

一类连续函数模拟退火算法及其收敛性分析^{*1)}

靳利霞 唐焕文 李 斌 计明军 朱训芝
(大连理工大学应用数学系, 大连, 116024)

摘 要

高维连续函数的全局优化问题普遍存在于计算生物学、计算化学等领域. 针对这类问题和现有连续函数模拟退火算法的某些不足, 本文给出了一类改进的模拟退火算法. 采用一种简单的方法证明了算法的全局收敛性. 数值结果表明, 对于高维连续函数, 该算法能够快速有效地收敛到全局最优点, 比较了两种新解产生方法的试验结果.

关键词: 连续函数, 全局优化, 模拟退火算法, 收敛性

MR (2000) 主题分类: 90C30, 65K05

A SIMULATED ANNEALING ALGORITHM FOR CONTINUOUS FUNCTIONS AND ITS CONVERGENCE PROPERTIES

Jin Lixia Tang Huanwen Li Bin Ji Mingjun Zhu Xunzhi

(Department of Applied Mathematics, Dalian University of Technology, Dalian 116024)

Abstract

The global optimization problems of continuous high-dimensional function frequently present in the fields of computational biology and computational chemistry. Regarding of the characters of this class of problems and the disadvantages of the existing simulated annealing algorithms, we propose an improved simulated annealing algorithm. Its convergence properties are proved by using a simple approach. Numerical tests on some typical optimization problems show that this is an effective algorithm. The testing results by two different new points generating approaches are compared.

Keywords: continuous function; global optimization; simulated annealing algorithm; convergence property

2000 Mathematics Subject Classification: 90C30, 65K05

1. 引 言

模拟退火算法是一种高效的全局优化算法, 自从 1983 年 Kirkpatrick^[1] 提出该算法以来, 已经在自然科学、工程技术和管理等诸多领域得到广泛应用. 模拟退火算法更多的是应用于组合优化问题; 离散情况下模拟退火算法的理论研究比较多. 对于组合优化问题, 模拟退火

* 2003 年 5 月 23 日收到.

1) 国家自然科学基金 (90103033) 资助.

过程可以描述为从一个状态转移到另一个状态的随机游动, 利用 Markov 链的性质, 分析模拟退火算法对应 Markov 链的周期性和不可约性, 可以证明在一定条件下模拟退火算法的全局收敛性^[2,3].

针对连续函数全局优化问题, 也已经有不少研究, 例如 Ali^[4] 等学者提出了基于局部搜索的连续函数模拟退火算法, 基于目标的连续函数模拟退火算法等. 然而, 连续函数的模拟退火过程更为复杂, 不能用一般的 Markov 链来描述, 关于算法的收敛性研究还比较少, 有文献利用 Langevin 型 Markov 扩散过程或连续状态 Markov 过程进行了理论分析, 但收敛性条件比较强, 例如要求目标函数二次可微等^[5,6]. 刘勇等研究了 R^d 上具有间断梯度的模拟退火算法的收敛性^[7]. 此外, 关于模拟退火算法如何提高计算效率和成功率的新近研究还有^[19-20].

将模拟退火算法引入连续函数全局优化问题, 一个关键是新解的产生方法和邻域结构的构造. 最直接的方法是在可行域上以均匀分布产生新解, 然而这种方法忽略了当前函数值的结构信息; 针对该问题, 有作者提出一种策略^[4], 即对于当前解 X , 以下面方式产生新解 Y :

$$g_{xy} = \begin{cases} LS(X), & \omega \geq q \\ 1/m(S), & \omega < q \end{cases}$$

其中, S 为可行域; $LS(X)$ 表示沿下降方向从当前解 X 产生新解 Y 的一种局部搜索方法; $m(S)$ 为可行域的 Lebesgue 测度; ω 是 $(0, 1)$ 之间一个随机数; q 是 $(0, 1)$ 之间一个固定值, 例如取 $q = 0.75$. 这种方法被称为基于局部搜索的模拟退火算法 (LSA). 显然, 这种算法不仅需要主观地确定 q 的值, 而且需要采用局部最优化方法.

本文提出了一类适用于连续函数全局优化问题的模拟退火算法, 采用直接产生新解的方法. 对于高维的连续函数, 每次更新只在一个方向上产生随机扰动优于在多个方向上同时产生扰动, 新解可以以不同的概率分布产生, 参照郭崇慧, 唐焕文证明一类进化算法收敛性的思想和方法^[8-10], 证明了以正态分布产生新解情况下, 算法的全局收敛性. 该方法比较简洁, 而且条件较弱 ($f \in C$).

本文第二部分为算法描述, 第三部分对算法的全局收敛性进行分析证明; 第四部分选取了八个典型算例对两种不同的新解产生方法进行了测试比较, 数值结果说明, 所给出的算法是可行有效的. 最后是结论部分, 总结了算法使用中需要注意的问题.

2. 算法描述^[11]

考虑具有上下界约束的连续函数全局优化问题:

$$(P) \quad \begin{cases} \min f(X) \\ X \in \Omega = [X^D, X^U] \end{cases} \quad (1)$$

其中, $X = (x_1, x_2, \dots, x_n)^T$, $f \in C$. 设 X^* 为 (P) 的全局最优解, 令 $f^* = f(X^*)$.

假设从当前解 $X = (x_1, x_2, \dots, x_n)^T$ 中随机选取一个分量 x_r ($r = 1, \dots, n$), 产生随机

扰动: $x'_r = x_r + \Delta$, 然后进行边界处理得到新解:

$$x_r' = \begin{cases} x_r^U - (x_r^D - x_r') & x_r' < x_r^D \\ x_r' & x_r^D \leq x_r' \leq x_r^U \\ x_r^D + (x_r' - x_r^U) & x_r' > x_r^U \end{cases} \quad (2)$$

关于随机扰动 Δ 的产生, 本文采用了两种不同方法:

- a. 新解在整个实数空间产生, 且 $\Delta \sim N(0, \sigma^2)$.
- b. 将新解分量 x'_r 限制在当前解分量 x_r 的一个邻域范围内, 即 $\Delta = rand \cdot scale \cdot (x_r^U - x_r^D)$, 其中 $rand$ 为 $[-1, 1]$ 内一个随机数, $scale$ 为邻域规模因子, u_r^U 和 u_r^D 分别为 x_r 的上下界.

本文算法的具体实现过程为:

step1: 给定初始温度 T_{\max} , 每一温度下的迭代次数 L_{\max} , 邻域规模因子 $scale$ (或正态分布标准差 σ), 温度下降因子 dt ($0 < dt < 1$);

step2: 随机产生初始解 $X^0 = (x_1^0, x_2^0, \dots, x_n^0)^T \in \Omega$, 计算目标函数值 $f(X^0)$;

step3: 判断是否满足终止条件, 若满足则结束; 否则, 令 $T := T \times dt$, $L := 1$, 转 step4;

step4: 从 $\{1, 2, \dots, n\}$ 中随机选取一个数 r , 使 X^0 在第 r 个方向产生随机扰动: $x'_r = x_r^0 + \Delta$, 并按式 (2) 进行边界处理得到 x_r , 直到 $x_r \in [x_r^D, x_r^U]$ 为止; 其它方向保持不变, $x_i = x_i^0 (i \neq r)$; 从而得到一个新解 $X = (x_1, x_2, \dots, x_n)^T$, 计算新解的目标函数值 $f(X)$;

step5: 计算目标函数的差值 $\Delta f = f(X) - f(X^0)$, 根据 Metropolis 准则接收新解, 即产生一个服从 $(0, 1)$ 均匀分布的随机数 η , 新解的接收概率为 $A = \min\{1, e^{-\Delta f/T}\}$, 如果 $\eta \leq A$, 则接收新解, 令 $X^0 := X$, $f(X^0) := f(X)$; 否则, 拒绝新解; 转 step6;

Step6: 若 $L > L_{\max}$, 则转 Step3; 否则令 $L := L + 1$, 转 step4.

该算法有以下几个可调参数:

- 1) 初始温度 T_{\max} : 主要根据目标函数取值范围来确定, 取值应保证初始接受率足够高;
- 2) 终止条件: 在最优值未知的情况下, 终止条件比较难确定, 一般采取两种准则: 一是给定终止温度; 二是连续多次降温, 目标函数值不再下降;
- 3) 温度下降因子 dt : 根据函数性质取 $0 < dt < 1$;
- 4) 同一温度迭代次数 L_{\max} : L_{\max} 与问题规模和解空间大小有关;
- 5) 邻域规模因子 $scale$: 邻域规模因子的取值和解空间有直接关系, 本文取 $0.1 \sim 1.0$.
- 6) 正态分布标准差 σ : σ 值大, 跳出局部极小点的可能也大; σ 值小, 收敛到极小点的速度快. 需要权衡二者进行选择.

3. 收敛性分析

为了对算法进行收敛性分析, 引入下列定义和引理^[12]:

定义 1. 设 $\xi_n (n = 1, 2, \dots)$ 为概率空间上定义的随机序列, 若存在随机变数 ξ , 使对于 $\forall \varepsilon > 0$, 有

$$\lim_{n \rightarrow \infty} P(|\xi_n - \xi| < \varepsilon) = 1 \quad (3)$$

则称随机序列 $\{\xi_n\}$ 依概率收敛于随机变数 ξ .

定义 2. 对于上述随机序列, 若存在随机变数 ξ , 使

$$P\left\{\lim_{n \rightarrow \infty} \xi_n = \xi\right\} = 1; \quad (4)$$

或者, 对于 $\forall \varepsilon > 0$, 有

$$P\left\{\bigcap_{n=1}^{\infty} \bigcup_{k \geq n} [|\xi_k - \xi| \geq \varepsilon]\right\} = 0 \quad (5)$$

则称随机序列 $\{\xi_n\}$ 以概率为 1 收敛于随机变数 ξ .

显然, 以概率为 1 收敛比依概率收敛要强.

引理 1. 波雷尔 - 坎特利 (Borel-Cantelli) 引理

设 A_1, A_2, \dots 是概率空间上的一事件序列, 令 $p_k = P\{A_k\}$.

若 $\sum_{k=1}^{\infty} p_k < \infty$, 则

$$P\left\{\bigcap_{n=1}^{\infty} \bigcup_{k \geq n} A_k\right\} = 0; \quad (6)$$

若 $\sum_{k=1}^{\infty} p_k = \infty$, 且各 A_k 相互独立, 则

$$P\left\{\bigcap_{n=1}^{\infty} \bigcup_{k \geq n} A_k\right\} = 1. \quad (7)$$

下面对前述模拟退火算法的收敛性进行分析.

由于问题 (P) 满足以下条件: 1) 可行域 Ω 为 R^n 中的有界单连通闭区域; 2) 目标函数 $f(X)$ 是区域 Ω 上的连续函数, 所以 $S = \{X | \arg \min_{X \in \Omega} f(X)\} \neq \emptyset$. 对于任给 $\varepsilon > 0$, 记

$$\begin{aligned} D_0 &= \{X \in \Omega | f(X) - f^* < \varepsilon\}; \\ D_1 &= \Omega \setminus D_0. \end{aligned} \quad (8)$$

其中, $f^* = \min_{X \in \Omega} f(X)$.

假设 1. 在 Metropolis 准则中, 当 $\Delta f = f(X) - f(X^0) > 0$ 时, 设 $\Delta f \geq \tau$, τ 为某个充分小的正数, 或为常数, 或比温度 t 的收敛阶小.

利用本文算法求解问题 (P), 在一定温度 t 下, 当 $X^{(k)}(t) \in D_i$ 时, $X^{(k+1)}(t) \in D_j$ 的概率记为 p_{ij} .

引理 2. 在假设 1 的前提下, 若新解产生满足 $\Delta \sim N(0, \sigma^2)$, 则:

$$\begin{aligned} q_{01} &= \lim_{t \rightarrow 0} p_{01} = 0, q_{00} = \lim_{t \rightarrow 0} p_{00} = 1 \\ q_{11} &= \lim_{t \rightarrow 0} p_{11} \leq c \in (0, 1), q_{10} = \lim_{t \rightarrow 0} p_{10} \geq 1 - c \in (0, 1). \end{aligned}$$

证明. 应用算法中的记号 $X^0 = X^{(k)}(t)$, $X^0 = (x_1^0, \dots, x_n^0)^T$, 并记 $\Omega_i = [x_i^D, x_i^U]$, 按照边界处理方式, 有

$$\begin{cases} m_1(x_i^U - x_i^D) + x_i' = x_i \\ x_i' - m_2(x_i^U - x_i^D) = x_i \end{cases} \quad (9)$$

其中, m_1, m_2 为非负整数. 上式相当于对实轴进行剖分, $[x_i^D, x_i^U]$ 称为第 0 区, 第 l 区可以表示为

$$[x_i^D + l(x_i^U - x_i^D), x_i^U + l(x_i^U - x_i^D)], l = 0, \pm 1, \pm 2, \dots$$

那么, 当 $X^{(k)}(t) \in D_0$ 时, $X^{(k+1)}(t) \in D_1$ 的概率为

$$\begin{aligned} & P(X^{(k+1)}(t) \in D_1 | X^{(k)}(t) \in D_0) \\ & \leq \sum_{i=1}^n \frac{1}{n} \sum_{l=-\infty}^{+\infty} \int_{(D_1)_{i,l}} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x'_i - x_i^0)^2}{2\sigma^2}} e^{-\frac{f(X^{(k+1)}(t)) - f(X^{(k)}(t))}{t}} dx'_i \end{aligned} \quad (10)$$

其中 $(\cdot)_{i,l}$ 代表在 i 分量上的对应于 l 区的部分. 由于假设 $f(X^{(k+1)}(t)) - f(X^{(k)}(t)) \geq \tau$, 至少是比 t 的收敛阶小, 那么

$$P(X^{(k+1)}(t) \in D_1 | X^{(k)}(t) \in D_0) \leq e^{-\frac{\tau}{t}} \quad (11)$$

$\lim_{t \rightarrow 0} P(X^{(k+1)}(t) \in D_1 | X^{(k)}(t) \in D_0) = 0$, 即 $q_{01} = \lim_{t \rightarrow 0} p_{01} = 0$, 显然, $q_{00} = \lim_{t \rightarrow 0} p_{00} = 1$.

1. 当 $X^{(k)}(t) \in D_1$ 时, $X^{(k+1)}(t) \in D_1$ 的概率为

$$\begin{aligned} & P(X^{(k+1)}(t) \in D_1 | X^{(k)}(t) \in D_1) \\ & \leq \sum_{i=1}^n \frac{1}{n} \sum_{l=-\infty}^{+\infty} \int_{(D_1)_{i,l}} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x'_i - x_i^0)^2}{2\sigma^2}} \min \left\{ 1, e^{-\frac{f(X^{(k+1)}(t)) - f(X^{(k)}(t))}{t}} \right\} dx'_i \\ & \leq \sum_{i=1}^n \frac{1}{n} \sum_{l=-\infty}^{+\infty} \int_{(D_1)_{i,l}} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x'_i - x_i^0)^2}{2\sigma^2}} dx'_i = c \end{aligned} \quad (12)$$

由于 $D_1 \subset [X^D, X^U]$, 则

$$P(X^{(k+1)}(t) \in D_1 | X^{(k)}(t) \in D_1) \leq c \in (0, 1) \quad (13)$$

从而

$$q_{11} = \lim_{t \rightarrow 0} p_{11} \leq c \in (0, 1), q_{10} = \lim_{t \rightarrow 0} p_{10} \geq 1 - c \in (0, 1).$$

定理 1. 在假设 1 的前提下, 对于问题 (P), 设在一定温度 t 下, 由算法产生的点列为 $X^{(k+1)}(t), (k = 1, 2, \dots)$, 若新解产生满足 $\Delta \sim N(0, \sigma^2)$, 则当温度 $t \rightarrow 0$ 时, 点列 $\{X^{(k)}(t)\}_{k=1}^{+\infty}$ 以概率为 1 收敛于 (P) 的全局极小点, 即

$$P\left(\lim_{k \rightarrow \infty} \lim_{t \rightarrow 0} f(X^{(k)}(t)) = f^*\right) = 1$$

证明.

$$\begin{aligned} P(X^{(k)}(t) \in D_1) &= P(X^{(k)}(t) \in D_1, X^{(k-1)}(t) \in D_1) \\ &\quad + P(X^{(k)}(t) \in D_1, X^{(k-1)}(t) \in D_0) \\ &= P(X^{(k)}(t) \in D_1, X^{(k-1)}(t) \in D_1) \\ &\quad + P(X^{(k)}(t) \in D_1 | X^{(k-1)}(t) \in D_0) P(X^{(k-1)}(t) \in D_0) \end{aligned}$$

由引理 2, $\lim_{t \rightarrow 0} P(X^{(k)}(t) \in D_1 | X^{(k-1)}(t) \in D_0) = 0$, 有

$$\lim_{t \rightarrow 0} P(X^{(k)}(t) \in D_1) = \lim_{t \rightarrow 0} P(X^{(k)}(t) \in D_1, X^{(k-1)}(t) \in D_1), \quad (14)$$

$$\begin{aligned} & P(X^{(k)}(t) \in D_1, X^{(k-1)}(t) \in D_1) \\ &= P(X^{(k)}(t) \in D_1, X^{(k-1)}(t) \in D_1, X^{(k-2)}(t) \in D_1) \\ & \quad + P(X^{(k)}(t) \in D_1, X^{(k-1)}(t) \in D_1, X^{(k-2)}(t) \in D_0) \end{aligned}$$

由算法决定的 $\{X^{(k)}(t)\}_{k=1}^{\infty}$ 是一个 Markov 链, 因此

$$\begin{aligned} & P(X^{(k)}(t) \in D_1, X^{(k-1)}(t) \in D_1, X^{(k-2)}(t) \in D_0) \\ &= P(X^{(k)}(t) \in D_1 | X^{(k-1)}(t) \in D_1, X^{(k-2)}(t) \in D_0) P(X^{(k-1)}(t) \in D_1, X^{(k-2)}(t) \in D_0) \\ &= P(X^{(k)}(t) \in D_1 | X^{(k-1)}(t) \in D_1) P(X^{(k-1)}(t) \in D_1 | X^{(k-2)}(t) \in D_0) P(X^{(k-2)}(t) \in D_0) \end{aligned}$$

从而

$$\begin{aligned} & \lim_{t \rightarrow 0} P(X^{(k)}(t) \in D_1, X^{(k-1)}(t) \in D_1) \\ &= \lim_{t \rightarrow 0} P(X^{(k)}(t) \in D_1, X^{(k-1)}(t) \in D_1, X^{(k-2)}(t) \in D_1) \end{aligned}$$

由归纳法可知

$$\begin{aligned} & \lim_{t \rightarrow 0} P(X^{(k)}(t) \in D_1) = \lim_{t \rightarrow 0} P(X^{(m)}(t) \in D_1, m = 1, \dots, k) \\ & \leq \prod_{m=2}^k \lim_{t \rightarrow 0} P(X^{(m)}(t) \in D_1 | X^{(m-1)}(t) \in D_1) = c^{k-1} \end{aligned} \quad (15)$$

对于任意 $\varepsilon > 0$, 令

$$q_k = \lim_{t \rightarrow 0} P\{|f(X^{(k)}(t)) - f^*| \geq \varepsilon\} = P\{|f(X^{(k)}(t \rightarrow 0)) - f^*| \geq \varepsilon\}, \quad (16)$$

则:

$$q_k = \begin{cases} 0 & \text{if } X^{(k)}(t \rightarrow 0) \in D_0 \\ \overline{p_k} & \text{if } X^{(k)}(t \rightarrow 0) \notin D_0. \end{cases} \quad (17)$$

由引理 2, $q_{01} = 0$, 所以

$$\overline{p_k} = P\{X^{(i)}(t \rightarrow 0) \notin D_0, i = 1, 2, \dots, k\} = q_{11}^k \leq c^{k-1}, \quad (18)$$

而

$$\sum_{k=1}^{\infty} q_k = \sum_{k=1}^{\infty} c^{k-1} = \frac{1}{1-c} < \infty, \quad (19)$$

由波雷尔-坎特里引理知:

$$P\left\{\bigcap_{n=1}^{\infty} \bigcup_{k \geq n} |f(x^{(k)}(t \rightarrow 0)) - f^*| \geq \varepsilon\right\} = 0. \quad (20)$$

根据定义可知, 在 $t \rightarrow 0$ 时, 算法以概率 1 收敛于全局极小点, 即:

$$P \left\{ \lim_{k \rightarrow \infty} \lim_{t \rightarrow 0} f(X^{(k)}(t)) = f^* \right\} = 1. \quad (21)$$

定理证毕.

引理 3. 在假设 1 的前提下, 若新解产生满足 $\Delta \sim [-b_i, b_i]$, 其中 $b_i = x_i^U - x_i^D$, 则:

$$\begin{aligned} q_{01} &= \lim_{t \rightarrow 0} p_{01} = 0, q_{00} = \lim_{t \rightarrow 0} p_{00} = 1 \\ q_{11} &= \lim_{t \rightarrow 0} p_{11} \leq c \in (0, 1), q_{10} = \lim_{t \rightarrow 0} p_{10} \geq 1 - c \in (0, 1) \end{aligned}$$

证明. 当 $X^{(k)}(t) \in D_0$ 时, $X^{(k+1)}(t) \in D_1$ 的概率为

$$P \left(X^{(k+1)}(t) \in D_1 \mid X^{(k)}(t) \in D_0 \right) \leq \sum_{i=1}^n \frac{1}{n} \int_{(D_1)_i} \frac{2}{2b_i} e^{-\frac{f(X^{(k+1)}(t)) - f(X^{(k)}(t))}{t}} dx'_i, \quad (22)$$

其中, $(\cdot)_i$ 代表在 i 方向上的投影. 上式是由于 $\Delta \sim [-b_i, b_i]$, 又由 $x'_i = x_i^0 + \Delta$, 从而 $x'_i \sim [x_i^0 - b_i, x_i^0 + b_i]$, 概率密度为 $\frac{1}{2b_i}$, 又由于采用的边界处理方法, 因此出现了分子上的数字 2.

在假设 1 的前提下, 即 $f(X^{(k+1)}(t)) - f(X^{(k)}(t)) \geq \tau$, 有

$$P \left(X^{(k+1)}(t) \in D_1 \mid X^{(k)}(t) \in D_0 \right) \leq e^{-\frac{\tau}{t}} \quad (23)$$

$$q_{01} = \lim_{t \rightarrow 0} p_{01} = \lim_{t \rightarrow 0} P \left(X^{(k+1)}(t) \in D_1 \mid X^{(k)}(t) \in D_0 \right) = 0$$

显然 $q_{00} = 1 - q_{01} = 1$.

当 $X^{(k)}(t) \in D_1$ 时, $X^{(k+1)}(t) \in D_1$ 的概率为

$$\begin{aligned} &P \left(X^{(k+1)}(t) \in D_1 \mid X^{(k)}(t) \in D_1 \right) \\ &\leq \sum_{i=1}^n \frac{1}{n} \int_{(D_1)_i} \frac{2}{2b_i} \min \left\{ 1, e^{-\frac{f(X^{(k+1)}(t)) - f(X^{(k)}(t))}{t}} \right\} dx'_i \\ &\leq \sum_{i=1}^n \frac{1}{n} \int_{(D_1)_i} \frac{2}{2b_i} dx'_i = c \end{aligned} \quad (24)$$

由于 $D_1 \subset [X^D, X^U]$, 则

$$P \left(X^{(k+1)}(t) \in D_1 \mid X^{(k)}(t) \in D_1 \right) \leq c \in (0, 1) \quad (25)$$

从而

$$q_{11} = \lim_{t \rightarrow 0} p_{11} \leq c \in (0, 1), q_{10} = \lim_{t \rightarrow 0} p_{10} \geq 1 - c \in (0, 1).$$

利用引理 3, 采用与定理 1 类似的证明方法, 易证下面的定理成立:

定理 2. 在假设 1 的前提下, 对于问题 (P), 设在一定温度 t 下, 由算法产生的点列为 $X^{(k+1)}(t), (k = 1, 2, \dots)$, 若新解产生满足 $\Delta \sim [-b_i, b_i]$, 其中 $b_i = x_i^U - x_i^D$, 则当温度 $t \rightarrow 0$ 时, 点列 $\{X^{(k)}(t)\}_{k=1}^{+\infty}$ 以概率为 1 收敛于 (P) 的全局极小点, 即

$$P \left(\lim_{k \rightarrow \infty} \lim_{t \rightarrow 0} f(X^{(k)}(t)) = f^* \right) = 1$$

证明. 略.

由定理 1 可见: 如果新解产生满足 $\Delta \sim N(0, \sigma^2)$, 本文算法以概率为 1 收敛于全局最优解. 若采用第二种新解产生方法, 即 $\Delta = rand \bullet scale \bullet (x_r^U - x_r^D)$, 当 $Scale = 1$ 时, Δ 服从均匀分布, 类似地, 可以证明算法的全局收敛性; 当 $Scale < 1$ 时, 新解被限制在当前解的一定邻域范围内, 此时关于算法的全局收敛性还有待进一步研究.

4. 数值试验

为了验证算法的有效性, 选取了八个典型的全局优化测试函数, 函数的维数从 2 维到 5000 维, 测试函数见附录.

表 1 列出的是采用新解产生方法 a, 即 $\Delta \sim N(0, \sigma^2)$ 的数值试验结果; 表 2 列出了采用新解产生方法 b, 即 $\Delta = rand \bullet scale \bullet (x_r^U - x_r^D)$ 得到的数值试验结果. 对每个函数进行了 20 - 50 次计算, 取函数平均调用次数和成功率为评价指标.

表 1 数值试验结果 (新解产生 a: $\Delta \sim N(0, \sigma^2)$)

算例	维数	Tmax	Lmax	dt	fai	平均调用次数	成功率, %	最优值	精度
F1	2	0.1	10	0.98	8	673	90	-0.998	10^{-4}
F2	3	100	6	0.9	1	778	80	0	10^{-4}
F3	3	10	50	0.96	8	6,627	41	-496.1761	10^{-2}
F4	6	500	5	0.8	0.8	994	75	-3.3223	10^{-4}
F5	10	100	5	0.9	5	10,734	39	0	10^{-3}
F6	100	1000	100	0.95	8	91,720	100	-78.3323	10^{-4}
F7(n=100)	100	200	100	0.9	0.8	57,500	80	0	10^{-4}
F7(n=1000)	1000	200	500	0.9	5	542,500	100	0	10^{-4}
F7(n=3000)	3000	200	1500	0.9	5	964,000	100	0	10^{-4}
F7(n=5000)	5000	200	2500	0.9	5	4,062,500	100	0	10^{-4}
F8(6 个原子)	18	10	80	0.95	1.5	26,810	100	-12.712062	10^{-2}
F8(10 个原子)	30	50	300	0.96	1.5	117,076	25	-28.422532	10^{-2}
F8(10 个原子)*	30	50	300	0.96	1.5	108,129	100	-28.422532	10^{-5}

* 为结合应用局部极小化策略 (Hooke-Jeeves 方法), 进行 50 次计算的结果

表 2 数值试验结果 (新解产生 b: $\Delta = rand \bullet scale \bullet (x_r^U - x_r^D)$)

算例	维数	Tmax	Lmax	dt	scale	平均调用次数	成功率, %	最优值	精度
F1	2	0.05	5	0.9	0.5	484	100	-0.998	10^{-4}
F2	3	500	5	0.8	0.2	382	100	0	10^{-4}
F3	3	10	30	0.96	1	3,662	47	-496.1761	10^{-2}
F4	6	100	20	0.9	0.5	4,947	70	-3.3223	10^{-4}
F5	10	1000	50	0.95	0.8	2,071	70	0	10^{-3}
F6	100	600	100	0.95	0.2	15,060	100	-78.3323	10^{-4}
F7(n=100)	100	500	50	0.95	0.2	24,180	100	0	10^{-4}
F7(n=1000)	1000	500	400	0.96	0.2	218,755	90	0	10^{-4}
F7(n=3000)	3000	200	1500	0.9	0.2	1,192,867	100	0	10^{-4}
F7(n=5000)	5000	200	2500	0.9	0.2	46,488,500	100	0	10^{-4}

从数值试验看出, 该算法具有较好的全局收敛性, 对所有的测试函数, 在一定的成功率下都能找到全局极小点. 算法的收敛速度主要与问题的规模及函数性质有关, 对于大多数 100

维以下函数而言, 算法能快速收敛到全局最优点, 随着维数的增加, 例如 F7, 尽管函数调用次数增加了, 算法成功率并没有降低. 算法收敛的成功率主要取决于函数本身的性质, 如果相对于局部极小点, 全局最优点所在的势阱很窄, 而且局部极小点与全局极小点函数值相差不多, 例如 F3 和 F5, 此时即使在低维情况下, 算法也容易陷入局部较小, 成功率比较低. 比较两种不同的新解产生方法, 容易看出在低维情况下, 新解产生方法 b 的收敛速度比方法 a 更快, 然而随着维数的增加, 这种现象发生了变化, 以 F7 为例, 对于 1000 维以上的情况, 采用新解产生方法 a, 函数平均调用次数更少, 说明采用正态分布产生新解的方法比较稳定.

Lennard-Jones 簇集能量极小化在分子构象研究中具有重要的实际意义, 另外, 由于 LJ 簇集势函数是一个描述简单, 求解困难的全局优化问题, 是检验全局优化方法的一个典型例题 (见附录 F8). 据估计, N 个原子的 LJ 函数的局部极小点不少于 $O(e^N)$. 通常用 LJn 表示 n 个原子系统的 Lennard-Jones 簇集问题, 我们采用新解产生方法 a, 对 LJ3-LJ10 进行了计算, (表中仅列出了 LJ6 和 LJ10), 当原子数增加到 8 时, 算法成功率开始下降, 对于 10 个原子的系统, 我们计算了 30 次, 成功率仅约为 25%. 考虑到函数是可微的, 在算法中加入了局部极小化过程, 成功率提高到了 100%, 然而, 局部极小化本身是一个计算时间比较长的过程, 随着问题规模的增大, 这个过程耗时也增加, 尽管考虑局部极小化后函数仅迭代几十步就可以收敛到全局极小点, 计算时间却与原来相差不多.

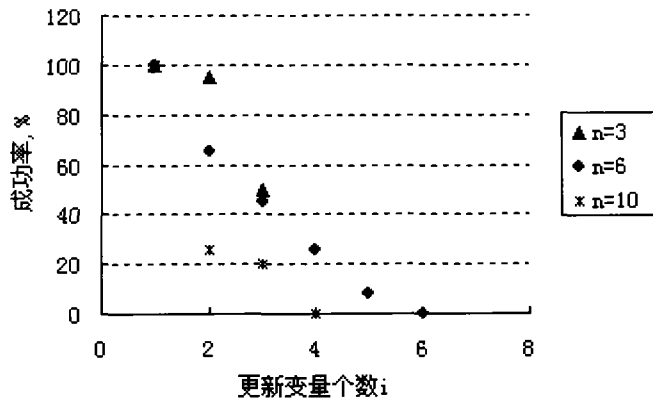


图 1 更新变量个数对成功率的影响

本文算法的一个特点是每次只更新一个变量, 尤其对于高维问题, 如果一次更新多个变量, 很难在所有方向同时达到极小点. 为了比较一次更新一个变量和一次更新多个变量对算法收敛效果的影响, 用 F2 函数取不同的 n 值进行计算, 结果见图 1. 图中, 三角形点代表目标函数变量数为 $n=3$ 的情况, 一次更新一个变量, 成功率达 100%, 一次更新两个变量, 成功率为 95%, 一次更新所有变量, 成功率只有 50%; 圆点代表 $n=6$ 的情况, 一次更新一个变量, 成功率为 100%, 随着更新变量数增加, 成功率降低, 当一次更新所有变量时, 便不再收敛; 星状点代表 $n=10$ 的情况, 一次更新一个变量, 成功率为 100%, 然而成功率随更新变量数的增

加迅速降低, 当一次更新 4 个变量时, 便得不到收敛结果. 利用算例 F6 进行计算, 采用新解产生方法 a, 当一次更新一个变量时, 函数调用次数为 91720, 一次更新两个变量则需要 500 000 次以上; 采用新解产生方法 b, 一次更新一个变量, 函数调用次数为 15060, 一次更新两个变量时, 则得不到收敛结果. 可以看出一次更新一个变量具有更好的效果.

5. 结 论

本文给出了一类适用于连续函数全局优化问题的模拟退火算法, 分析了算法的收敛性, 数值试验表明, 该算法对于高维的全局优化问题也是可行有效的. 关于算法的使用我们有以下几点建议:

1) 根据问题的不同可以选取不同的新解产生方法, 不仅仅限于本文给出的两种方法; 对于本文给出的两种方法, 解决低维问题应优先考虑采用新解产生方法 b, 对高维问题则应优先考虑新解产生方法 a;

2) 参数的调整对于算法的收敛速度有重要影响;

3) 在解决实际问题时考虑局部极小化可以提高成功率, 然而, 局部极小化耗费的时间也应该给予考虑. 可以根据不同问题, 确定不同的局部极小化策略.

4) 对于全局极小势阱很陡的函数, 算法收敛的成功率相对较低.

5) 关于温度下降函数, 本文采用简单的线性函数, 也可以考虑采用其它形式的温度下降方式, 只要保证趋近于 0, 同样可以得到全局收敛的结果.

附录: 测试函数 [13-18]

F1: De Jong 函数

$$\min f(X) = 0.002 - \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6}, \quad -65.536 \leq x_i \leq 65.536, \quad i = 1, 2.$$

其中,

$$\begin{aligned} a_{1j} &= \{-32, -16, 0, 16, 32, -32, -16, 0, 16, 32, -32, -16, 0, 16, 32, \\ &\quad -32, -16, 0, 16, 32, -32, -16, 0, 16, 32\}; \\ a_{2j} &= \{-32, -32, -32, -32, -32, -16, -16, -16, -16, \\ &\quad -16, 0, 0, 0, 0, 0, 16, 16, 16, 16, 16, 32, 32, 32, 32\} \end{aligned}$$

这是一个多峰函数, 有 25 个局部极小点, 其中有一个是全局极小点, 全局极小值为: $f^* = f(-32, -32) = -0.998$.

F2:

$$\min f(X) = \frac{\pi}{n} \{k_1 \sin^2(\pi y_1) + \sum_{i=1}^{n-1} (y_i - k_2)^2 [1 + k_1 \sin^2(\pi y_{i+1})] + (y_n - k_2)^2\}, \quad n = 3$$

$$-10 \leq x_i \leq 10, \quad i = 1, 2, \dots, n$$

其中, $y_i = 1 + \frac{1}{4}(x_i + 1)$, $k_1 = 10$, $k_2 = 1$.

在可行域内有大约 5^3 个局部极小点, 全局最优值为: $f^* = 0$.

F3:

$$\min f(X) = - \left| \prod_{i=1}^n x_i \times \sin(x_i) \right|, x_i \in [0, 3\pi], i = 1, 2, \dots, 300,$$

最优解为: $x^* = (7.98, 7.98, \dots, 7.98)^T$.

F4: Hartmann function

$$\min f(X) = - \sum_{i=1}^4 c_i \exp \left[- \sum_{j=1}^6 a_{ij} (x_j - p_{ij})^2 \right], \quad 0 < x_j < 1, \quad j = 1, 2, \dots, 6$$

其中:

i	a_{ij}						c_i	p_{ij}					
1	10.0	3.00	17.0	3.50	1.70	8.00	1.0	0.1312	0.1696	0.5569	0.0124	0.8283	0.5886
2	0.05	10.0	17.0	0.10	8.00	14.0	1.2	0.2329	0.4135	0.8307	0.3736	0.1004	0.9991
3	3.00	3.50	1.70	10.0	17.0	8.00	3.0	0.2348	0.1451	0.3522	0.2883	0.3047	0.6650
4	17.0	8.00	0.05	10.0	0.10	14.0	3.2	0.4047	0.8828	0.8732	0.5743	0.1091	0.0381

四个局部极小点, 近似等于 $p_i = (p_{i1}, \dots, p_{i6})^T, i = 1, 2, 3, 4; f(p_i) \approx -c_i; x = (0.201, 0.150, 0.477, 0.275, 0.311, 0.657)^T$ 为全局极小点, $f^* = -3.3223$.

F5: Gn10

$$\min f(X) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, \quad -600 \leq x_i \leq 600,$$

有几千个极小点, 全局极小值为: $f^* = f(0) = 0$.

F6: General test function

$$\min f(X) = \frac{1}{100} \sum_{i=1}^{100} (x_i^4 - 16x_i^2 + 5x_i), \quad -10 \leq x_i \leq 10,$$

函数在可行域内有 2^{100} 个局部最优解, 全局最优值为 $f^* = -78.3323$.

F7: Rastrigin's Function

$$\min f(X) = n + \sum_{i=1}^n [x_i^2 - \cos(2\pi x_i)], \quad -5.12 \leq x_i \leq 5.12, \quad i = 1, 2, \dots, n$$

$f(X)$ 为极多峰函数, 分别取 $n = 10, 20, 100, 1000 \dots$ 进行计算, 全局最优值为: $f^* = f(0) = 0$.

F8: Lennard-Jones 簇问题 [13]

LJ 函数描述一个多原子系统中原子对之间的相互作用, 不考虑原子性质的差异和多原子之间的相互作用. 对于包含 N 个原子的原子簇, LJ 函数可表示为:

$$E = \sum_{i=1}^N \sum_{j=1}^{i-1} \left[\frac{1}{r_{ij}^{12}} - 2 \cdot \frac{1}{r_{ij}^6} \right],$$

其中: r_{ij} 为原子 i 和原子 j 之间的距离.

参 考 文 献

- [1] S.Kirkpatrick, J.C.D.Gelat, M.P.Vecchi, Optimization by simulated annealing, *Science*, 220(1983), 671-680.
- [2] 康立山, 谢云, 尤矢勇, 罗祖华等, 非数值并行算法 - 模拟退火算法, 科学出版社, 1994.
- [3] 邢文训, 谢金星, 现代优化计算方法, 清华大学出版社, 1999.
- [4] M.M.Ali, A.Torn, S.Viitanen, A direct search variant of the simulated algorithm for optimization involving continuous variables, *Computers & Operation Research*, 29(2002), 87-102.
- [5] R.L.Yang, Convergence of the simulated annealing algorithm for continuous global optimization, *Journal of Optimization Theory and Applications*, 104(2000), 691-716.
- [6] M. Locatelli, Simulated annealing algorithms for continuous global optimization: convergence conditions, *Journal of Optimization Theory and Applications*, 104(2000), 121-133.
- [7] 刘勇, 龚光鲁, 钱敏平, R^d 上具有间断梯度的势函数的模拟退火, 中国科学 (A 辑), 31(2002), 97-103.
- [8] 李宏, 唐焕文, 郭崇慧, 一类进化策略的收敛性分析, 运筹学学报, 3:4(1999), 79-83.
- [9] 郭崇慧, 唐焕文, 演化策略的全局收敛性, 计算数学, 23:1(2001), 105-110.
- [10] 郭崇慧, 唐焕文, 一种改进的进化规划算法及其收敛性, 高等学校计算数学学报, 1(2002), 51-56.
- [11] 靳利霞, 唐焕文, 模拟退火算法的一种改进及其在蛋白质结构预测中的应用, 系统工程理论与实践, 22:9(2002), 92-96.
- [12] 梁之舜, 邓集贤等, 概率论与数理统计, 高等教育出版社, 北京, 1988.
- [13] R.H. Leary, Global optima of Lennard-Jones clusters, *Journal of Global Optimization*, 11(1997), 35-53.
- [14] 杨若黎, 顾基发, 一种高效的模拟退火全局优化算法, 系统工程理论与实践, 17:5(1997), 29-35.
- [15] 何险峰, 周家驹, 遗传算法的初步研究及改进后的遗传算法程序 IGA1.0, 计算机与应用化学, 15:4(1998), 211-216.
- [16] T. Trafalis, S.Kasap, A novel metaheuristics approach for continuous global optimization, *Journal of Global Optimization*, 23(2002), 171-190.
- [17] R. Chelouch, P. Siarry, Tabu search applied to global optimization, *Computers & Operation Research*, 123(2000), 256-270.
- [18] 李敏强, 寇纪淞, 林丹, 李书全, 遗传算法的基本理论与应用, 科学出版社, 2002.
- [19] K.F.C. YIU, Y. LIU, K.L. TEO, A hybrid descent method for global optimization, *Journal of Global Optimization*, 28(2004), 229-238.
- [20] Mingjun Ji, Huanwen Tang, Application of chaos in simulated annealing, *Chaos, Solitons & Fractals*, 21(2004), 933-941.