

文章编号: 1001-0920(2005)05-0579-04

## 函数优化问题的一种异步并行模拟退火算法

王华, 唐国金

(国防科技大学 航天与材料工程学院, 湖南 长沙 410073)

**摘要:** 针对工程中的多极值点复杂函数的优化问题, 提出一种完全异步的粗粒度并行模拟退火算法。在一定时间内将一条Markov链分裂成多条Markov链, 并结合其他多种改进方法, 获得了可扩展的并行效果, 提高了算法应用的灵活性。数值计算表明, 该方法可显著提高算法的收敛速度。

**关键词:** 模拟退火; 并行算法; 函数优化

**中图分类号:** TP18 **文献标识码:** A

## An asynchronous parallel simulated annealing algorithm for function optimization problems

WANG Hua, TANG Guo-jin

(College of Aerospace and Material Engineering, National University of Defense Technology, Changsha 410073, China. Correspondent: WANG Hua, Email: wanghua21cn@21cn.com)

**Abstract:** To complex function optimization problems with multimimimize in practice, an asynchronous parallel simulated annealing algorithm is developed, which splits one Markov chain into several Markov chains, and combines other enhanced method. The performance of algorithm scales up with the increase of processors, and the character of asynchronous makes algorithm more suitable for all kinds of processor. The numerical computation results indicate that the proposed algorithm can significantly improve the convergence speed.

**Key words:** simulated annealing; parallel algorithm; function optimization

### 1 引言

在工程实际中, 经常遇到高维、非凸且有多个局部极值点的函数优化问题<sup>[1]</sup>, 对其采用传统的优化算法通常不能求得理想解。近年来, 模拟退火算法(SA)以其高效、健壮和通用灵活等特点, 广泛用于求解全局函数优化问题。

模拟退火算法的核心是模拟热力学, 尤其是模拟液体冷却成固体或金属退火的过程。模拟退火算法对于大规模优化问题, 尤其拥有众多局部极点的全局优化问题具有极高的效率。人们已应用模拟退火方法有效地解决了著名的旅行商问题。但模拟退火算法需要的优化时间很长, 所以人们试图通过对标准算法的改进来提高算法性能。

Hao<sup>[2]</sup>利用遗传算法的天然并行性, 将模拟退

火算法与遗传算法相结合, 以提高算法的速度和全局收敛性; Press<sup>[3]</sup>将单纯形方法与模拟退火算法相结合来改进算法的计算效率; 王凌<sup>[4]</sup>基于假设检验减少模拟退火算法的搜索次数, 提高了算法的性能。

本文针对模拟退火算法优化时间长的主要缺点以及函数优化的特点, 提出了一种粗粒度的异步并行机制混合模拟退火算法, 同时利用传统优化算法在局部函数优化问题中收敛快速的优点, 在适当的时候将传统算法与模拟退火算法相结合, 进一步提高算法的计算效率。由于本文算法的完全异步特性, 使得算法具有极大的灵活性, 能够更好地适应各种不同的硬件条件。数值实验表明, 本文方法在保证优化质量的前提下能大大提高模拟退火算法的计算效率。

收稿日期: 2004-05-31; 修回日期: 2004-07-15

作者简介: 王华(1977—), 男, 陕西韩城人, 博士生, 从事优化控制等研究; 唐国金(1963—), 男, 湖南澧县人, 从事动力学系统建模与仿真、结构完整性等研究。

## 2 标准模拟退火算法描述

模拟退火算法最早由Metropolis<sup>[5]</sup>等提出。它对于随进程递减的温度 $T$ 的每一取值,持续进行“产生新解-判断-接受/舍弃新解”的迭代过程。该过程对应于固体在某一温度下趋于热平衡的过程。

标准模拟退火算法的一般步骤可描述如下:

Step 1: 给定初温  $T = T_0$ , 随机产生初始状态  $x = x_0$ ;

Step 2: 根据给定的概率密度函数产生一个随机变量, 产生新状态  $x_{\text{new}} = \text{Generate}(x)$ ;

Step 3: 如果  $\min\{1, \exp[-(E(x_{\text{new}}) - E(x))/T]\}$  大于  $(0, 1)$  之间的随机数 (Metropolis 准则), 则接受新解,  $x = x_{\text{new}}$ , 其中  $E(x)$  为状态  $x$  对应的目标函数值;

Step 4: 若满足抽样稳定准则, 则转 Step 5; 否则, 转 Step 2;

Step 5: 退温;

Step 6: 若算法满足终止准则, 则算法结束, 输出结果; 否则, 转 Step 2;

## 3 基于并行思想的改进

模拟退火算法返回高质量近似解的时间花费较多, 当问题规模增大时, 会因难以承受运算时间而使算法丧失可行性。因此在保证一定优化质量的前提下, 提高算法的搜索效率是算法改进的主要内容, 而采用并行搜索结构是诸多改进方法中最直接最有效的一种, 它能够真正缩减算法的运算时间。

模拟退火算法并行实现的一种可能途径是独立试验并行<sup>[6]</sup>, 即在各处理机上独立完成算法, 然后比较各自的结果, 选取一个最优的作为整个实验的结果。该方法虽然提高了模拟退火算法求得全局最优解的概率, 并具有完全异步并行的特点, 但其实质相当于各台处理机的分别串行计算, 并不是一种真正的并行方法。

另一种并行途径是协同试验并行<sup>[6]</sup>, 即在每次循环中, 由各台处理机独立产生新解并判断, 然后按某个法则在各处理机的解中选择一个作为新解, 在这个新解的基础上进行下一轮循环试验。这种并行方法属于同步并行策略, 通常需要较高的通信开销, 适合于用同一档次处理器组成的硬件环境。

本文借鉴混乱松弛并行的特点, 将概率接受新解思想引入公共存储区的最优解接受中, 同时将经典模拟退火算法中的一条Markov链分裂为在一定时间内独立生长的多条Markov链, 取得了可扩展的并行性。考虑到传统局部优化算法在局部优化问题中的优异性能, 本文基于传统优化算法和模拟退火算法结合的思想, 将Powell法与模拟退火算法

相结合, 既可跳出局部极值点, 又可加快优化速度, 提高局部收敛性能。本文算法的另一个优点是避免了经典模拟退火算法对参数的依赖性, 在不需精心设计算法参数的情况下, 同样可取得满意的结果。

在本文设计的模拟退火算法中, 首先随机产生最优解  $x_g$ , 存储在公共存储区, 给定公共存储区的初温  $T_g = T_{g0}$ , 然后每个节点均执行如下算法:

Step 1: 给定初温  $T = T_0$ , 随机产生初始状态  $x = x_0$ ;

Step 2: 根据给定的概率密度函数产生一个随机变量, 产生新状态  $x_{\text{new}} = \text{Generate}(x)$ ;

Step 3: 如果  $\min\{1, \exp[-(E(x_{\text{new}}) - E(x))/T]\}$  大于  $(0, 1)$  之间的随机数 (Metropolis 准则), 则接受新解,  $x = x_{\text{new}}$ ;

Step 4: 以概率  $\alpha(T)$  调用Powell法;

Step 5: 若满足抽样稳定准则, 则转 Step 6; 否则, 转 Step 2;

Step 6: 调用Powell法对当前最优解进一步优化;

Step 7:  $T$  退温;

Step 8: 每经过  $m$  次退温后与公共存储区通信, 如果满足  $\min\{1, [1 - (1 - h)(E(x) - E(x_g))/T_g]^{1/(1-h)}\}$  大于  $(0, 1)$  之间的随机数 (Tsallis 准则), 则更新公共存储区的最优解; 否则  $\text{NoUpdate} = \text{NoUpdate} + 1$ , 并以公共存储区中最优解为新状态继续循环, 其中  $E(x)$  为状态  $x$  对应的目标函数值;

Step 9: 公共存储区温度  $T_g$  退温;

Step 10: 如果  $\text{NoUpdate} > N$ , 则令  $\text{NoUpdate} = 0$ , 然后  $T$  回火;

Step 11: 若满足算法终止准则, 则算法结束, 输出结果; 否则, 转 Step 2。

为了避免模拟退火算法对参数的依赖性, 对不同的节点取不同的冷却进度表。例如, 可在其中一个节点上取较高的初温和慢速的降温衰减函数, 使得算法能够在大领域范围内搜索, 增加跳出局部极小的机率。而在另一个节点上取相对较低的初温和较快的降温衰减函数, 使得算法能够快速到达极值点。为了增加算法的鲁棒性, 可在不同节点取不同的新状态接受准则和温度更新函数。另外, Markov链的长度选取以在每个温度上都能恢复准平衡为宜, 通常可取  $L = 100n$  或  $L = 10n_s$ , 这里,  $n$  为所求解问题的维数,  $n_s$  为每个温度下状态成功转移的次数。因为算法与传统局部优化算法相结合, 所以终止温度可不必选得非常低即可得到理想的结果。

#### 4 算法关键参数和操作的设计

模拟退火算法的性能很大程度上取决于算法参数和操作的选取设计, 通常只能通过启发式规则或大量的实验选取. 虽然本文算法对初值具有一定的鲁棒性, 但选取合适的参数和操作对算法性能的提高仍是非常重要的.

##### 4.1 模型扰动

模拟退火算法中新状态的产生是对当前状态进行扰动得到的. 状态产生函数最常用的方案是

$$x(k+1) = x(k) + \eta \xi$$

其中:  $\eta$  为扰动幅度参数,  $\xi$  为随机扰动参数. 随机扰动可以服从 Cauchy, Gauss 和均匀分布, 甚至是混沌机制.

本文采用文献[7]的方法产生新状态, 即依赖于温度的似 Cauchy 分布方法

$$x_i = x_i + y_i(B_i - A_i), \quad (1)$$

$$y_i = T \operatorname{sgn}(u - 0.5) [(1 + 1/T)^{|u-0.5|} - 1] \quad (2)$$

式中:  $x_i$  为当前状态中的第  $i$  个变量;  $u$  为  $[0, 1]$  区间上均匀分布的随机数;  $[A_i, B_i]$  为  $x_i$  的取值范围, 且要求扰动后的  $x_i \in [A_i, B_i]$ ;  $\operatorname{sgn}()$  为符号函数.

采用依赖于温度的似 Cauchy 分布产生新模型的优点是: 在高温情况下进行大范围的搜索, 在低温时搜索仅在当前模型附近进行, 而且由于似 Cauchy 分布有一平坦的“尾巴”, 使其易于迅速跳出局部极值, 加快了模拟退火方法的收敛速度.

##### 4.2 接受概率讨论

Tsallis<sup>[8]</sup> 将 Shannon 熵  $S = -K \sum_{i=1}^W p_i \ln p_i$  (此处  $p_i$  是相应的几率) 推广为

$$S_q = -K \sum_{i=1}^W p_i^q / (q-1). \quad (3)$$

可以看出, 当  $q \rightarrow 1$  时, 式(3)为 Shannon 熵的表达式. 式中:  $\sum_{i=1}^W p_i^q = 1$ ,  $W$  为可能态的总数;  $K$  为常数;  $q$  为任意实数;  $p_i$  为微观状态的概率.

由式(3)导出的广义 Gibbs 分布为

$$p_i = [1 - \beta(q-1)E_j]^{1/(q-1)} / Z_q, \quad (4)$$

$$Z_q = \sum_{j=1}^W [1 - \beta(q-1)E_j]^{1/(q-1)}. \quad (5)$$

其中:  $Z_q$  为推广配分函数,  $\beta = 1/KT$  为 Lagrange 常数. 当  $q \rightarrow 1$  时, 式(4)和(5)即为

$$p_i = \exp(-\beta E_j) / Z_1, \quad (6)$$

$$Z_q = \sum_{j=1}^W \exp(-\beta E_j), \quad (7)$$

又恢复了 Boltzmann 统计, 式中  $E_j$  为系统能量.

式(6)和(7)即为常规的模拟退火算法的理论基础.

由式(4)可给出新的接受概率计算公式<sup>[9]</sup>

$$P = [1 - (1-h)\Delta E/T]^{1/(1-h)}. \quad (8)$$

式中:  $\Delta E = E(x) - E(x_0)$ ,  $E(x)$  为扰动得到的新模型目标函数,  $E(x_0)$  为当前模型的目标函数;  $T$  ( $T = 1/\beta$ ) 为温度;  $h$  为实数. 新模型按式(8)计算得到的概率进行接受, 称为 Tsallis 准则.

当  $h \rightarrow 1$  时, 式(8)为

$$P = \exp(-\Delta E/T), \quad (9)$$

也就是常规的模拟退火算法所用的 Metropolis 准则的接受概率计算式, 即 Boltzmann 概率分布.

##### 4.3 温度更新函数

通常当自然系统慢速降温时, 才能到达最小能量状态. 如果金属快速降温或淬火, 到达的将不是最小能量状态, 而是具有更高能量的多晶或非晶状态. 因此, 退火过程的本质是慢速降温, 允许充足的时间重新分布, 直到失去迁移率.

常用的控制温度衰减的函数是

$$T_{k+1} = \lambda T_k \quad (10)$$

其中:  $T$  为温度,  $\lambda$  的取值范围一般是  $0.8 < \lambda < 0.999$ . 式(10)对控制参数的衰减量是随算法进程递减的, 因此符合金属的降温规律.

#### 5 数值仿真

选取如下 Generalized Schwefel 函数作为算法测试问题 (维数取为 10):

$$f_1(x) = - \sum_{i=1}^n (x_i \sin(\sqrt{|x_i|})), \quad (11)$$

$$-500 \leq x_i \leq 500$$

上述函数是一个典型的多极小函数, 它在二维情况下的函数图像如图 1 和图 2 所示. 函数的全局极小点在  $x_i = 420.9687, i = 1, 2, \dots, n$ ;  $n$  个次最优极小点在  $x_i = 420.9687, i = 1, 2, \dots, n, i \neq j; x_j = -302.5232$ . 因此最优极小点远离全局极小点, 故对于一般的搜索算法而言, 极易陷入局部极小.

分别用串行算法、协同试验并行策略和本文的并行策略对算例随机计算 50 次 (并行算法的

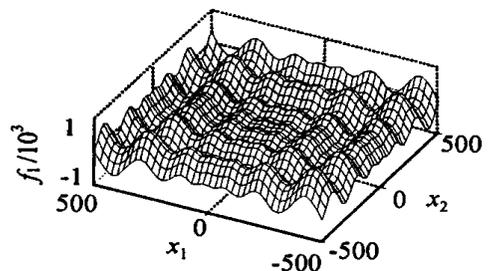


图 1 二维 Generalized Schwefel 函数的图形

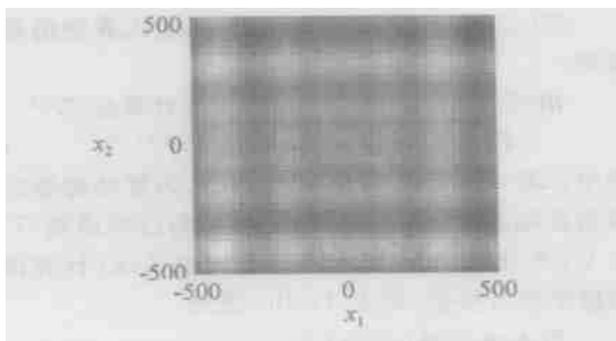


图2 二维 Generalized Schwefel 函数的等高线图(颜色越深,其值越小)

Markov 链比串行算法短,缩短程度根据处理器台数的不同而定),采用两个处理器时的仿真统计结果如表1所示。当搜索最终解与问题最优解的绝对误差小于0.2%时,则认为算法搜索完成。

表1 两种并行策略的模拟结果

并行策略	加速比	效率
协同试验并行	1.822 0	0.911 0
本文并行策略	1.837 4	0.918 7

本文的并行方法采用完全异步的并行策略,每台处理器的试验过程中公共存储区的内容都可能被别的处理器改变。因此,当控制参数  $T$  较大时,各个试验的接受概率也大,可能出现一些混乱。但随着  $T$  的减小及解的优化,混乱将会逐渐减少,这与混乱松弛并行类似。但本文算法中公共存储区根据概率接受各处理器的新解,在一定程度上降低了机器出错的概率,有利于算法的全局收敛。与协同试验并行策略的结果相比,本文算法的加速比没有明显的优势,但本文的并行是一种完全异步的并行策略,具有更大的灵活性,尤其在档次不同的处理器组成的并行环境中,能够更好地发挥利用各处理器的处理能力,取得比协同试验并行更大的加速比。

## 6 结 论

在工程问题中,经常需要解决高维、非凸且多极值函数的优化问题,而传统局部优化算法针对这类问题无能为力。模拟退火算法常常能够有效地解决这类问题,但它存在优化时间长的缺点。本文提出了一种完全异步的粗粒度并行模拟退火算法,并在算法中结合了回火、记忆和传统局部优化算法,能够显著减少优化时间,具有灵活高效的特点。

可进一步将粗粒度、细粒度的并行算法相结合,得到混合型的并行模拟退火算法,以进一步提高算法的灵活性,充分利用计算资源。这种算法的拓扑呈层次结构,粗粒度算法位于顶层,细粒度算法位于底层,或顶层、底层皆用粗粒度算法,以获得更快的运算速度。

## 参考文献 (References)

- [1] 王华,唐国金,雷勇军. 有限推力轨迹优化问题的直接打靶法研究[J]. *中国空间科学技术*, 2003, 23(5): 51-56  
(Wang H, Tang G J, Lei Y J. The direct shooting method solving finite-thrust trajectory optimization[J]. *Chinese Space Science and Technology*, 2003, 23(5): 51-56.)
- [2] Hao C, Nicholas S F, Daniel W W. Parallel genetic simulated annealing: A massively parallel SMD algorithm[J]. *Parallel and Distributed Systems*, 1998, 9(2): 126-136.
- [3] Press W H, Teukolsky S A, Vetterling W T, et al. *Numerical recipes in C++: The art of scientific computing* [M]. Cambridge: Cambridge University Press, 2002: 448-459.
- [4] 王凌,郑大钟. 随机优化问题一类基于假设检验的模拟退火算法[J]. *控制与决策*, 2004, 19(2): 183-186  
(Wang L, Zheng D Z. Simulated annealing approach based on hypothesis test for stochastic optimization problems[J]. *Control and Decision*, 2004, 19(2): 183-186.)
- [5] Metropolis N, Rosenbluth A W, Rosenbluth M N, et al. Equation of state calculations by fast computing machines[J]. *J of Chemical Physics*, 1953, 21(6): 1087-1092.
- [6] 唐立山,谢云,尤矢勇,等. *非数值并行算法——模拟退火算法*[M]. 北京: 科学出版社, 1994.
- [7] Ingber L. Very fast simulated reannealing [J]. *Mathematical and Computer Modeling*, 1989, 12(8): 967-973.
- [8] Tsallis C. Possible generalization of Boltzmann-Gibbs statistics[J]. *J of Statistical Physics*, 1988, 52(1-2): 479-487.
- [9] Penna T J P. Traveling salesman problem and tsallis statistics[J]. *Physical Review E*, 1995, 51(1): R1-R3.