



Mathematical Modeling

数学建模

同济大学数学科学学院

第十六章

遗传算法

目录/Contents | 第十六章 遗传算法

第一节 遗传算法基本概念

第二节 遗传算法基本操作

第三节 遗传算法应用



- 达尔文(1859) -- 物竞天择, 适者生存
- John Holland(University of Michigan, 1975) \\
Adaptation in Natural and Artificial System
- 遗传算法作为一种有效的工具, 已广泛地应用于最优化问题求解之中。
- 遗传算法是一种基于自然群体遗传进化机制的自适应全局优化概率搜索算法。它摒弃了传统的搜索方式, 模拟自然界生物进化过程, 采用人工的方式对目标空间进行随机化搜索。



- 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。
- **进化规则**: 适者生存, 较好的解保留, 较差的解淘汰



生物进化	遗传算法
环境	适应函数
适者生存	适应函数值大的保留概率较大
个体	问题的一个解
染色体	解的编码
基因	编码的元素
种群	选定的一组解
交叉	以一定的方式由双亲产生后代
变异	编码的某些分量发生随机变化

目录/Contents | 第十六章 遗传算法

第一节 遗传算法基本概念

第二节 遗传算法基本操作

第三节 遗传算法应用

- **选择(selection)**：根据各个个体的适应值, 按照一定的规则或方法, 从第 t 代群体 $P(t)$ 中选出一些优良的个体遗传到下一代群体 $P(t+1)$ 中。
- **交叉(crossover)**：将群体 $P(t)$ 内的各个个体随机搭配成对, 对每一个个体, 以某个概率 P_c (称为交叉概率, crossover rate) 交换它们之间的部分染色体。
- **变异(mutation)**：对群体 $P(t)$ 中的每一个个体, 以某一概率 P_m (称为变异概率, mutation rate) 改变某一个或一些基因上的基因值为其它的等位基因。



遗传算法设计

- 如何进行编码?
- 如何产生初始种群?
- 如何定义适应函数?
- 如何进行遗传操作(复制、交叉、变异)?
- 如何产生下一代种群?
- 如何定义停止准则?

**编码**

- 表现型空间
- 基因型空间
- 例：变量 $x = (x_1, x_2, \dots, x_n)^T$ ，每个分量 $x_i \in [0, 2^L)$ 为整数，则一个解可以表示为长度为 nL 的二进制编码
(100100011001011100...1101)

编码 (coding), 解码 (decoding)

适应函数

- 遗传算法在搜索中不依靠外部信息, 仅以适应函数为依据, 利用群体中每个染色体(个体)的适应值来进行搜索。以染色体适应值的大小来确定该染色体被遗传到下一代群体中的概率。染色体适应值越大, 该染色体被遗传到下一代的概率也越大; 反之, 染色体的适应值越小, 该染色体被遗传到下一代的概率也越小。

因此适应函数的选取至关重要, 直接影响到GA(Genetic Algorithm)的收敛速度以及能否找到最优解。

- 群体中的每个染色体都需要计算适应值。
- 适应函数一般由目标函数变换而成。

适应函数

➤ 直接将目标函数转化为适应函数

➤ 若目标函数为最大化问题:

$$Fitness(x) = f(x)$$

➤ 若目标函数为最小化问题:

$$Fitness(x) = -f(x)$$

➤ **缺点:** 可能不满足某些概率非负的要求;

➤ **缺点:** 某些代求解的函数值分布上相差很大, 由此得到的评价适应值可能不利于体现群体的评价性能, 影响算法的性能。

**适应函数**

- 界限构造法
- 以目标函数为最大化问题为例

$$Fitness(x) = \begin{cases} f(x) - C_{\min}, & \text{if } f(x) > C_{\min}, \\ 0, & \text{others} \end{cases}$$

- C_{\min} 为 $f(x)$ 最小值的下界。
- **缺点:** 需要估计 C_{\min}



选择(Selection)

- **选择(复制)**操作把当前种群的染色体按与适应值成正比例的概率复制到新的种群中
- **主要思想**: 适应值较高的染色体有较大的选择(复制)机会
- 一种**实现方法**: **轮盘赌**选择 (Roulette wheel selection)
 - 将种群中所有染色体的适应值相加求总和, 染色体适应值按其比例转化为选择概率 P_s
 - 产生一个在0与总和之间的随机数 m
 - 从种群中编号为 1 的染色体开始, 将其适应值与后续染色体的适应值相加, 直到累加和等于或大于 m

选择(Selection)

求 $\max f(x) = \frac{x^2 + x}{2}$

染色体编号	1	2	3	4	5	6
染色体	01110	11000	00100	10010	01100	00011
适应度	105	300	10	171	78	6
被选概率	0.1567	0.4478	0.0149	0.2552	0.1164	0.0090
适应度累计	105	405	415	586	664	670

产生随机数，确定对应的染色体



选择(Selection)

➤ 模拟轮盘赌的算法

1. $r = \text{rand}(0,1), s=0, i=0;$
2. 如果 $s \geq r$, 则转 4;
3. $s = s + p(x_i), i = i + 1$, 转 2;
4. x_i 即为被选中的染色体, 输出;
5. 结束。



选择(Selection)

- 其他选择法:
 - 随机遍历抽样(Stochastic universal sampling)
 - 局部选择(Local selection)
 - 截断选择(Truncation selection)
 - 锦标赛选择(Tournament selection)
- **特点:** 选择操作得到的新的群体称为交配池, 交配池是当前代和下一代之间的中间群体, 其规模为初始群体规模。选择操作的作用效果是提高了群体的平均适应值(低适应值个体趋于淘汰, 高适应值个体趋于生存), 但这也损失了群体的**多样性**。选择操作没有产生新的个体, 群体中最好个体的适应值不会改变。
- **其它**



交叉(Crossover)

- 遗传交叉(杂交、交配、有性重组)操作发生在两个染色体之间, 由两个被称之为双亲的父代染色体, 经杂交以后, 产生两个具有双亲的部分基因的新的染色体, 从而检测搜索空间中新的点。
- **选择(复制)**操作每次作用在一个染色体上, 而交叉操作每次作用在从交配池中随机选取的两个个体上(交叉概率 P_c)。
- 交叉产生两个子染色体, 他们与其父代不同, 且彼此不同, 每个子染色体都带有双亲染色体的遗传基因。

单点交叉

- 在双亲的父代染色体中随机产生一个交叉点位置;
- 在交叉点位置分离双亲染色体;
- 互换交叉点位置右边的基因码产生两个子代染色体;
- 交叉概率 P_c , 一般范围为 (60%, 90%), 平均约 80%
- 例:

父代:	1101		11110000
	0000		00111111
子代:	0000		11110000
	1101		00111111



交叉和选择

- 单点交叉操作可以产生与父代染色体完全不同的子代染色体; 它不会改变父代染色体中相同的基因。但当双亲染色体相同时, 交叉操作是不起作用的。
- 假如交叉概率 $P_c = 50\%$, 则交配池中50%的染色体(一半染色体)将进行交叉操作, 余下的50%的染色体进行选择(复制)操作。
- 遗传算法利用选择和交叉操作可以产生具有更高平均适应值和更好染色体的群体。

变异(Mutation)

- 以变异概率 P_m 改变染色体的某一个基因, 当以二进制编码时, 变异的基因由0变成1, 或者由1变成0
- 变异概率 P_m 一般为种群规模的倒数或染色体长度的倒数, 或平均约1%-2%
- 例:
父代: 110100001110100101010001010101
子代: 110100000110100101010001011101
- 比起选择和交叉操作, 变异操作是算法中的次要操作, 但它在恢复群体中失去的多样性方面具有潜在的作用。



变异(Mutation)

- 在算法执行的开始阶段, 染色体中一个特定位上的值1可能与好的性能紧密联系, 即搜索空间中某些初始染色体在那个位上的值1可能一致产生高的适应值。因为越高的适应值与染色体中那个位上的值1相联系, 选择操作就越会使群体的遗传多样性损失。
- 等到达一定程度时, 值0会从整个群体中那个位上消失, 然而全局最优解可能在染色体中那个位上为0。如果搜索范围缩小到实际包含全局最优解的那部分搜索空间, 在那个位上的值0就可能正好是到达全局最优解所需要的。



停止准则

- 种群中个体的最大适应值超过预设值
- 种群中个体的平均适应值超过预设值
- 种群中个体的进化代数超过预设值



算法框架

1. 随机产生初始种群;
2. 计算种群中每个个体的适应度值, 判断是否满足停止条件, 若不满足, 则转第 3 步, 否则转第 6 步;
3. 按由个体适应值所决定的某个规则选择将进入下一代的个体;
4. 按交叉概率 P_c 进行交叉操作, 生产新的个体;
5. 按变异概率 P_m 进行变异操作, 生产新的个体;
6. 输出种群中适应度值最优的染色体作为问题的满意解或最优解。

目录/Contents | 第十六章 遗传算法

第一节 遗传算法基本概念

第二节 遗传算法基本操作

第三节 遗传算法应用



- **组合优化**: 遗传算法是寻求组合优化问题满意解的最佳工具之一。实践证明, 遗传算法对于组合优化问题中的NP完全问题非常有效。例如, 遗传算法已经在求解旅行商问题(Traveling Salesman Problem, TSP)、背包问题(Knapsack Problem)、装箱问题(Bin Packing Problem) 等方面得到成功的应用
- **生产调度问题**: 生产调度问题在很多情况下所建立起来的数学模型难以精确求解, 即使经过一些简化之后可以进行求解也会因简化得太多而使求解结果与实际相差太远。现在遗传算法已经成为解决复杂调度问题的有效工具
- 其它: **自动控制、人工智能、图像处理、模式识别、机器学习**

- TSP问题
- 100个城市对应 4.6663×10^{155} 条路径
- 编码：路径编码向量

(5,1,7,8,9,4,6,2,3)

直接表示一条路径

- 一般的交叉操作

123|45

543|21



123|21

543|45



➤ 选取两个交叉点，给出映射关系

➤ 例如

123456789

937826514



{4-8, 5-2, 7-5}

➤ 交换编码

xxx8265xx

xxx4567xx

➤ 继承编码

1x38265x9

93x45671x

➤ 映射编码

173826549

932456718



- 记取父个体从第二个交叉点开始城市码的排列顺序, 当到达表尾时, 返回表头继续记录, 直到第二个交叉点。
- 例如
123456789
937826514
- 子代
xxx4567xx
- 父代排列
9-3-7-8-2-6-5-1-4
- 删除已确定编码
9-3-8-2-1
- 复制
938456721



1. 找一个可以应用遗传算法的问题
2. 它可以使用其他方法吗
3. 有什么其他的现代优化算法



Mathematical Modeling 数学建模

同济大学数学科学学院

学海无涯，祝你成功！