

模拟退火算法中的退火策略研究

高 尚

(华东船舶工业学院 电子与信息系,江苏 镇江 212003)



摘 要:退火策略是模拟退火算法中的重要一环,本文将研究退火策略对模拟退火算法的影响问题,给出了 MATLAB 语言程序,典型复杂函数优化的仿真表明退火速率应适中。

关键词:模拟退火算法;退火策略;优化;MATLAB

中图分类号:O224

文献标识码:A

引言

模拟退火算法最早思想由 Metropolis 在 1953 年提出,1983 年 Kirkpatrick 等成功地将退火思想引入组合优化领域。模拟退火算法是局部搜索算法的扩展,理论上来说,它是一个全局最优算法。目前已在工程中得到了广泛应用^[1~7],诸如生产调度、控制工程、机器学习、神经网络、图像处理等领域。退火策略是模拟退火算法中的重要一环,有关退火策略对模拟退火算法的影响的文献并不多,本文将研究退火策略对模拟退火算法的影响问题。

1 模拟退火算法

模拟退火算法用于优化问题的出发点是基于物理中固体物质的退火过程与一般优化问题的相似性。算法的基本思想是从一给定解开始的,从邻域中随机产生另一个解,接受准则允许目标函数在有限范围内变坏,它由一控制参数 t 决定,其作用类似于物理过程中的温度 T ,对于控制参数 t 的每一取值,算法持续进行“产生新解 - 判断 - 接受或舍弃”的迭代过程,对应着固体在某一恒定温度下趋于热

平衡的过程。经过大量的解变换后,可以求得给定控制参数 t 值时优化问题的相对最优解。然后减小控制参数 t 的值,重复执行上述迭代过程。当控制参数逐渐减小并趋于零时,系统亦越来越趋于平衡状态,最后系统状态对应于优化问题的整体最优解。该过程也称冷却过程。由于固体退火必须缓慢降温,才能使固体在每一温度下都达到热平衡,最终趋于平衡状态,因此,控制参数的值须缓慢衰减,才能确保模拟退火算法最终趋于优化问题的整体最优解。

对于一般无约束优化问题: $\min f(X)$

模拟退火算法的一般框架^[1~3]:

给定起、止“温度” T 、 T_0 , 模拟参数初始化 X_0 ;

While ($T > T_0$) do

在 X_0 的邻域内模拟产生随机扰动 X ;

计算扰动引起的目标函数(能量)值的变化 E ;

若, $E \leq 0$, 接受新值, 否则若

$\exp(-E/T) > \text{rand}(0,1)$ ($\text{rand}(0,1)$ 表示 0~1 之间的随机数), 也接受新值, 否则就拒绝;

确定新的参数值, 若扰动被接受, 则 $X_0 \leftarrow X_0 + X$, 否则 $X_0 \leftarrow X_0$;

若接受新值, 降温 $T \leftarrow \text{update}(T)$, 否则不降温;

End

其中: $T \leftarrow \text{update}(T)$ 就是退火策略, 也就是温度下降方法。

2 常见的退火策略

收稿日期:2002-07-10

基金项目:华东船舶工业学院青年基金资助(2002313)

作者简介:高尚(1972-),男,江苏镇江人,硕士,讲师,主要从事系统理论与优化等方面的研究。

常见的退火策略有下面几种:设 t_k 为第 k 迭代时温度。

对数下降^[2]: $t_k = \log(k + k_0)$

快速降温^[2]: $t_k = 1/(1 + k)$

直线下降^[1]: $t_k = (1 - \frac{K}{k}) t_0$

指数退温^[1,2]: $t_k = t_{k-1}$

四种退火方法的温度下降速度是不一样的,指数退温是最常用的一种策略,其温度变化很有规律,直接与参数 a 相关。这里只研究指数退温策略,以此来总结规律。

由 $t_k = at_{k-1}$ 可知, $t_k = a^k t_0 = t_0 e^{\ln a \cdot k}$ 因此称为指数退温,设起始温度为 $T = 10000$,其温度随着参数值不同与迭代次数 a 的关系如图 1 所示。

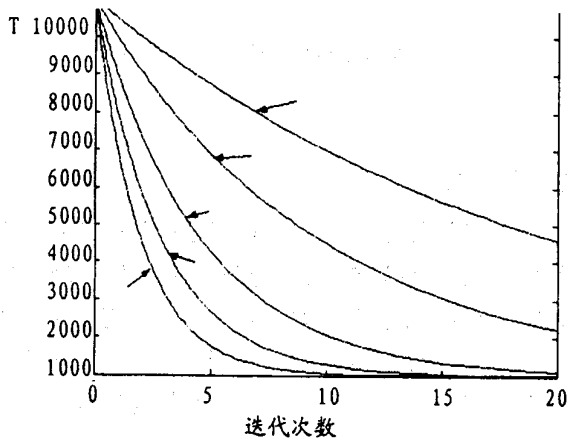


图 1 温度随着参数 值不同与迭代次数的关系图

由图 1 可知: 值越小,退火速度越快,所以我们重点研究不同 值对模拟退火算法的影响。

3 程序设计与测试分析

3.1 程序设计

为了说明问题,以下面 4 个国内外学者经常采用的优化测试函数^[2]来研究。

$$F_1 = 100(x_1^2 - x_2)^2 + (1 - x_1)^2 \quad (1)$$

$$F_2 = \max(|x_1|, |x_2|) \quad (2)$$

$$F_3 = \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2} - 0.5 \quad (3)$$

$$F_4 = x_1^2 + x_2^2 \quad (4)$$

以解 F_1 为例来说明用 MATLAB 语言设计的方法与步骤:

第一步:编写目标函数文件 aneal-f1.m

```
function y = aneal-f1(x)
```

```
y = 100 * (x(1) * x(1) - x(2)^2 + (1 - x(1))^2);
```

第二步:编写模拟退火算法程序 aneal-t1.m
(假设起始温度 $T = 10000$,终止温度 $T_0 = 1$, $a = 0.9$)

```
x1 = [1 1]; % 参数初始化
```

```
T = 100000;
```

```
a = 0.9;
```

```
N = 1;
```

```
while T > 1
```

```
    f1 = aneal-f1(x1); %计算原来目标函数
```

```
    x2(1) = x1(1) + 0.2 * (rand - 0.5);
```

```
    x2(2) = x1(2) + 0.2 * (rand - 0.5);
```

```
    f2 = aneal-f1(x2); %计算新的目标函数
```

```
    if f2 - f1 < 0
```

```
        T = T * a;
```

```
        x1 = x2;
```

```
        cc(N) = f2;
```

```
        N = N + 1;
```

```
    elseif exp((f1 - f2) / T) > rand
```

```
        T = T * a;
```

```
        x1 = x2;
```

```
        cc(N) = f2;
```

```
        N = N + 1;
```

```
    end
```

```
end
```

```
f1 = aneal-f1(x1); %计算最终目标函数
```

```
disp('The result are:')
```

```
disp(x(1) X(2))
```

```
disp(x1)
```

```
disp('The objective function is:')
```

```
disp(f1)
```

```
tt = 1:N - 1;
```

```
plot(tt,cc) %绘画出迭代过程
```

为测试需要,运行 1000 次的程序如下:

```
time1 = datestr(now) %初始时间
```

```
mm = 1;
```

```
while mm <= 1000
```

```
    aneal-t1 %调用程序
```

```
    opf(mm) = f1;
```

```
    mm = mm + 1;
```

```

end
time2 = datestr(now) %运行后时间
min(opf) %最好目标值
max(opf) %最差目标值
mean(opf) %最差目标值

```

3.2 测试分析

对每个程序运行 1000 次,对 F_1 函数的模拟参数初始化 $X_0 = [0, 0]$, 结果如表 1, 最优解为 $F_{1\min} = F_1(1, 1) = 0$; 对 F_2 函数的模拟参数初始化 $X_0 = [1, 1]$, 结果如表 2, 最优解为 $F_{2\min} = F_2(0, 0) = 0$; 对 F_3 函数的模拟参数初始化 $X_0 = [1, 1]$, 结果如表 3, 最优解为 $F_{3\min} = F_3(0, 0) = -1$; 对 F_4 函数的模拟参数初始化 $X_0 = [1, 1]$, 结果如表 4, 最优解为 $F_{4\min} = F_4(0, 0) = 0$ 。

表 1 取不同值时解目标函数的结果

结果	运行 1000 次 总的时间(s)	最好 目标值	最差 目标值	平均 目标值
0.95	49	0.0089	9.8853	1.6919
0.9	25	0.0026	35.3377	2.1987
0.8	12	0.0572	57.5572	3.8577
0.7	8	0.0765	60.2747	4.5553
0.6	5	0.1456	76.4038	4.4441

表 2 取不同值时解目标函数的结果

结果	运行 1000 次 总的时间(s)	最好 目标值	最差 目标值	平均 目标值
0.95	60	0.0626	3.7922	1.4744
0.9	21	0.0931	2.8654	1.3305
0.8	11	0.2778	2.1908	1.2162
0.7	7	0.0765	2.1093	1.1913
0.6	6	0.3973	1.8274	1.1617

表 3 取不同值时解目标函数的结果

结果	运行 1000 次 总的时间(s)	最好 目标值	最差 目标值	平均 目标值
0.95	52	-0.9903	-0.0025	-0.3595
0.9	26	-0.9885	-0.0025	-0.2578
0.8	12	-0.9397	-0.0025	-0.1621
0.7	8	-0.9067	-0.0025	-0.1078
0.6	6	-0.8782	-0.0025	-0.0880

表 4 取不同值时解目标函数的结果

结果	运行 1000 次 总的时间(s)	最好 目标值	最差 目标值	平均 目标值
0.95	48	0.0057	19.7519	3.0901
0.9	21	0.0043	15.1206	2.6141
0.8	10	0.0485	7.0043	2.2828
0.7	7	0.1019	6.1188	2.1953
0.6	5	0.2034	4.9562	2.1665

从表 1~4 可知, 值越小, 退火速度越快, 迭代次数减少, 运行时间短; 对于最好目标值这一指标来说, 应取大点; 对于最差目标值和平均目标值这两个指标来说, 取值不具有统一性结论, 对于目标函数, 应取大点, 而对于目标函数 F_1 , 应取小点。

综合几个指标, 建议 取适中。对于其他退火函数, 采取的策略应该使退火速率适中。图 2 显示了 $\alpha = 0.9$ 对 F_1 函数, 用模拟退火算法的运行的某一次的迭代过程。

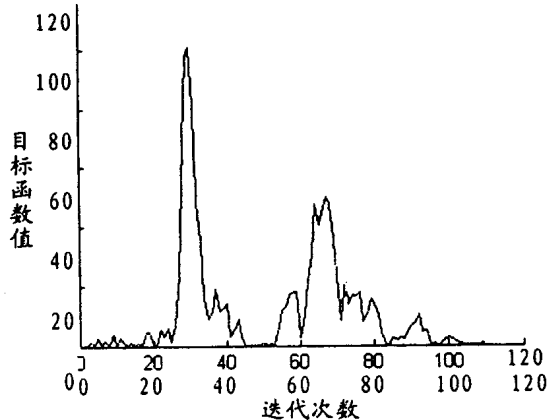


图 2 对函数模拟退火算法的一次的迭代过程

4 结束语

通过对测试函数的具体运行结果分析了退火策略对模拟退火的影响, 其理论分析有待进一步研究。模拟退火算法是依赖邻域结构的迭代方法, 模拟退火算法对选择试验解比较敏感^[6,7], 对如何选择初值, 如何找邻解, 如何设置初温等参数设计还有待研究。

参考文献:

- [1] 邢文循, 谢金星. 现代优化计算方法[M]. 北京: 清华大学出版社, 1999: 90 - 129.
- [2] 王凌. 智能优化算法及其应用[M]. 北京: 清华大学出版社, 2001(10): 17 - 35.
- [3] 刘宝碇, 赵瑞清. 随机规划与模糊规划[M]. 北京: 清华大学出版社, 1998: 45 - 50.
- [4] 张德富, 顾卫刚, 沈平. 一种解旅行商问题的并行模拟退火算法[J]. 计算机研究与发展, 1995(2).
- [5] 田湃, 杨自厚, 张嗣瀛. 一类非线性规划的模拟退火求解[J]. 控制与决策, 1994, 9(3): 173 - 177.
- [6] Kirkpatrick S, Gelatt Jr C D, Vecchi Jr M P. Optimization by simulated annealing[J]. Science, 1983.
- [7] A. Bolte, U. W. Thonemann, Optimisation simulated annealing schedules with genetic programming[J]. European Journal of Operational Research 92 (1996): 402 - 416.

(下转第 26 页)

方程,其中的四面体网格生成和自适应网格的细化均用 Delaunay 三角剖分技术实现。计算实践表明,非结构网格的自适应技术可以用相对较少的网格计算出较精确的结果,使计算效率得到明显的提高。

参考文献:

- [1] Jameson A., Baker T.J. and Weatherhill N. P.. Calculation of Inviscid Transonic Flow over a Complete Aircraft [J]. AIAA Paper 86 - 0103, 1986.
- [2] Loehner R.. FEM in CFD: Grid Generation, Adaptivity and Parallelization [J]. N92 - 27679, 1992.
- [3] Owen S J. A Survey of Unstructured Mesh Generation Technology. steve.owen @ansys.com.
- [4] 朱培焯. 空间曲面的非结构网格生成 [J]. 航空学报, 2001, 22(2): 151 - 154.
- [5] 朱培焯. 三维非结构网格自动生成 [J]. 计算物理, 2001, 18(6): 573 - 576.

Three-Dimensional Adaptive Unstructured Mesh for Euler Equations

ZHU Pei-ye

(Aeronautical Computing Technique Research Institute, Xi'an 710068, China)

Abstract: The three-dimensional Euler equations are solved on unstructured tetrahedral meshes using an adaptive strategy. The basic algorithm for the Euler solver consists of an explicit vertex-based finite volume method. The Delaunay triangulation is used both to generate the meshes and to refine the meshes. The transonic flow over the ONERA M6 wing is calculated by means of the adaptive strategy, and the results are compared with experimental data to show the accuracy and efficiency of the present method.

Key words: adaptive mesh technique; Euler equations; unstructured mesh

(上接第 22 页)

Research on annealing strategy in Simulated Annealing Algorithm

GAO Shang

(East China Shipbuilding Institute, Zhenjiang 212003, China)

Abstract: The annealing strategy is an important step in simulated annealing algorithm. The effect of annealing strategy in simulated annealing algorithm is researched in this paper. MATLAB programs of simulated annealing algorithm are given. Simulation results of typical complex function optimization show that the speed of annealing must be moderate.

Key words: simulated annealing algorithm; annealing strategy; optimization; MATLAB