
- 每个粒子都有一个由**目标函数**决定的**适应值(fitness value)**，并且知道**自己**到目前为止发现的**最好位置(pbest)**和现在的位置 X_i 。这个可以看作是粒子自己的飞行经验。
- 除此之外，每个粒子还知道到目前为止**整个群体**中所有粒子发现的**最好位置(gbest)** ($gbest$ 是 $pbest$ 中的最好值)。这个可以看作是粒子同伴的经验。
- **粒子就是通过自己的经验和同伴中最好的经验来决定下一步的运动**

□ PSO初始化为一群随机粒子(**随机解, 类似GA中的初始群**)。然后通过迭代找到最优解。在每一次的迭代中, 粒子通过跟踪两个“极值” (*pbest*, *gbest*)来更新自己。

□ 在找到这两个最优值后, 粒子通过下面的公式来更新自己的速度和位置

□ (1) 式 (**更新速度**)

$$v_i(t) = v_i(t-1) + c_1 \times rand() \times (pbest_i - x_i) + c_2 \times rand() \times (gbest_i - x_i)$$

□ (2) 式 (**更新位置**)

$$x_i(t) = x_i(t-1) + v_i(t)$$

时间单位取1, 故 $\Delta t = 1$,
所以位移 $v_i(t) \times \Delta t = v_i(t)$

□ 上两式中, $i=1, 2, \dots, M$, M 是该群体中粒子的总数

□ 参数说明

- $v_i(t)$ 是粒子在 t 时刻的速度；
- $pbest$ 和 $gbest$ 如前定义；
- $rand()$ 是介于（0、1）之间的随机数；
- x_i 是粒子的当前位置。
- c_1 和 c_2 是学习因子，通常取 $c_1 = c_2 = 2$

□ 在每一维，粒子都有一个**最大限制速度** V_{max} ，如果某一维的速度超过设定的 V_{max} ，那么这一维的速度就被限定为 V_{max} 。（ $V_{max} > 0$ ）

□ 以上面两个公式为基础，形成了后来PSO的标准形式

$$v_i(t)=v_i(t-1)+c_1\times rand()\times(pbest_i - x_i)+c_2\times rand()\times(gbest_i - x_i)$$

□ 从社会学的角度来看，公式(1)的

- 第一部分称为**记忆项**，表示上次速度大小和方向的影响；
- 第二部分称为**自身认知项**，是从当前点指向粒子自身最好点的一个**矢量**，表示粒子的动作来源于自己经验的部分；
- 第三部分称为**群体认知项**，是一个从当前点指向种群最好点的**矢量**，反映了粒子间的协同合作和知识共享。

□ 粒子就是通过自己的经验和同伴中最好的经验来决定下一步的运动

□ 1998年Shi等人在进化计算的国际会议上发表了一篇论文《A modified particle swarm optimizer》对前面的公式(1)进行了修正。引入**惯性权重因子**。

□ (3) 式

$$v_i(t) = \omega \times v_i(t-1) + c_1 \times rand() \times (pbest_i - x_i) + c_2 \times rand() \times (gbest_i - x_i)$$

ω 非负，称为**惯性因子**

□ 公式(2)和(3)被视为 **标准PSO算法**。

-
- ω 值较大，全局寻优能力强，局部寻优能力弱； ω 值较小反之。
 - 初始时，Shi等人将 ω 取为常数，后来实验发现，动态 ω 能够获得比固定值更好的寻优结果。动态 ω 可以在PSO搜索过程中线性变化，也可根据PSO性能的某个测度函数动态改变。
 - 目前，采用较多的是Shi建议的**线性递减权值**(linearly decreasing weight, LDW)策略
-

$$\omega^{(t)} = (\omega_{ini} - \omega_{end})(G_k - g) / G_k + \omega_{end}$$

□ G_k 为最大进化代数， ω_{ini} 为初始惯性权值， ω_{end} 为迭代至最大代数时惯性权值。

□ 典型取值

$$\omega_{Ini} = 0.9, \quad \omega_{end} = 0.4。$$

□ ω 的引入使PSO算法性能有了很大提高，针对不同的搜索问题，**可以调整全局和局部搜索能力**，也使得PSO算法能成功的应用于很多实际问题

标准PSO算法描述

□ 标准PSO算法的流程：

Step1: 初始化一群微粒(群体规模为 m)，包括随机位置和速度；

Step2: 评价每个微粒的适应度；

Step3: 对每个微粒，将其适应值与其经过的最好位置 $pbest$ 作比较，如果较好，则将其作为当前的最好位置 $pbest$ ；

Step4: 对每个微粒，将其适应值与其经过的最好位置 $gbest$ 作比较，如果较好，则将其作为当前的最好位置 $gbest$ ；

Step5: 根据(2)、(3)式调整微粒速度和位置；

Step6: 未达到结束条件则转Step2。

迭代终止条件根据具体问题一般选为最大迭代次数 G_k 或(和)微粒群迄今为止搜索到的最优位置满足预定最小适应阈值

局部和全局最优算法

- 方程(2)和(3)中 $pbest$ 和 $gbest$ 分别表示微粒群的局部和全局最优位置，当 $c_1 = 0$ 时，则粒子没有了自我（*Personal*）认知能力，变为只有社会的模型(*social-only*):

$$v_i(t) = \omega \times v_i(t-1) + c_2 \times rand() \times (gbest_i - x_i)$$

- 此时，被称为**全局PSO算法**。粒子有扩展搜索空间的能力，具有较快的收敛速度，但由于缺少局部搜索，对于复杂问题比标准PSO **更易陷入局部最优**

-
- 当 $c_2=0$ 时，则粒子之间没有社会信息，模型变为**只有认知**(*Cognition-Only*)模型：

$$v_i(t) = \omega \times v_i(t-1) + c_1 \times rand() \times (pbest_i - x_i)$$

- 此时，被称为**局部PSO算法**。由于个体之间没有信息的交流，整个群体相当于多个粒子进行盲目的随机搜索，**收敛速度慢**，因而得到最优解的可能性小
-

参数分析

- 参数有：群体规模 m ，惯性因子 ω ，学习因子 c_1 和 c_2 ，最大速度 V_{max} ，迭代次数 G_k 。
 - 群体规模 m 一般取20~40，对较难或特定类别的问题可以取到100~200。
 - 最大速度 V_{max} 决定当前位置与最好位置之间的区域的分辨率(或精度)。
 - 如果太快，则粒子有可能越过极小点；
 - 如果太慢，则粒子不能在局部极小点之外进行足够的探索，会陷入到局部极值区域内。
 - 这种限制可以达到防止计算溢出、决定问题空间搜索的粒度的目的
-

参数分析

□ **权重因子**：包括惯性因子 ω 和学习因子 c_1 和 c_2

- ω 使粒子保持着运动惯性，使其具有扩展搜索空间的趋势，有能力探索新的区域。
 - c_1 和 c_2 代表将每个粒子推向 $pbest$ 和 $gbest$ 位置的统计加速项的权值。
 - 较低的值允许粒子在被拉回之前可以在目标区域外徘徊
 - 较高的值导致粒子突然地冲向或越过目标区域
-

参数分析

- **参数设置**: 如果令 $c_1 = c_2 = 0$, 粒子将一直以当前速度的飞行, 直到边界。很难找到最优解。
 - 如果 $\omega = 0$, 则速度只取决于当前位置和历史最好位置, 速度本身没有记忆性。假设一个粒子处在全局最好位置, 它将保持静止, 其他粒子则飞向它的最好位置和全局最好位置的加权中心。粒子将收缩到当前全局最好位置。
 - 在加上第一部分后, 粒子有扩展搜索空间的趋势, 这也使得 ω 的作用表现为针对不同的搜索问题, 调整算法的全局和局部搜索能力的平衡。
 - ω 较大时, 具有较强的全局搜索能力;
 - ω 较小时, 具有较强的局部搜索能力
-

参数分析

- 通常设 $c_1=c_2=2$ 。Suganthan的实验表明： c_1 和 c_2 为常数时可以得到较好的解，但不一定必须等于2。Clerc引入收敛因子(**constriction factor**) K 来保证收敛性。

$$v_i(t) = K[v_i(t-1) + \varphi_1 \times rand() \times (pbest_i - x_i) + \varphi_2 \times rand() \times (gbest_i - x_i)]$$

- 其中

$$K = \frac{2}{\left| 2 - \varphi - \sqrt{\varphi^2 - 4\varphi} \right|}, \varphi = \varphi_1 + \varphi_2, \varphi > 4.$$

参数分析

- 通常取 ω 为4.1，则 $K=0.729$ 。
 - 实验表明，与使用惯性权重的PSO算法相比，使用收敛因子的PSO有更快的收敛速度。其实只要恰当的选取 ω 和 c_1 、 c_2 ，两种算法是一样的。
 - 因此使用收敛因子的PSO可以看作使用惯性权重PSO的特例。
 - 恰当的选取算法的参数值可以改善算法的性能
-

PSO与其他算法比较

□ 与遗传算法比较

□ 共性:

- 都属于仿生算法。
- 都属于全局优化方法。
- 都属于随机搜索算法。
- 都隐含并行性。
- 根据个体的适配信息进行搜索，因此不受函数约束条件的限制，如连续性、可导性等。
- 对高维复杂问题，往往会遇到早熟收敛和收敛性能差的缺点，都无法保证收敛到最优点

□ 差异:

- PSO有记忆，好的解的知识所有粒子都保存，而GA的知识随着种群的改变而改变。
 - PSO中的粒子仅仅通过当前搜索到最优点进行共享信息，所以这是一种单共享项信息机制。而GA中，染色体之间相互共享信息，使得整个种群都向最优区域移动。
 - PSO相对于GA，没有交叉和变异操作，粒子通过内部速度进行更新，原理更简单、参数更少、实现更容易。
-

□ PSO与ANN

□ GA可以用来研究NN的三个方面：

- 网络连接权重
- 网络结构
- 学习算法

已有利用**PSO**来进行神经网络训练。研究表明**PSO**是一种很有潜力的神经网络算法。速度较快且有较好的结果。且没有遗传算法碰到的问题

□ 优势：

- 可处理传统方法不能处理的问题，例如不可导的节点传递函数或没有梯度信息。

□ 缺点：

- 在某些问题上性能不是特别好；网络权重的编码和遗传算子的选择有时较麻烦。
-

目录

- 进化计算
 - 遗传算法
 - 人工生命
 - 粒子群优化算法
 - 蚁群算法
 - 小结
-

蚁群算法

□ 蚁群算法的提出:

■ 1992年，意大利学者M. Dorigo在其博士论文中提出**蚂蚁系统**（Ant System）。

□ 近年来，M. Dorigo等人进一步将蚂蚁算法发展为一种通用的优化技术——**蚁群优化**（Ant Colony Optimization, ACO）

□ **蚁群算法**是**群体智能**理论研究领域的一种主要算法

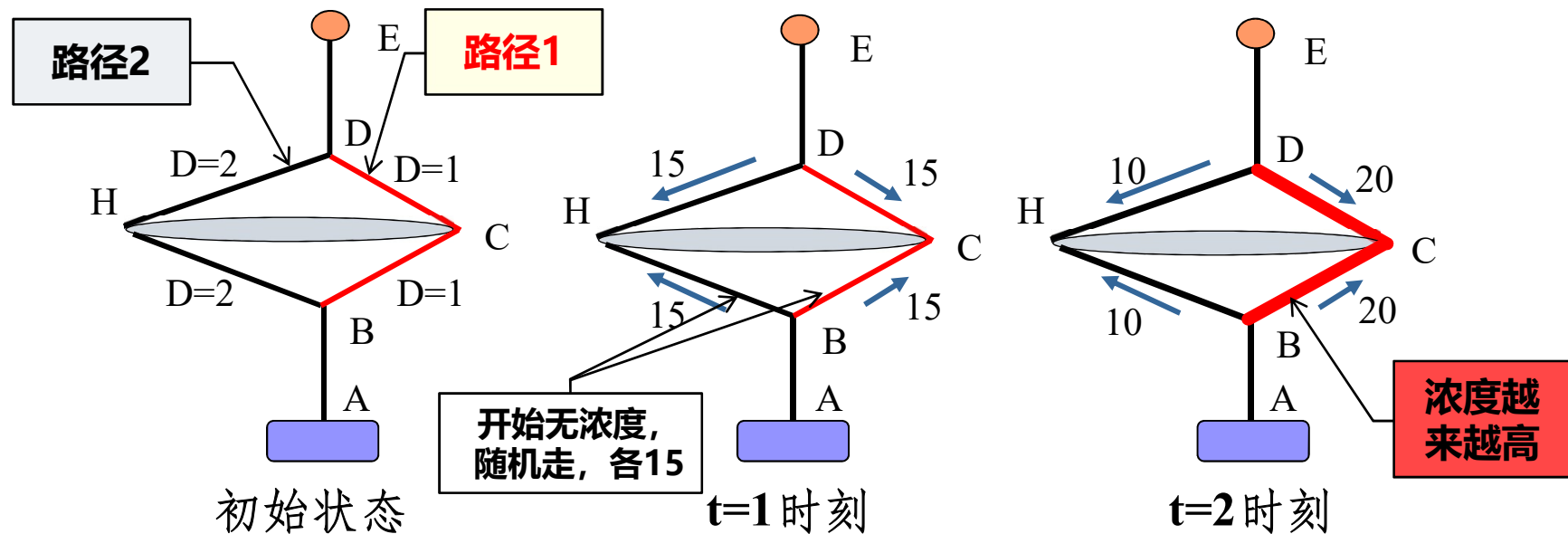
■ 被用于解决大多数**优化**问题或者能够转化为优化求解的问题。



ACO基本原理

□ 蚁群的集体行为表现出一种(外激素)信息正反馈现象:

- 某一路径上走过的蚂蚁越多(外激素的**浓度**就越高), 则后来者选择该路径的概率就越大。蚂蚁个体之间通过信息的交流(**嗅浓度**)达到搜索食物的目的。激素停留时间为1单位



应用例子-蚁群TSP系统模型

□ 模拟实际蚂蚁的行为：

- m 表示蚁群中蚂蚁的数量； $m = \sum_{i=1}^n b_i(t)$
- n 表示城市的数量；
- $b_i(t)$ 表示 t 时刻位于城市 i 的蚂蚁个数；
- $d_{ij} (i=1, 2, \dots, n)$ 表示城市 i 和城市 j 之间的距离；
- τ_{ij} 表示 t 时刻在 ij 连线上残留的信息量。

- 在初始时刻，设 $\tau_{ij}(0) = C$ (C 为常数)，各条路径上信息量相等。蚂蚁 k ($k = 1, 2, \dots, m$) 在运动过程中，根据各条路径上的信息量决定转移方向。

□ $p_{ij}^k(t)$ 表示在 t 时刻蚂蚁由位置 i 转移到位置 j 的概率

$$p_{ij}^k(t) = \frac{\tau_{ij}^{\alpha} \eta_{ij}^{\beta}(t)}{\sum_{s \in allowed_k} \tau_{is}^{\alpha} \eta_{is}^{\beta}(t)}$$

□ 其中,

- $allowed_k = \{0, 1, \dots, n-1\}$ 表示蚂蚁 k 下一步允许选择的城市。
 - τ_{ij} 表示 t 时刻在城市 i 、 j 连线上残留的信息量。
 - $\eta_{ij}^{\beta}(t)$ 表示由城市 i 转移到城市 j 的期望程度
 - α, β 分别表示蚂蚁在运动过程中所积累的信息及启发式因子在蚂蚁路径选择中所起的不同作用。
-

-
- 经过 n 个时刻，蚂蚁完成一次循环。各路径上信息量要根据下式作调整：

$$\tau_{ij}(t+n) = \tau_{ij} + \Delta\tau_{ij} = \tau_{ij} + \sum_{k=1}^n \Delta\tau_{ij}^k$$

τ_{ij} 表示第 k 只蚂蚁在本次循环中留在路径 ij 上的信息量；

$\Delta\tau_{ij}$ 表示本次循环中留在路径 ij 上的信息量：

在初始时刻， $\tau_{ij}(0) = C(\text{const})$ ， $\Delta\tau_{ij} = 0$ ，其中 $ij = 0, 1, \dots, n-1$ 。

- 与真实蚁群系统不同，人工蚁群系统具有一定的记忆功能，这里用 tabu_k ($k = 1, 2, \dots, m$)来记录蚂蚁 k 目前已经走过的城市。随着时间的推移，以前留下的信息逐渐消逝，用参数 $(1-p)$ 表示信息消逝的程度
-

应用

- **蚁群算法**源于对自然界中的蚂蚁寻找蚁巢到食物以及食物回到蚁巢的最短路径方法的研究。它是一种并行算法，所有“蚂蚁”均独立行动，没有监督机构。
 - **蚁群算法**又是一种合作算法，依靠群体行为进行寻优；
 - **蚁群算法**还是一种鲁棒算法，只要对算法稍作修改，就可以求解其他组合优化问题。
 - 二次分配问题
 - 作业调度问题
 - 电话网络路径选择
 - 数据通信网络的路由优化
 -
-

目录

- 进化计算
 - 遗传算法
 - 人工生命
 - 粒子群优化算法
 - 蚁群算法
 - 小结
-

小结

- **进化计算遵循自然界优胜劣汰、适者生存的进化准则，模仿生物群体的进化机制，并被用于处理计算复杂系统的优化问题**
 - 遗传算法：模仿生物遗传学和自然选择机理，通过人工方式构造的一类搜索算法
 - 粒子群优化算法：模拟鸟群社会的共生合作，是一种基于群体搜索的算法
 - 蚁群算法：研究蚂蚁寻找食物路径的自然行为，用于求解**TSP**问题、分配问题和调度问题，在复杂优化问题特别是离散优化问题的方面优势明显