

现代优化算法

李琛

Fudan Univerisity

lic.tiny@gmail.com

January 10, 2016

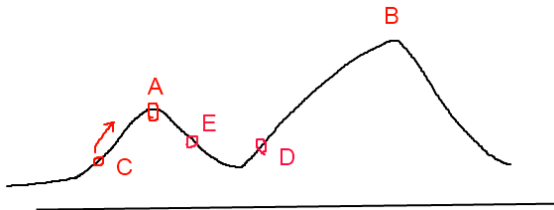
Overview

- 1 爬山法
- 2 模拟退火算法
- 3 遗传算法

爬山法

爬山算法是一种简单的贪心搜索算法，该算法每次从当前解的临近解空间选择一个最优解作为当前解，直到达到一个局部最优解。

- 局部最优
- 高地：无法确定搜索最佳方向，会产生随机走动，使得搜索效率降低
- 山脊：搜索可能会在山脊的两面来回震荡，前进步伐很小。



伪代码

```
1  currentNode = startNode;
2  loop do
3      L = NEIGHBORS(currentNode);
4      nextEval = -INF;
5      nextNode = NULL;
6      for all x in L
7          if (EVAL(x) > nextEval)
8              nextNode = x;
9              nextEval = EVAL(x);
10     if nextEval <= EVAL(currentNode)
11         //Return current node since no better neighbors exist
12         return currentNode;
13     currentNode = nextNode;
```

模拟退火算法

模拟退火算法 (Simulated Annealing, SA) 来源于固体退火原理, 将固体加温至充分高, 再让其徐徐冷却, 加温时, 固体内部粒子随温升变为无序状, 内能增大, 而徐徐冷却时粒子渐趋有序, 在每个温度都达到平衡态, 最后在常温时达到基态, 内能减为最小。

模拟退火算法是基于 Monte-Carlo 迭代求解策略的一种随机寻优算法, 其出发点是基于物理中固体物质的退火过程与一般组合优化问题之间的相似性。模拟退火算法从某一较高初温出发, 伴随温度参数的不断下降, 结合概率突跳特性在解空间中随机寻找目标函数的全局最优解, 即在局部最优解能概率性地跳出并最终趋于全局最优。

算法简介

假设材料在状态 i 下的能量为 $E(i)$ ，那么材料在温度 T 时从状态 i 进入状态 j 遵循如下规律：

- 如果 $E(j) \leq E(i)$ ，接受该转换
 - 如果 $E(j) > E(i)$ ，则以 $e^{\frac{E(i)-E(j)}{kT}}$ 的概率接受
- 其中 K 为玻尔兹曼常数， T 为当前温度

证明

在某一特定温度 T 下，经过充分转换，材料达到热平衡，此时材料处于状态 i 的概率满足玻尔兹曼分布

$$P_T(x = i) = \frac{e^{-\frac{E(i)}{KT}}}{\sum_{j \in S} e^{-\frac{E(j)}{KT}}}$$

显然

$$\lim_{T \rightarrow \infty} T \frac{e^{-\frac{E(i)}{KT}}}{\sum_{j \in S} e^{-\frac{E(j)}{KT}}} = \frac{1}{|S|}$$

其中 $|S|$ 表示集合 S 中的状态数量

证明

温度下降时

$$\begin{aligned}\lim_{T \rightarrow 0} T \frac{e^{-\frac{E(i) - E_{\min}}{KT}}}{\sum_{j \in S} e^{-\frac{E(j) - E_{\min}}{KT}}} &= \lim_{T \rightarrow 0} T \frac{e^{-\frac{E(i) - E_{\min}}{KT}}}{\sum_{j \in S_{\min}} e^{-\frac{E(j) - E_{\min}}{KT}} + \sum_{j \notin S_{\min}} e^{-\frac{E(j) - E_{\min}}{KT}}} \\ &= \lim_{T \rightarrow 0} T \frac{e^{-\frac{E(i) - E_{\min}}{KT}}}{\sum_{j \in S_{\min}} e^{-\frac{E(j) - E_{\min}}{KT}}} = \begin{cases} \frac{1}{|S_{\min}|} & \text{若 } i \in S_{\min} \\ 0 & \text{otherwise} \end{cases}\end{aligned}$$

上式表明，当温度很低时，材料会有很大概率进入最小状态

伪代码

```
1  def anneal(sol):
2      old_cost = cost(sol)
3      T = 1.0
4      T_min = 0.00001
5      alpha = 0.9
6      while T > T_min:
7          i = 1
8          while i <= 100:
9              new_sol = neighbor(sol)
10             new_cost = cost(new_sol)
11             ap = acceptance_probability(old_cost, new_cost, T)
12             if ap > random():
13                 sol = new_sol
14                 old_cost = new_cost
15             i += 1
16             T = T*alpha
17     return sol, cost
```

遗传算法

遗传算法 (GA , Genetic Algorithm) , 也称进化算法。遗传算法是受达尔文的进化论的启发, 借鉴生物进化过程而提出的一种启发式搜索算法。

简单说来就是, 繁殖过程会发生基因交叉 (Crossover) , 基因突变 (Mutation) , 适应度 (Fitness) 低的个体会被逐步淘汰, 而适应度高的个体会越来越多。那么经过 N 代的自然选择后, 保存下来的个体都是适应度很高的, 其中很可能包含史上产生的适应度最高的那个个体。

算法流程

编码

需要将问题的解编码成字符串的形式才能使用遗传算法。最简单的一种编码方式是二进制编码，即将问题的解编码成二进制位数组的形式。

初始种群

在开始遗传算法迭代过程之前，需要对种群进行初始化。设种群大小为 pop_size ，每个染色体或个体的长度为 chromo_size 。一般随机生成初始种群，但是如果知道种群的分布，也可以按照此分布来生成初始种群。假设生成的初始种群为 $(v_1, v_2, \dots, v_{\text{popsize}})$ 。

算法流程

选择

选择一些染色体来产生下一代。一种常用的选择策略是“比例选择”，也就是个体被选中的概率与其适应度函数值成正比。假设群体的个体总数是 M ，那么一个个体 X_i 被选中的概率为 $\frac{f(X_i)}{\sum_{i=1}^M f(X_i)}$ 。

比例选择实现算法就是所谓的“轮盘赌算法”(Roulette Wheel Selection)

交叉 (Crossover)

随机选择 2 条染色体，以交叉概率 P_c 交换部分基因，来构造下一代的 2 条新的染色体

变异 (Mutation)

在繁殖过程，新产生的染色体中的基因会以一定的概率出错，称为变异。变异发生的概率记为 P_m

适应度函数 (Fitness Function)

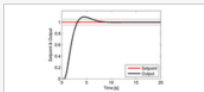
用于评价某个染色体的适应度，用 $f(x)$ 表示。

伪代码

```
1 // Pc : 交叉发生的概率
2 // Pm : 变异发生的概率
3 // M : 种群规模
4 // G : 终止进化的代数
5 // Tf : 进化产生的任何一个个体的适应度函数超过Tf , 则可以终止进化过程
6 初始化Pm , Pc , M , G , Tf等参数。随机产生第一代种群Pop
7 do {
8     计算种群Pop中每一个体的适应度F(i)。
9     初始化空种群newPop
10    do {
11        根据适应度以比例选择算法从种群Pop中选出2个个体
12        if ( random ( 0 , 1 ) < Pc ) 对2个个体按交叉概率Pc执行交叉操作
13        if ( random ( 0 , 1 ) < Pm ) 对2个个体按变异概率Pm执行变异操作
14        将2个新个体加入种群newPop中
15    } until ( M个子代被创建 )
16    用newPop取代Pop
17 }until ( 任何染色体得分超过Tf , 或繁殖代数超过G )
```

MATLAB 工具箱

以 MATLAB 为平台所编制的遗传算法优化工具箱 (Genetic Algorithm Optimization Toolbox, 缩写为 GAOT) 可以利用 MATLAB 强大的仿真功能, 进行 GA 的各种优化运算。



GAOT-ECM: Extension For Control And Modeling

by Andreea Ion

02 Jun 2015 (Updated 15 Jul 2015)

GAOT-ECM implements genetic algorithms for industrial plant identification and PID controller design

 [Watch this File](#)

★★★★★

4.5 | 2 ratings

[Rate this file](#)

42 Downloads (last 30 days)

File Size: 89.1 KB

File ID: #51072

Version: 1.0

The End