

一种高效的模拟退火全局优化算法^{*}

杨若黎 顾基发

(中国科学院系统科学研究所, 北京 100080)

摘要 提出了一种确定模拟退火算法温度更新函数的启发式准则, 构造了适当的产生随机向量的概率密度函数, 应用该启发式准则导出了相应的温度更新函数。新的温度更新函数与退火时间的幂函数成反比, 与优化问题的变量维数无关。数值计算结果表明, 采用新的温度更新函数及相应的概率密度函数的模拟退火算法可以显著地提高求解全局优化问题的计算效率。

关键词 模拟退火 全局优化 随机搜索

An Efficient Simulated Annealing Algorithm for Global Optimization

Yang Ruoli Gu Jifa

(Institute of Systems Science, Chinese Academy of Sciences, Beijing 100080)

Abstract A heuristic criterion for determining the temperature updating function of simulated annealing algorithm is proposed in this paper. An appropriate form of probability density function for generating the random vectors is constructed. The temperature updating function corresponding to the probability density function is derived by using the proposed heuristic criterion. The new temperature updating function derived is inversely proportional to a power function of the annealing time and is independent of the dimension of the optimization problems. The numerical computation results indicate that the simulated annealing algorithm with the new temperature updating function and the corresponding probability density function can improve significantly the computational efficiency for solving the global optimization problems.

Keywords simulated annealing; global optimization; random search

1 引言

许多实际优化问题的目标函数都是非凸的, 存在许多局部最优解, 但是, 有效地求出一组非凸目标函数的全局最优解至今仍是一个难题。特别是随着优化问题规模的增大, 局部最优解的数目将会迅速增加, 这无疑使寻找大规模复杂系统优化问题的全局最优解变得更加困难。

求解全局优化问题的方法可分为两类^[1], 一类是确定性方法, 另一类是随机性方法。前者基于确定性的搜索策略, 在目标函数满足特定的限制条件下可以对求得全局最优解提供确定性的保证, 这类方法一般适用于求解满足特定要求的一些特殊问题。后者在搜索策略中引入了适当的随机因素, 对目标函数一般不需要特殊的限制条件, 具有比较广泛的适用性, 由于采用随机搜索策略, 这类方法只能在概率的意义上为求得全局最优解提供保证。

模拟退火算法^[2,3]是 80 年代初期发展起来的一种求解大规模组合优化问题的随机性方法。它以优化

^{*} 本文于 1996 年 10 月 14 日收到
本文得到国家自然科学基金资助(N. 69474033)

问题的求解与物理系统退火过程的相似性为基础,利用Metropolis算法^[4]并适当地控制温度的下降过程实现模拟退火,从而达到求解全局优化问题的目的。模拟退火算法在求解TSP、VLSI电路设计等NP-完全的组合优化问题上取得了令人满意的结果^[5,6],并应用于求解连续变量函数的全局优化问题^[7~9]。它具有适用范围广,求得全局最优解的可靠性高,算法简单,便于实现等优点。模拟退火算法在搜索策略上与传统的随机搜索方法不同,它不仅引入了适当的随机因素,而且还引入了物理系统退火过程的自然机理。这种自然机理的引入使模拟退火算法在迭代过程中不仅接受使目标函数值变“好”的试探点,而且还能够以一定的概率接受使目标函数值变“差”的试探点,接受概率随着温度的下降逐渐减小。模拟退火算法的这种搜索策略有利于避免搜索过程因陷入局部最优解而无法自拔的弊端,有利于提高求得全局最优解的可靠性。

本文讨论求解如下连续全局优化问题的模拟退火算法

$$\min_{x \in R^n} f(x) \quad (1)$$

式中 $f: R^n \rightarrow R$ 是目标函数。模拟退火算法的性能主要取决于控制温度下降过程的温度更新函数。温度更新函数的确定不仅与优化问题本身的特性有关,而且与产生随机向量的概率密度函数有关。由于求解问题(1)的模拟退火算法的收敛性理论^[10,11]没有给出如何确定适当的温度更新函数的方法,所以,在实际应用中一般是根据所采用的产生随机向量的概率密度函数按照某种启发式方法确定适当的温度更新函数。文[12~13]提出了一种根据产生随机向量的概率密度函数确定温度更新函数的启发式准则,并对正态分布和柯西分布分别导出了相应的温度更新函数,前者与退火时间的对数函数成反比,后者与退火时间成反比。文[14]应用该启发式准则对各分量相互独立的多维柯西分布和文[14]中所提出的一种概率密度函数分别导出了相应的温度更新函数,前者与退火时间的 $1/n$ 次方成反比,后者与退火时间的 $1/n$ 次方的指数函数成反比,其中 n 是优化问题的变量维数。由于温度更新函数中包含退火时间的 $1/n$ 次方,当 n 较大时温度下降得比较缓慢。为了提高模拟退火算法的计算效率,文[15]提出在上述指数形式的温度更新函数中,可将退火时间的 $1/n$ 次方改为 q/n 次方,其中 $q > 1$ 。实际计算结果表明,当 $q = n$ 时,采用指数形式温度更新函数的模拟退火算法的计算效率可以有明显的提高。

本文提出了一种新的确定温度更新函数的启发式准则,通过构造适当的产生随机向量的概率密度函数,使应用该启发式准则导出的温度更新函数不含有退火时间的 $1/n$ 次方。新的温度更新函数与退火时间的 m 次方成反比,其中 $m \geq 1$ 是给定的常数。数值计算结果表明,采用新的温度更新函数及相应的概率密度函数的模拟退火算法可以进一步提高求解全局优化问题的计算效率,是一种高效的模拟退火全局优化算法。

2 模拟退火算法

求解全局优化问题(1)的模拟退火算法的步骤如下:

Step1 给定初始解 $x^0 \in R^n$ 和初始温度 $T_0 > 0$,给定产生随机向量的概率密度函数和控制温度下降过程的温度更新函数,给定常数 $\beta > 0$,计算 $f(x^0)$,置 $X^0 = x^0, X_{\min} = x^0, f_{\min} = f(x^0), k = 0$ 。

Step2 根据给定的概率密度函数产生一个随机向量 Z^k ,利用当前迭代点 X^k 和随机向量 Z^k 产生一个新的试探点 Y^k ,即

$$Y^k = X^k + Z^k \quad (2)$$

计算 $f(Y^k)$ 。

Step3 产生一个在 $(0, 1)$ 上均匀分布的随机数 η ,计算在给定当前迭代点 X^k 和温度 T_k 下接受试探点 Y^k 的概率 $P_a(Y^k | X^k, T_k)$,即

$$P_a(Y^k | X^k, T_k) = \min \left\{ 1, \exp \left[\frac{f(X^k) - f(Y^k)}{\beta T_k} \right] \right\}, \quad (3)$$

如果 $\eta \leq P_a(Y^k | X^k, T_k)$,则置 $X^{k+1} = Y^k, f(X^{k+1}) = f(Y^k)$;否则置 $X^{k+1} = X^k, f(X^{k+1}) = f(X^k)$ 。

Step4 如果 $f(X^{k+1}) < f_{\min}$,则置 $X_{\min} = X^{k+1}, f_{\min} = f(X^{k+1})$ 。

Step5 如果迭代终止条件满足, 则算法结束, X_{\min} 就作为近似的全局最优解, f_{\min} 为相应的最优值; 否则继续 Step6.

Step6 根据给定的温度更新函数产生一个新的温度 T_{k+1} , 置 $k = k + 1$, 转至 Step2.

在上述算法步骤中, 常数 β 的作用是对目标函数进行适当的比例变换, 控制目标函数值的变化范围. 需要指出, 上述模拟退火算法只有一层循环, 在每一温度下只产生一个随机的试探点, 而通常的模拟退火算法有两层循环, 内循环在同一温度下产生一组随机的试探点, 外循环更新温度.

3 确定温度更新函数的启发式准则

基于梯度的各种优化算法易于陷入局部最优解的主要原因是因为它们只在当前迭代点的领域内利用梯度信息进行局部搜索, 没有在大范围内进行搜索的策略. 由于单独进行大范围搜索将会降低计算效率和解的精度, 所以求解全局优化问题的随机搜索算法一般都采用大范围的粗略搜索与局部的精细搜索相结合的搜索策略. 在随机搜索算法的迭代过程中, 首先按照一定的概率分布在较大的范围内随机地产生试探点, 以实现大范围的粗略搜索, 然后逐步缩小随机产生试探点的范围, 使搜索过程逐渐变为局部的精细搜索. 这种搜索策略的基本思想是, 在搜索过程的初始阶段找到全局最优解所在的一个局部区域, 然后逐渐缩小搜索范围, 最终求出全局最优解. 如果在初始的大范围搜索阶段没有找到全局最优解所在的区域, 在后面的局部搜索阶段将可能陷入局部最优解, 因为在局部搜索阶段随机产生的所有试探点将会逐渐集中在当前迭代点的局部范围内.

模拟退火算法是通过适当地控制温度的变化过程实现大范围的粗略搜索与局部的精细搜索相结合的搜索策略. 由于产生随机向量的概率密度函数以及接受试探点为新的当前迭代点的接受概率都与温度有关, 所以, 当温度较高时, 随机产生的试探点的散布范围较大, 并且能够以较大的概率接受使目标函数值增加的试探点, 从而可实现大范围搜索. 随着温度逐渐下降, 随机产生的试探点的散布范围逐渐减小, 接受使目标函数值增加的试探点的概率也逐渐减小, 从而使搜索过程逐渐变为局部探索. 如果随着温度逐渐下降, 随机产生的试探点越来越集中在当前迭代点的局部范围内, 那么, 在温度较低时, 模拟退火算法将近似于传统的随机搜索算法, 因而一旦陷入局部最优解将很难逃脱出来. 因此, 为了提高模拟退火算法求得全局最优解的可靠性和计算效率, 一方面要保持适当的温度下降速度, 另一方面要使产生的随机向量保持一定的散布程度, 使随机产生的试探点不能都集中在当前迭代点的局部范围内. 当产生随机向量的概率密度函数与温度有关时, 温度更新函数不仅决定了温度的下降速度, 而且决定了在整个退火过程中所产生的随机向量的散布程度. 因此, 我们可得出如下确定温度更新函数的一种启发式准则: 对于任意给定的包含坐标原点的有界区域 $A \subset R^n$ 以及任意给定的 $k_0 > 0$, 温度更新函数应使在所有退火时间 $k \geq k_0$ 所产生的随机向量全部落在区域 A 内的概率等于零.

4 随机向量的产生

为了构造产生随机向量的概率密度函数, 我们首先定义如下的一维条件概率密度函数:

$$g_1(v|T) = \frac{T^{1/m}}{2m(|v| + T)^{(m+1)/m}}, \quad v \in R \quad (4)$$

式中 $T > 0$ 是给定的参数, $m \geq 1$ 是给定的常数. 假定模拟退火算法在任意退火时间 k 所产生的随机向量 z^k 的各个分量相互独立, 并且每一分量的概率密度函数都由 (4) 式定义, 那么, 在给定温度 T_k 下产生随机向量 z^k 的条件概率密度函数 $p_1(\cdot | T_k)$ 可表示为:

$$p_1(z^k | T_k) = \prod_{i=1}^n g_1(z_i^k | T_k) \quad (5)$$

式中 $z^k = (z_1^k, z_2^k, \dots, z_n^k)$. 具有上述条件概率密度函数的随机向量 z^k 可通过下式产生:

$$z_i^k = \text{sign}(U_i) T_k \left(\frac{1}{|U_i|^m} - 1 \right), \quad i = 1, 2, \dots, n \quad (6)$$

式中 Z_i^k 是随机向量 Z^k 的第 i 个分量, U_1, U_2, \dots, U_n 是一组两两相互独立的在 $[-1, 1]$ 上均匀分布的随机变量, $\text{sign}(\cdot)$ 是符号函数。

除了采用上述定义的条件概率密度函数产生随机向量 Z^k 以外, 我们还可以利用已有概率分布的随机变量的函数产生随机向量 Z^k 。为此, 我们定义如下的一维条件概率密度函数:

$$g_2(v|T) = \begin{cases} \frac{T^{1/m}}{m(v+T)^{(m+1)/m}} & v \geq 0 \\ 0 & v < 0 \end{cases} \quad (7)$$

式中 $v \in R, T > 0$ 是给定的参数, $m \geq 1$ 是给定的常数。设 $V_1^k, V_2^k, \dots, V_n^k$ 是一组具有概率密度函数 $g_2(\cdot | T_k)$ 的随机变量, W_1, W_2, \dots, W_n 是一组在 $[-1, 1]$ 上均匀分布的随机变量, $V_1^k, V_2^k, \dots, V_n^k, W_1, W_2, \dots, W_n$ 两两相互独立, 则模拟退火算法在任意退火时间 k 所产生的随机向量 Z^k 可由如下一组随机变量的函数定义:

$$Z_i^k = \frac{W_i V_i^k}{\sqrt{\sum_{j=1}^n W_j^2}}, \quad i = 1, 2, \dots, n \quad (8)$$

式中 Z_i^k 是随机向量 Z^k 的第 i 个分量。假定对所有的 $i \in \{1, 2, \dots, n\}, Z_i^k \geq 0$, 那么, 根据随机变量函数的概率分布, 可得随机向量 Z^k 在给定温度 T_k 下的条件概率密度函数 $p_2(\cdot | T_k)$ 为

$$p_2(Z^k | T_k) = \frac{1}{2^n} \prod_{i=1}^n g_2\left(\frac{Z_i^k}{s_i} \sqrt{\sum_{j=1}^n s_j^2} \middle| T_k\right) = \frac{1}{s_i} \sqrt{\sum_{j=1}^n s_j^2} d s_1 \dots d s_n, \quad Z_i^k \geq 0, i = 1, 2, \dots, n \quad (9)$$

式中 $Z^k = (Z_1^k, Z_2^k, \dots, Z_n^k)$ 。由于具有概率密度函数 $g_2(\cdot | T_k)$ 的随机变量 V_i^k 可通过下式产生:

$$V_i^k = T_k \left(\frac{1}{U_i^m} - 1 \right), \quad i = 1, 2, \dots, n \quad (10)$$

式中 U_1, U_2, \dots, U_n 是一组两两相互独立的在 $[0, 1]$ 上均匀分布的随机变量, 所以, 由随机变量函数 (8) 所定义的随机向量 Z^k 可通过下式产生:

$$Z_i^k = \frac{W_i T_k}{\sqrt{\sum_{j=1}^n W_j^2}} \left(\frac{1}{U_i^m} - 1 \right), \quad i = 1, 2, \dots, n \quad (11)$$

根据上述定义的产生随机向量的概率密度函数, 利用上述启发式准则, 可确定相应的温度更新函数。

5 温度更新函数

下面定理给出了在上述启发式准则的意义下概率密度函数 (5) 所对应的温度更新函数。

定理 1 模拟退火算法中产生随机向量的概率密度函数由 (5) 式定义, 假定在不同退火时间所产生的随机向量相互独立, 那么, 如果温度更新函数为

$$T_k = \frac{T_0}{k^m}, \quad k = 1, 2, \dots \quad (12)$$

式中 $T_0 > 0$ 是初始温度, $m \geq 1$ 与 (4) 式定义的一维概率密度函数 $g_1(\cdot | T_k)$ 中的 m 相同, 则对于任意给定的包含坐标原点的有界区域 $A \subset R^n$ 以及任意给定的 $k_0 > 0$, 在所有退火时间 $k \geq k_0$ 所产生的随机向量 Z^k 全部落在区域 A 内的概率等于零。

证明 对于任意给定的实数 $\alpha > 0$, 令 $A = \{Z \in R^n | Z_i \leq \alpha, i = 1, 2, \dots, n\}$ 。在任意的退火时间 k 与温度 T_k 下, 由概率密度函数 (5) 所产生的随机向量 Z^k 落在区域 A 内的概率为

$$\begin{aligned} P(Z^k \in A) &= \int_0^\alpha \dots \int_0^\alpha \prod_{i=1}^n \frac{(T_k)^{1/m}}{2m(|z_i^k| + T_k)^{(m+1)/m}} dz_1^k \dots dz_n^k \\ &= \prod_{i=1}^n \int_0^\alpha \frac{(T_k)^{1/m}}{2m(z_i^k + T_k)^{(m+1)/m}} dz_i^k \\ &\leq \int_0^\alpha \frac{(T_k)^{1/m}}{2m(\zeta + T_k)^{(m+1)/m}} d\zeta \end{aligned}$$

$$\begin{aligned}
 &= 1 - \left(\frac{T_k}{\alpha + T_k} \right)^{1/m} \\
 &\leq 1 - \left(\frac{T_k}{\alpha + T_0} \right)^{1/m}
 \end{aligned}$$

对于任意给定的 $k_0 > 0$, 在退火时间 $k \geq k_0$ 内所产生的随机向量 Z^k 全部落在区域 A 内的概率为

$$\begin{aligned}
 \prod_{k=k_0} P(Z^k \in A) &\leq \prod_{k=k_0} \left[1 - \left(\frac{T_k}{\alpha + T_0} \right)^{1/m} \right] \\
 &= \exp \left\{ \ln \prod_{k=k_0} \left[1 - \left(\frac{T_k}{\alpha + T_0} \right)^{1/m} \right] \right\} \\
 &= \exp \left\{ \sum_{k=k_0} \ln \left[1 - \left(\frac{T_k}{\alpha + T_0} \right)^{1/m} \right] \right\}
 \end{aligned}$$

利用不等式 $\ln(1-y) \leq -y, 0 \leq y < 1$, 以及由 (12) 式定义的温度更新函数, 上式可得

$$\begin{aligned}
 \prod_{k=k_0} P(Z^k \in A) &\leq \exp \left\{ - \sum_{k=k_0} \left(\frac{T_k}{\alpha + T_0} \right)^{1/m} \right\} \\
 &= \exp \left\{ - \sum_{k=k_0} \left(\frac{T_0}{\alpha + T_0} \right)^{1/m} \frac{1}{k} \right\} \\
 &= \exp \left\{ - \left(\frac{T_0}{\alpha + T_0} \right)^{1/m} \sum_{k=k_0} \frac{1}{k} \right\} \\
 &= 0
 \end{aligned}$$

因此, 定理的结论成立。

同样可以证明, (12) 式也是概率密度函数 (9) 所对应的温度更新函数。

通常的模拟退火算法所采用的温度更新函数为 $T_k = \alpha_k T_{k-1}$, $0 < \alpha_k < 1, k \geq 1$, 根据与物理系统退火过程的类比关系可得, 在温度较高时, α_k 可以取较小的值, 使温度有较快的下降速度, 随着温度逐渐减少, α_k 的值应逐渐增大, 使温度的下降速度逐渐减慢, 当 T_k 接近于零时, α_k 应接近于 1。定理 1 给出的温度更新函数可以表示为 $T_1 = T_0, T_k = \left(\frac{k-1}{k} \right)^m T_{k-1}, k \geq 2$, 相当于 $\alpha_k = 1, \alpha_k = \left(\frac{k-1}{k} \right)^m, k \geq 2$ 。可以看出, 当 $k \geq 2$ 时, 随着退火时间 (即迭代次数) k 的增加, α_k 的值逐渐增大, 当 $k \rightarrow \infty$ 时, $\alpha_k \rightarrow 1$, 并且 m 的值越小, α_k 趋近于 1 的速度越快, 温度的下降速度越慢。由此可见, 定理 1 给出的温度更新函数符合与物理系统退火过程的类比关系, 而且通过适当地选择 m 的值可以控制温度的下降速度。

在前面给出的模拟退火算法步骤中, 采用定理 1 给出的温度更新函数及相应的产生随机向量的概率密度函数可得到一种高效的拟退火全局优化算法。

6 边界约束的处理方法

对于具有边界约束的全局优化问题:

$$\begin{aligned}
 &\min_x f(x) \\
 &\text{s.t. } a_i \leq x_i \leq b_i, i = 1, 2, \dots, n
 \end{aligned} \tag{13}$$

式中 $x = (x_1, x_2, \dots, x_n)$, a_i 和 b_i 分别是 x_i 的下界和上界。为了使模拟退火算法随机产生的试探点满足给定的边界约束条件, 可采用如下方法对上述模拟退火算法的 Step 2 做适当的修正。第一种方法是对每个新产生的试探点先检查是否满足给定的边界约束条件, 如果满足, 则计算该试探点的目标函数值, 否则重新产生试探点。第二种方法是先对随机产生的试探点进行适当的处理, 保证其满足给定的边界约束条件, 然后再计算相应的目标函数值。假定第 k 次迭代所产生的新的试探点记为 Y^k , 它可能不满足给定的边界约束条件, 则满足边界约束条件的试探点 Y^k 可通过下式得到:

$$Y_i^k = \begin{cases} b_i - (Y_i^k - b_i) \text{MOD} (b_i - a_i) & \text{if } Y_i^k > b_i \\ \hat{Y}_i^k, & \text{if } a_i \leq \hat{Y}_i^k \leq b_i \\ a_i + (a_i - \hat{Y}_i^k) \text{MOD} (b_i - a_i), & \text{if } Y_i^k < a_i \end{cases} \quad (14)$$

式中 \hat{Y}_i^k 和 Y_i^k 分别是试探点 \hat{Y}^k 和 Y^k 的第 i 个分量, $(a) \text{MOD} (b)$ 表示 a/b 的余数。采用第一种方法处理边界约束, 每次迭代可能要重复产生多个试探点, 采用第二种方法处理边界约束, 每次迭代只需产生一个试探点。

7 算例

考虑如下的全局优化问题:

$$\begin{aligned} \min_x f(x) &= \frac{1}{100} \sum_{i=1}^{100} (x_i^4 - 16x_i^2 + 5x_i) \\ \text{s.t. } &-10 \leq x_i \leq 10, \quad i = 1, 2, \dots, 100 \end{aligned}$$

该问题的目标函数在可行域内有 2^{100} 个局部最优解, 全局最优解的目标函数值为 -78.3323 。应用本文提出的模拟退火算法求解该全局优化问题时, 取 $T_0 = 10^7, m = 3, \beta = 1, x_i^0 = 10, 1 \leq i \leq 100$, 算法的迭代终止条件为 $|f_{\min} - f^*| < 10^{-3}$, 其中 $f^* = -78.3323$, 随机向量由 (8) 式产生, 边界约束采用 (14) 式处理。为了与本文提出的模拟退火算法进行比较, 我们应用文 [14] 提出的采用指数形式温度更新函数的模拟退火算法求解该全局优化问题, 为了克服温度更新函数中退火时间的 $1/n$ 次方的影响, 根据文 [15] 可把温度更新函数取为如下形式

$$T_k = T_0 \exp(-ck), \quad k = 1, 2, \dots \quad (15)$$

式中 $T_0 > 0$ 是初始温度, $c > 0$ 是给定的常数。对于本算例, 取 $c = 0.01$, 算法的其它参数值、初始解及迭代终止条件都与上述相同。表 1 给出了应用本文提出的模拟退火算法和文 [14] 提出的模拟退火算法求解该全局优化问题的计算结果, 其中每种算法都在使用不同的产生随机数的种子下分别运行了 10 次, 表中分别给出了每种算法在 10 次运行中所需计算目标函数值次数的最小值、平均值和最大值。

表 1 应用本文和文 [14] 的模拟退火算法求解算例的计算结果

算法类型	计算目标值次数的最小值	计算目标值次数的平均值	计算目标值次数的最大值
本文	19980	23664	27599
文 [14]	34274	38197	41853

从表 1 可以看出, 与文 [14] 提出的采用指数形式温度更新函数的模拟退火算法相比, 本文提出的模拟退火算法可以显著地减少计算目标函数值的平均次数, 提高求得全局最优解的计算效率, 是一种高效的模拟退火全局优化算法。

8 结束语

本文提出了一种确定模拟退火算法温度更新函数的启发式准则, 构造了适当的产生随机向量的概率密度函数, 并应用该启发式准则导出了相应的温度更新函数。新的温度更新函数与退火时间的 m 次方成反比, $m \geq 1$ 是给定的常数, 不含有退火时间的 $1/n$ 次方。数值计算结果表明, 采用新的温度更新函数及相应的概率密度函数的模拟退火算法可以显著地提高求解全局优化问题的计算效率, 是一种高效的模拟退火全局优化算法。

参 考 文 献

- 1 Torn A and Zilinskas A. Global Optimization, Springer-Verlag, 1989
- 2 Kirkpatrick S Gelatt, C D Jr and Vecchi M P. Optimization by Simulated Annealing Science 1983, (220): 671- 680
- 3 Cerny V. Thermodynamical Approach to the Travelling Salesman Problem: An Efficient Simulation Algorithm, Journal of Optimization Theory and Applications, 1985, (45): 41- 45
- 4 Metropolis N, Rosenbluth A W, Rosenbluth M N, Teller A H and Teller, E. Equations of State Calculation by Fast Computing Machines The Journal of Chemical Physics, 1953, (21): 1087- 1092
- 5 Laarhoven P J M van and Aarts E H L. Simulated Annealing: Theory and Applications, D Reidel Publishing Company, Dordrecht, 1987.
- 6 Aarts E H L and Korst J H M. Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing, Wiley, Chichester, 1989
- 7 Vanderbilt D and Louie S G. A Monte Carlo Simulated Annealing Approach to Optimization over Continuous Variables Journal of Computational Physics, 1984, (56): 259- 271.
- 8 Bohachevsky IO, Johnson M E and Stein M L. Generalized Simulated Annealing for Function Optimization Technometrics, 1986, (28): 209- 217.
- 9 Corana A, Marchesi M Martini C and Ridella S. Minimizing Multimodal Functions for Continuous Variables with the "Simulated Annealing" Algorithm. ACM Transactions on Mathematical Software, 1987, (13): 262- 280
- 10 Bădescu C J P. Convergence Theorems for a Class of Simulated Annealing Algorithms on R^d , Journal of Applied Probability, 1992, (29): 885- 895
- 11 Romeijn H E and Smith R L. Simulated Annealing for Constrained Global Optimization Journal of Global Optimization, 1994, (5): 101- 124
- 12 Szu H H. Non-Convex Optimization, Proc SPIE, 1986, (698): 59- 65
- 13 Szu, H H and Hartley P L. Fast Simulated Annealing Physics Letter A, 1987, (122): 157- 162
- 14 Ingber, L. Very Fast Simulated Reannealing Mathematical and Computer Modelling 1989, (12): 967 - 973
- 15 Ingber L. Simulated Annealing: Practice versus Theory. Mathematical and Computer Modelling, 1993, (18): 29- 57.