

模拟退火算法的弛豫模型与时间复杂性分析

李元香 项正龙 张伟艳

(武汉大学计算机学院 武汉 430072)

摘 要 模拟退火算法的设计思想来源于对金属高温下热运动的模拟,其理论基础是统计物理和气体运动论. 本文将模拟退火算法的运行比拟为气体的运动,基于经典的气体输运理论,特别是 BGK 弛豫方程,提出并建立了模拟退火算法弛豫时间模型. 对此模型进行整体和局部的分析,给出了退火温度和退火过程的马尔科夫链长度的理论估计. 依据理论估计,提出了预退火与全退火的两阶段退火策略,以便算法运行时对马尔科夫链长度进行动态设置. 随后,进一步分析了退火过程的时间复杂性,在此基础上,结合退火温度设置和前期建立的动力系统模型,分析了模拟退火算法的总体时间复杂性,所得的理论分析结果与早期基于大量实验分析的经验结果一致,并得到了一个有实用价值的算法停止准则. 最后,通过测试问题的求解进行实验分析与验证,实验结果表明了马尔科夫链长度动态设置法的有效性,与通常的马尔科夫链长度固定设置法比较,动态设置法的马尔科夫链总长度比固定法少 30% 以上,实验结果也表明了本文提出的算法停止准则的有效性,两方面的实验分析也有效地支持了时间复杂性分析的理论结果.

关键词 模拟退火算法;弛豫模型;时间复杂性分析;动态设置;停止准则

中图法分类号 TP301 **DOI号** 10.11897/SP.J.1016.2020.00796

A Relaxation Model and Time Complexity Analysis for Simulated Annealing Algorithm

LI Yuan-Xiang XIANG Zheng-Long, ZHANG Wei-Yan

(School of Computer Science, Wuhan University, Wuhan 430072)

Abstract Simulated annealing algorithm, which comes from the simulation of thermal motion of metal at high temperature, is of deep theoretical foundations from statistical physics and gas movement theory. Early theoretical researches for simulated annealing algorithm on convergence analysis and time complexity analysis were mainly based on the Markov chain theory in the random process, and only obtained the analysis conclusion in probability. Due to the simulated annealing algorithm has statistical physical theory foundations, which can be used to analyzing and modeling by a variety of mathematical and physical analysis methods. In this paper, the algorithm is compared to the gas movement. By comparing the searching process of the simulated annealing algorithm to the gas movement to the equilibrium state, the change of the function value during the searching of the simulated annealing algorithm can be regarded as the particles' relaxation process. Built on the classical gas transport theory, especially the BGK relaxation equation, a relaxation time model for the algorithm is proposed and established. Thus, by solving the model integrally and partially, some theoretical estimation is gained about the annealing temperature and the Markov chain length setting of the algorithm. Based on the proposed relaxation model, a two-stage annealing strategy including pre-annealing and full-annealing is raised to set

the Markov chain length dynamically along the algorithm process. At the same time, a rough estimate of the initial temperature of the algorithm is obtained. Therefore, based on the proposed dynamical Markov chain length setting, we first analyze the running time complexity of a single annealing process. Then, the relaxation time model is developed to analyze the total time complexity of the algorithm by combining the set of annealing temperature and the pre-established dynamical system models. Following the approach, the theoretical total time complexity of the simulated annealing algorithm is derived, which is consistent with the early empirical conclusion summarized from a large number of experiments. The time complexity of the algorithm is a constant related to the complexity of the problem. In addition, a novel stop criterion for stopping the algorithm process is proposed. Finally, to evaluate the theoretical analysis, several experimental studies are presented on TSP-problem and CEC2014 benchmark functions. Firstly, the results show that the Markov chain length setting by the dynamic method can be lower 30% total Markov chain length than the empirical fixed setting method, and reduce the time complexity required by the algorithm and it greatly improves the efficiency of the algorithm. Secondly, the results show that the stop criterion is available for controlling the algorithm process. Under the dynamic Markov chain setting method, the simulated annealing algorithm can stop searching in the case of multiple consecutive small Markov chain length in the iterative searching process. Thirdly, the experiments also show the time complexity of the simulated annealing, which is a constant related to the problem dimension, is consistent with the empirical conclusion. In summary, theoretical and experimental analyses effectively demonstrate that the relaxation model can be used to analyze the time complexity of the simulated annealing algorithm in theory.

Keywords simulated annealing algorithm; relaxation model; time complexity; dynamic settings; stop criterion

1 引言

模拟退火算法是一个经典的拟物类自然计算方法,核心思想是模拟金属热浴之后的缓慢降温(退火)达到能量最低态(即常温固态)的过程,以随机“跳阱”策略克服局部搜索法的缺陷,搜索优化问题的全局最优解^[1].自20世纪80年代初提出以来^[2],首先应用于求解组合优化问题,特别是对典型的NP难题,旅行商问题(TSP)的求解取得了显著成效.其后,该算法推广到系统建模与复杂数值优化问题的求解中,获得了广泛的关注和应用,促使其理论和应用研究取得了长足的进展.现在,模拟退火算法已融入以群体搜索为特征的自然与演化算法的设计与应用之中^[3-6].

理论上,基于随机过程理论将算法的搜索寻优看成随机过程,建立其马尔科夫(Markov)链模型进行收敛性分析,得到概率意义下的收敛性定理^[7-8],针对几个典型问题,分析了模拟退火算法求解这些

问题的时间复杂性^[9-11].而大量的工作则是从实际应用出发的实验研究与分析,以获得有实用价值的经验结果,从而对算法进行卓有成效的改进,大大提高其求解效果,如缓慢降温与退火时刻表的确定,回火退火策略等.这些策略的运用通常需要观察算法的运行过程与性态,并结合实际问题的特征,从数值实验中获得有用的启发信息,即从计算中认识问题,把握问题特征,由此提出与问题相适应的改进策略.这与其后兴起的以遗传算法为代表的一类演化算法,或者称为计算智能的思想一脉相承,因此,模拟退火算法及其改进策略渐渐融入以群体搜索为特征的演化算法之中,依然焕发着新的活力,有力地促进了演化算法的设计和应用研究^[12-15].

在演化算法的时间复杂性分析上,近年也有许多方法出现.文献[16]将漂移分析方法引入演化算法时间复杂性分析,通过定义距离函数来计算算法每代向最优解前进的平均距离从而获得算法的平均首次到达时间.文献[17]提出了基于吸收马尔科夫链的演化算法计算复杂性分析框架,将演化算法的

时间复杂度度量表达为马尔科夫链理论中的首次到达时间. 文献[18]通过研究两个马尔科夫链之间进行对齐映射, 提出一种通用的演化算法时间复杂性分析的方法, 可称为交替分析法 (switch analysis). 总体而言, 直到目前, 对自然与演化类算法的理论分析主要是以随机过程为理论工具, 分析的首要任务就是建立算法的马尔科夫链模型, 得到概率意义下的收敛性、时间复杂性和收敛速度.

模拟退火算法就是将目标函数的函数值比拟为物理系统的能量, 模拟物理系统趋于能量最小态 (平衡态) 的过程, 其灵魂就是 Metropolis (梅特罗玻利斯) 准则, 驱动算法跳出局部最优的“陷阱”, 保持搜索全局最优解的潜力, 控制算法的退火和运行 (参见 3.2 节算法 1 的后半部分). 该准则来自于气体运动论与统计物理学中的 Gibbs (吉布斯) 分布, 它描述了物理 (气体) 系统宏观上处于平衡态时, 各微观态出现的几率 (即统计规律), 是系统能量与温度的函数. 因此由上可知, 模拟退火算法有着明确的物理背景, 抓住这一点还可以进一步挖掘模拟退火算法的数学物理模型与理论基础, 一方面深化模拟退火算法设计与分析的理论和应用研究, 另一方面, 以简单而经典的、物理背景明确的模拟退火算法为例, 以物理学为基础, 将模拟退火算法推广应用于演化算法的设计与分析, 发展一类分析自然与演化算法的数学物理模型与方法.

事实上, 近些年在演化算法研究领域, 模拟退火算法及其物理学基础已经深入地融入一类以群体搜索为特征的群智能算法的设计、分析与应用之中. 如运用热力学与统计力学的理论建立群体演化的动力学模型, 分析群体演化的动力学特征和多样性保持策略. 文献[19]仿照模拟退火算法的退火机制, 提出了基于热动力学选择规则的遗传算法, 以统计热力学中的自由能极小定律为依据进行种群的选择和替换, 实现种群系统的能量和熵之间的动态竞争机制, 从而自适应调整算法广域探测和局部搜索之间的均衡. 文献[20]利用自由能极小原理和熵增原理来实现演化群体的寻优机制, 建立了基于分子运动论的动力学模型, 在此模型的基础上提出了一种近似最优的群体选择和替换规则. 文献[21]建立了一个多时间尺度动力系统模型, 进而提出了一类求解符号建模问题的混合演化算法. 最近, 我们沿着上述思路, 基于弹性力学原理建立了模拟退火算法的常微分方程动力系统模型^[22], 进而运用弹性力学中的极小位能原理 (与统计物理中系统的能量最小态相对

应) 分析了模拟退火算法的收敛性和收敛速度. 在动力系统理论的框架下, 获得了有理论意义和对算法的设计与改进有实用价值的结果, 并可推广这类方法对演化算法进行动力系统建模与分析.

本文将进一步发展这一思想方法, 从统计物理和非均匀气体的输运理论之中汲取思想源泉, 获得理论指导, 借鉴分析方法. 首先, 基于非均匀气体输运的理论和弛豫方程挖掘模拟退火算法更深层的物理学基础, 建立其弛豫时间模型; 然后, 对弛豫时间模型进行整体与局部分析, 获得模拟退火算法退火温度的理论估计, 以及弛豫时间或马尔科夫链长度的理论估计; 其次, 结合问题特征给出马尔科夫链长度的动态设置策略和方法, 并进行实验检验与分析, 以期获得时间复杂性分析的理论模型和实验依据; 再者, 前期运用建立的动力系统模型分析了模拟退火算法的收敛性^[22], 并给出了收敛速度估计, 本文将结合文献[22]中给出的收敛速度估计导出模拟退火算法的一个有效的停止准则; 最后, 就是基于弛豫时间模型, 在理论分析和实验分析的基础上, 对模拟退火算法进行时间复杂性的理论分析和实验评估.

本文的目的是希望以模拟退火算法为典型例子, 更深入地探究其数学物理基础, 以利借鉴推广, 发展自然与演化计算的算法设计与分析方法, 建立更广义的数学物理模型, 用于指导算法设计, 探讨自然计算的基础理论. 本文第 2 节简介非均匀气体的输运理论与弛豫方程, 进而建立模拟退火算法的弛豫时间模型, 对此模型进行整体上的分析, 获得退火温度的理论估计; 第 3 节对上述模型进行局部的分析获得弛豫时间的理论估计, 讨论马尔科夫链长度的动态设置策略和方法; 第 4 节导出算法的停止准则, 基于退火温度和退火时间的理论估计, 从总体上分析模拟退火算法的时间复杂性; 第 5 节进行实验验证与分析, 主要是马尔科夫链长度动态设置方法的分析检验和算法停止准则的验证, 综合两方面的实验结果, 也就验证与支持了时间复杂性分析的理论结果; 第 6 节是结论与讨论, 展望这类数学物理方法推广应用到以群体搜索为特征的演化算法的可扩展性和能行性.

2 模拟退火算法的弛豫模型

2.1 气体输运的弛豫方程

设有气体系统 S , 宏观上处于平衡态, 环境温度 T 恒定, 但微观上可以处于任何一个可达的微观态,

因而,系统能量可变,用 E_i 表示系统处于微观态 i 时的能量. 根据统计物理中的正则系综理论,系统处于微观态 i 的几率分布函数是下列 Gibbs 正则分布^[23]

$$p_i = \frac{1}{Z} \exp\left(\frac{-E_i}{k_B T}\right) \quad (1)$$

其中, k_B 是 Boltzmann (波尔兹曼) 常数, $Z = \sum_i \exp(-E_i/k_B T)$ 称为系综配分函数(从数学角度看,就是一个归一化常数), \sum_i 表示对所有可能的微观态求和,指数函数部分称为 Boltzmann 因子.

气体输运研究建立的经典理论是 Boltzmann 积分-微分方程,它是从分子碰撞的理论假设导出分布函数满足的积分-微分方程^[24]. 这个方程是研究气体系统非平衡涨落与输运的基础,但是,它显然过于复杂,不便于求解和分析. 此后,针对不同的具体研究对象,便出现了许多简化该方程的研究成果,其中有一个简便而且实用的简化方程是由 Bhatragar、Gross 和 Krook 三人提出的^[25]. 为了简化 Boltzmann 积分-微分方程,他们提出碰撞间隔理论,得到如下的弛豫方程,以他们的名字命名为 BGK 方程^[24,25]:

$$\frac{\partial p}{\partial t} = -\frac{p - p_0}{\tau} \quad (2)$$

这个方程的含义是 $p - p_0$ 按指数 $\exp(-t/\tau)$ 的方式衰减. 其中, p_0 表示气体系统平衡态的几率分布,而 p 是相对于平衡态的扰动态几率分布, τ 称为弛豫时间.

气体系统总是自发地向平衡态趋近,这个过程就是弛豫过程,故 τ 的含义就是,气体系统从偏离平衡态的某扰动状态自发趋于某平衡态附近所需的时间. 从微观上看,一个宏观可观测的平衡状态对应的微观态的态数最多,这些微观态统称为最可几状态. 设环境温度为 T , 气体系统的最可几状态就是能量最低态,设这个最低能量为 E_0 , 又设相应于一个能量最低态的任一扰动状态的能量为 E , 显然, $E \geq E_0$, E_0 可视为常数. 将式(1)给出的几率分布函数,代入弛豫方程(2),当环境温度不变,且气体系统处于近平衡态时,其能量 E (随而分布函数 p) 对空间的变化不敏感,即与空间变量是弱相关的^[23-25], 忽略这种弱相关性,而只关注时间变化导致的分布函数和能量的变化则有

$$\frac{dE}{dt} = \frac{k_B T}{\tau} \left[1 - \exp\left(\frac{E - E_0}{k_B T}\right) \right] \quad (3)$$

这个形式上的推导将弛豫方程进一步简化为只与扰动态能量相关的常微分方程. 基于此方程,进行

类比和一些理论推演与分析将得到模拟退火算法的弛豫时间模型,以此为基础可以获得模拟退火算法设计和改进的思路与方向,如估计弛豫时间和退火“温度”等. 再进一步,就是进行模拟退火算法的时间复杂性分析.

2.2 弛豫时间模型与退火温度

将方程(3)应用于模拟退火算法,设求解的优化问题为 $\min_{i \in S} f(i)$, 此处, S 为可行解的集合. 此时能量 E 比拟为目标函数 f 的函数值,不失一般性可假定函数 f 的最小值为零,即 $\min_{i \in S} f(i) = 0$. 设第 k 次退火时的控制参数,称为退火“温度”为 T_k (对应于方程(3)中的 $k_B T$), 此时与气体系统平衡态能量相对应的是目标函数的局部极小值,也即第 k 次退火后获得的平稳解,设为 f_k . 又设第 $k-1$ 次退火后获得的平稳解为 f_{k-1} , 从 f_{k-1} 变化到 f_k 所需的时间为 τ_k , 比拟为模拟退火算法在控制参数 T_k 时,第 k 次退火的弛豫时间,则方程(3)变为

$$\frac{df}{dt} = \frac{T_k}{\tau_k} \left[1 - \exp\left(\frac{f - f_k}{T_k}\right) \right] \quad (4)$$

方程(4)的导出是基于目标函数值 f 与气体系统能量的类比,故而它也应与空间变量弱相关. 注意这里所说的“空间变量”只是一个虚拟概念,与函数本身的自变量无关. 自变量的变化是由算法的退火(或搜索)过程引起的,而在算法中退火过程就是一个时变过程,因此,自变量的变化正好映射了时间变化引起的函数值的变化.

事实上,模拟退火算法的运行及搜索优化解的过程是按离散搜索步进行的,相应于控制参数 T_k 时的搜索步数(即内层循环次数) L_k , 按早期的随机过程分析理论,被称为马尔科夫链长度. 内层循环过程模拟的就是气体系统在温度恒定时向平衡态的弛豫过程,因此, L_k 就是弛豫时间 τ_k 的离散化表示. 于是,基于气体输运理论,特别是由碰撞间隔理论而简化的 BGK 弛豫方程(2)导出了方程(4),即为模拟退火算法的弛豫时间模型,简称弛豫模型. 它给出了分析算法时间复杂性的一个基本理论模型,也是进行算法设计与改进的理论基础. 下面以及其后的 3.1 节,将在某些合理的假设下,对方程(4)从整体(积分求解)和局部(差分求解)两方面进行具体的分析.

第 $k-1$ 次退火结束后获得的平稳解 f_{k-1} , 也是第 k 次退火开始时的起始解,对方程(4)在时间区间 $[0, \tau_k]$ 上积分有

$$f_k - f_{k-1} = T_k \left[1 - \frac{1}{\tau_k} \int_0^{\tau_k} \exp\left(\frac{f - f_k}{T_k}\right) dt \right] \quad (5)$$

根据模拟退火算法的运行机制,此处从理论上可以认为 f 是一个定义在区间 $[f_k, f_{k-1}]$ 上的随机变量,设其概率密度分布为 $\rho(f)$.

但也不可否认该算法在实际运行时会有 $f > f_{k-1}$ 的情形出现,特别是在算法运行的初期尤其如此.不过,跟踪算法运行过程表明,算法在向后期的退火过程转变时,这种情形出现的概率会越来越小,以至到了后期的退火过程 $f > f_{k-1}$ 就成了零概率事件.从另一方面理解,算法的某次“跳阱”而出现 $f > f_{k-1}$ 的情形,也就相当于由这个 f 定义了一个新的 f_{k-1} .

作了以上注解再进一步考察方程(5),运用统计物理中的各态经历假说^[23],方程(5)右端第二项的时间平均等于空间平均.即

$$\frac{1}{\tau_k} \int_0^{\tau_k} \exp\left(\frac{f-f_k}{T_k}\right) dt = \int_{f_k}^{f_{k-1}} \rho(f) \exp\left(\frac{f-f_k}{T_k}\right) df \quad (6)$$

在统计物理中,各态经历假说也称为 Gibbs 系综平均假设,是平衡态统计物理学的基础^[24],它给出了时间到空间的转换.

由 Metropolis 准则,从当前解 i (目标函数值为 $f(i)$) 产生的新解 j (假设其函数值就是 f) 被接受的概率是

$$r_k = \exp\left(\frac{f(i)-f}{T_k}\right) \quad (7)$$

模拟退火算法的大量数值实验表明,在同一个控制参数 T_k 下,每个搜索步新产生解的接受概率 r_k 相差无几.从式(7)也可以看出 r_k 的大小主要由具有较高数量级的 T_k 决定,即 r_k 对新解的函数值 f 不敏感.也就是说,在退火进程中,各个新解 f 被接受的概率几乎相等.若第 k 次退火运用 Metropolis 准则进行了充分地搜索,即 L_k 足够长,则从理论分析的角度可以近似地将 f 看成是区间 $[f_k, f_{k-1}]$ 上均匀分布的连续型随机变量^[26],即有

$$\rho(f) = \frac{1}{f_{k-1} - f_k} \quad (8)$$

式(8)代入式(6)得

$$\frac{1}{\tau_k} \int_0^{\tau_k} \exp\left(\frac{f-f_k}{T_k}\right) dt = \frac{T_k}{f_{k-1} - f_k} \left(\exp\left(\frac{f_{k-1}-f_k}{T_k}\right) - 1 \right) \quad (9)$$

将式(9)右端的指数函数作 Taylor 展开到 3 次方,并将计算结果代入式(5)整理后得

$$T_k = \frac{1}{3} (f_{k-1} - f_k) \quad (10)$$

式(10)从理论上给出了退火温度一个简单的估计式,它表明退火温度值的数量级与目标函数取值区

间差值的数量级相当.这个结果为模拟退火算法进行实际问题求解时的温度的设置提供了有益的参考,如在初始温度设定时,首先估计目标函数的最大值和最小值,于是初始温度 T_0 便可取与它们差值的数量级相当的某个值.这里有必要说明的是,为了得到估计式(10)作了一些假设和近似计算与推导,所以其中的 1/3 没有实质上的意义,在进行实际的算法设计和问题求解时,可以依据问题的特征用一个恰当的常数代替.

事实上若要精确计算,可以将式(9)右端的指数函数展开为无穷级数,前面为了演算简单明了,只对其展开到 3 次方,舍弃的是一个收敛的级数 $\sum_{n=4}^{\infty} \frac{1}{n!} x^n$,

其中 $x = \frac{f_{k-1} - f_k}{T_k}$. 由于 T_k 的数量级通常高于或相当于函数值的数量级,因此 x 是小量,上述舍弃的级数也是一个较小的数,所以估计式(10)具有一般的理论意义.

当然,在对算法进行改进和将算法应用于实际问题的求解时,可在其后加上一个关于 x 的适当函数,这在数值计算中通常称为正则化项,目的是为了数值计算的稳定性.而从改进算法的角度出发,可以依据式(10)对退火温度进行动态设置,那么被加上的这个正则化项,主要是为了弥补在理论推导中被舍弃的那个级数(本文重点讨论模拟退火算法的时间复杂性,关于动态设置退火温度将另作研究).但为了简便也不失一般性,后面的理论分析将仍用式(10).

3 马尔科夫链长度的动态设置

3.1 弛豫时间估计

上面的 2.2 节将模拟退火算法的退火过程比拟为气体系统的弛豫过程,依据方程(4)从整体上对退火过程进行分析,得到了退火温度的理论估计式.

本小节继续以方程(4)为基础,考察算法运行的内部细节,对退火过程进行局部的分析.设第 k 次退火时,两个相邻搜索步之间对应一个基本的时间单位 Δt_k ,称为离散时间步长,由 2.2 节的讨论显然有 $\Delta t_k = \tau_k / L_k$. 设从当前解 i 产生了新解 j ,若新解 j 被接受,与气体运动相比拟,可认为系统经过时间 Δt_k 从状态 i 弛豫到状态 j . 以 Δt_k (不失一般性,可设 $\Delta t_k = 1$) 为离散时间步长,用差商代替导数对方程(4)进行离散化,立即得到如下的差分方程:

$$f(j) - f(i) = \frac{T_k}{L_k} \left[1 - \exp\left(\frac{f(i) - f_k}{T_k}\right) \right] \quad (11)$$

这里需要做点说明, 由于 L_k 是 τ_k 的离散化表示, 具有时间的含义, 因此, 式(11)两边的符号必须一致. 对于真实气体, 其自发的运动使能量趋于下降, 故有 $f(j) < f(i)$, f_k 比拟为平衡态的能量, 则有 $f(i) > f_k$, 再者显然有 $T_k > 0$, 因此式(11)两边正负号一致, 不会产生矛盾. 但是, 模拟退火算法常常要接受恶化的“差解”, 以跳出局部最优的“陷阱”, 故常有 $f(j) > f(i)$, 这时式(11)两边就会出现符号矛盾. 为了化解这一矛盾可以作如下理解, 设想系统经过 Δt_k 时间从状态 i 变迁到状态 j 的过程, 在其后的某个时刻会有一个逆过程从状态 j 弛豫回到状态 i , 因此, 对于 $f(j) > f(i)$ 的情形, 就认为方程(11)是后面对应的那个逆过程的差分方程.

作了以上说明, 避免符号矛盾的简单方法就是对式(11)两边取绝对值, 于是有第 k 次退火时马尔科夫链长度(也是弛豫时间)的估计式,

$$L_k = \frac{T_k}{|f(j) - f(i)|} \left| 1 - \exp\left(\frac{f(i) - f_k}{T_k}\right) \right| \quad (12)$$

还有一种情形需要补充说明, 如果新解 j 没有被接受, 则认为在这一瞬间系统处于停滞, 状态没有改变, 也就没有差分方程(11). 鉴于此, 可以将接受新产生解的搜索步称之为有效搜索步.

式(12)只是用一次有效的搜索步作为马尔科夫链长度的估计, 显然, 由于问题的复杂性和算法运行时的随机性, 难以获得恰当而有用的估计值. 但是, 该公式表明了马尔科夫链长度应与问题相关, 而且, 为进行算法运行时的估算提供了定量计算依据和可用的启发信息. 由此对每个 T_k , 算法运行时就可以预估与之相对应的退火过程的马尔科夫链长度, 即进行动态设置, 而不必如往常那样, 取定一个固定长度. 与前面相对应, 常规取固定值的方法可称为固定设置法.

3.2 马尔科夫链长度设置

前已阐明, 基于单个有效搜索步, 式(12)不足以给出马尔科夫链长度的有效估算, 于是, 从概率论中的大数定理中, 自然会想到用多个有效搜索步估计值的平均进行估算. 这个设想很容易办到, 只要在基本模拟退火算法的每次退火之前先执行若干次 Metropolis 准则, 记录其中的有效搜索步, 并运用式(12)或其后的式(14)计算相应有效搜索步给出的马尔科夫链长度, 然后平均得到此次退火的马尔科夫链长度估算值. 这就相当于将每次退火分为前后两个阶段, 前面可称为预退火阶段, 后面则称为全退火阶段, 预退火阶段为全退火阶段预估马尔科夫链

的长度. 基于以上说明, 就得到了如下改进的模拟退火算法, 简称为动态设置模拟退火算法.

改进的策略就是, 在算法的第 k 次退火开始前加入一个预退火循环, 即在控制参数 T_k 时先执行固定次数(算法中的 M 次循环)的 Metropolis 准则, 以确定全退火阶段的马尔科夫链长度 L_k . 还有一点细节是, 因为第 k 次退火开始时 f_k 是未知的, 所以, 在算法中设置了一个记忆装置, 记录算法运行到当前所得到的目标函数的最小值 f_{\min} , 用以代替式(12)或其后的式(14)中的 f_k . 有了上述交待, 改进的算法描述如下.

算法 1. SA AwithSET(L_k).

输入: i_0, T_0, M

输出: $f(i)$

INITIALIZE(i_0, T_0);

BEGIN

$k := 0; i := i_0; f_{\min} := f(i_0);$

REPEAT

$L_k := 0;$

FOR $n := 1$ to M DO

BEGIN

GENERATE(j from i);

IF $f(j) < f(i)$ THEN

BEGIN

IF $f_{\min} > f(j)$ THEN $f_{\min} := f(j);$

CALC&.ACCUM(L_k);

$i := j;$

END;

ELSE IF $f(j) > f(i)$ &

$\exp\{[f(i) - f(j)]/T_k\} > \text{random}[0, 1)$

THEN

BEGIN

CALC&.ACCUM(L_k);

$i := j;$

END;

END;

ENDFOR;

MEAN(L_k);

FOR $n := 1$ to L_k DO

BEGIN

GENERATE(j from i);

IF $f(j) \leq f(i)$ THEN

BEGIN

IF $f_{\min} > f(j)$ THEN $f_{\min} := f(j);$

$i := j;$

END;

ELSE IF

```

exp { [f(i) - f(j)] / T_k } > random [0, 1)
THEN i := j;
END;
ENDFOR;
k := k + 1;
CALCULATE (T_k);
UNTIL stop criterion;
END;

```

算法 2 中的语句 CALC&ACCUM(L_k) 表示先用式(12)或其后的式(14)根据有效搜索步计算马尔科夫链长度 L_k 的估计值, 然后与前面已有的 L_k 作累加. MEAN(L_k) 语句表示, 预退火阶段结束后对 L_k 作总平均.

4 时间复杂性分析

模拟退火算法的运行时间主要决定于两个因素: 其一是退火温度以及它的递减方式, 其二是每次退火过程的马尔科夫链长度. 而这两个因素又相互关联, 互为表里. 但从算法的实际运行进程看, 其运行时间主要消耗在一次次的退火过程之中, 故以下小节先重点分析退火过程的时间复杂性.

4.1 退火时间分析

在 3.1 节对退火过程进行的局部分析中, 得到了第 k 次退火的马尔科夫链长度 (弛豫时间), 实际上也是退火时间的估计式. 为了推导演算简便和理论分析结果的简明, 同时也不失本质特征, 这里对估计式(12)再做化简. 使用的手段与 2.2 节一样, 为了推算简明, 首先只将式中的指数函数作 Taylor 展开到 3 次方, 并注意到式(10)得出

$$L_k = \frac{f(i) - f_k}{|f(j) - f(i)|} + \frac{3}{2} \frac{(f(i) - f_k)^2}{|f(j) - f(i)| (f_{k-1} - f_k)} + \frac{3}{2} \frac{(f(i) - f_k)^3}{|f(j) - f(i)| (f_{k-1} - f_k)^2} \quad (13)$$

回顾 2.2 节关于 f (也即这里的 $f(i)$ 和 $f(j)$) 所作的随机变量注解, 应有 $0 < f(i) - f_k \leq f_{k-1} - f_k$. 于是, 由式(13)推得如下不等式

$$L_k \leq 4 \frac{f(i) - f_k}{|f(j) - f(i)|} \quad (14)$$

同样如 2.2 节关于式(10)的说明, 这里的 4 也没有实质上的意义, 只当是一个常数 C 即可.

其次若要更精确地推算, 事实上, 式(12)中的指数函数可进一步展开为无穷级数, 此时有, 常数 $C =$

$$\sum_{n=1}^{\infty} 3^{n-1} / n!$$

不等式(14)还可以进一步简化, 就是运用式(8)来估计(14)中分母的期望值或均值. 因为已假设函数值 f 是区间 $[f_k, f_{k-1}]$ 上的连续型随机变量, 令 $y = |f(j) - f(i)|$, 则它也是连续型随机变量, 只是其取值区间向左平移 $f(i)$, 这里将 $f(i)$ 暂时看成固定值, 而 $f(j)$ 作为随机变量. 由概率论知识推得 y 的密度分布函数也是由式(8)给出^[26], 但其期望值的计算要以 $f(i)$ 为界分两段积分, 经过符号变换和积分上下限的调整有下式

$$E[y] = \frac{1}{f_{k-1} - f_k} \left(\int_0^{f(i) - f_k} f df + \int_0^{f_{k-1} - f(i)} f df \right) \quad (15)$$

此处做点注释: 如果将 $f(i)$ 当成条件, 在此先验条件下求随机变量 y 的条件期望值 $E[y | f(i)]$ ^[27], 推得的结果也如式(15).

求式(15)的积分会得到一个含有 $f(i)$ 的表达式, 于是, 再将 $f(i)$ 也看成随机变量, 有期望值 $E[f(i)] = 1/2(f_k + f_{k-1})$, 代入后得随机变量 y 的期望值 $E[y] = 1/4(f_{k-1} - f_k)$. $E[y]$ 替换不等式(14)中的分母立得

$$L_k \leq 4C \quad (16)$$

式中已用常数 C 替换了式(14)中的 4. 不等式(16)表明任何一次退火过程的马尔科夫链的长度 (或时间) L_k 均有一个固定的上界.

自从模拟退火算法提出以来, 众多学者进行了大量理论和应用研究, 基于大量测试实例和应用问题求解所作的实验分析也总结出了这样的结论. 就是第 k 次退火过程循环到一定次数后, 再搜索下去对目标函数值没有明显地改善, 这个现象说明算法搜索已局部收敛到第 k 次退火的平稳解, 从算法运行的效果或效率来看, 表明 L_k 有一个实验意义上的上界, Aarts 和 Korst^[28] 在他们的著作中做了很好的全面总结.

与早期研究中的实验分析相印证, 表明了这里的理论分析结果与实验分析得到的经验结果的一致性, 为模拟退火算法研究与应用提供了理论依据. 还有一点细节需要说明, 就是从不等式(14)到不等式(16)只对分子做了简单放大处理, 如果将分子中的 $f(i)$ 也用其期望值代入, 则不等式(16)中的系数为 2, 但这不影响结果的本质.

算法运行到后期, L_k 的上界还可以更加缩小, 这时就要用到我们前期建立的动力系统模型, 引言中已作介绍. 基于弹性力学理论建立了两个常微分方程动力系统模型, 它们分别刻画模拟退火算法前中期的局部收敛和后期的全局收敛. 而且进行了收敛速度分析, 对后期的全局收敛给出了收敛速度和

衰减因子的估计,且与动力系统模型方程中的阻尼系数密切相关.关于收敛速度的结果有,第 k 次退火时的收敛速度为 $\alpha_k = 1/2\eta_k$, η_k 是阻尼系数,而且理论分析和实验验证均表明 α_k (或 η_k) 是随退火次数 k 的增加而递增的,与收敛速度相关联的衰减因子是 $\gamma_k = e^{-\alpha_k t}$.

据此关注当前解 $f(i)$ 的衰减或收敛情况:首先,经过一步衰减到 $f(j)$ 则有近似表示

$$f(j) = e^{-\alpha_k} f(i) \quad (17)$$

如果是 $f(j) > f(i)$ 的情形,参见前面 3.1 节的注解.这里为简便起见取 $\Delta t_k = 1$,也无损一般性,因为 L_k 是 τ_k 的离散化表示.其次,设 $f(i)$ 经过 L 个有效搜索步收敛到第 k 次退火后的平稳解 f_k ,同样有 f_k 的近似表示

$$f_k = e^{-\alpha_k L} f(i) \quad (18)$$

这里显然有 $L \leq L_k$.将式(17)和式(18)代入式(14)就得到如下不等式

$$L_k \leq C \frac{e^{\alpha_k}}{e^{\alpha_k} - 1} \quad (19)$$

由于 α_k 随 k 递增,当 α_k 很大时不等式(19)右端的分式约等于 1,这表明算法后期退火过程的马尔科夫链长度只是中前期的 1/4 或 1/2,可设置为一个不太大的常数.这也说明了算法后期实际上在进行快速的局部下降搜索.

4.2 总体时间复杂性分析

前已指出,总体而言模拟退火算法的时间复杂性主要取决于两个因素,即退火温度及其下降的方式和退火过程的时间或马尔科夫链的长度.上一小节分析表明,每次退火过程的时间均有一个上限,于是整个算法的退火时间就主要由初始温度 T_0 和温度的递减速率决定.

按照通常模拟退火算法的设定方式,设温度的递减因子为 β ,它是一个接近但小于 1 的正数,以控制温度的缓慢下降,因而下降的速率就是 $1 - \beta$,即 $T_k = \beta T_{k-1}$,或

$$T_k = \beta^k T_0 \quad (20)$$

对式(20)两边取对数有, $k = \frac{\ln(T_k/T_0)}{\ln\beta}$. 运用

式(10),选取一个合适的 Δ ,它体现了终止温度与初始温度在数量级上的差别,则有算法的一个停止准则为

$$T_k = C_0 (f_{k-1} - f_k) \leq \Delta \quad (21)$$

这里用 C_0 替换了式(10)中的 $1/3$.于是算法终止于满足下列条件的 k 为

$$k > \frac{\ln(\Delta/T_0)}{\ln\beta} \quad (22)$$

如,可取 $K_{fin} = \left\lfloor \frac{\ln(\Delta/T_0)}{\ln\beta} \right\rfloor + 1$, $\left\lfloor \frac{\ln(\Delta/T_0)}{\ln\beta} \right\rfloor$ 表示取整数.这样,运用不等式(16)就能得到算法运行时马尔科夫链的总长度

$$L_{tal} = \sum_{k=1}^{K_{fin}} L_k \leq 4K_{fin}C \quad (23)$$

由前面的分析可知,它与 T_0 、 β 和 Δ 密切相关.

其实,考虑到不等式(19)及其后面的说明,对马尔科夫链总长度还可以估计得更细致一点.设定一个退火“温度”中值 T_{mid} (如取与 T_0 相差一个或几个数量级),令 $K_{mid} = \left\lfloor \frac{\ln(T_{mid}/T_0)}{\ln\beta} \right\rfloor + 1$, 则

$$L_{tal} \leq (3K_{mid} + K_{fin})C \quad (24)$$

显然, $K_{mid} < K_{fin}$.

举个简单例子说明一下,如取 $T_{mid} = 1/100T_0$, $\Delta = 1/10\,000T_0$, $\beta = 0.98$, 则有, $K_{mid} = 228$, $K_{fin} = 456$, 为总退火次数(即外层循环总次数).运用式(23), $L_{tal} \leq 1824C$, 用式(24), $L_{tal} \leq 1140C$. 由前面的理论推导可知,这里的 C 与目标函数值的大小相关.

从这个简单的举例看到,马尔科夫链的总长度不大,表明了模拟退火算法的高效性.另一方面,总体上与问题的关联性不强,也表明了模拟退火算法具有较好的普适性,对问题的复杂性不敏感.当然,在实际应用时还是要考察问题的特征,如前述初始温度 T_0 的选取,马尔科夫链长度的动态设置等,就需要与具体问题相结合.

这里除了从理论上分析获得了模拟退火算法总的时间复杂性式(23)和式(24)以外,还提出了一个算法停止准则,即式(21),以进一步改进算法,涉及三个重要的参数, T_0 、 β 和 Δ , 必定与问题相关.遗憾的是,虽然基于式(10)选取一个合适的 Δ , 式(21)似乎是一个有效的停止准则,但是由于算法运行时的随机性和问题的复杂性,往往在算法运行的前中期也会出现 $C_0(f_{k-1} - f_k) \leq \Delta$ 的情形,而且很难取到恰当的 Δ 值.因此,式(21)只具有理论上的意义,它明确了停止准则与退火温度的密切关系.

为了找到实际有效的算法停止准则,注意到式(19),它是算法运行到了后期时马尔科夫链长度的估计值,此时与退火次数 k 无关,明显小于中前期退火过程的马尔科夫链长度,几乎是一个不大的常数.于是停止准则可设置为若连续若干次退火的马尔科夫链长度都很小,而且几乎相等,则算法终止.由此,

得到算法停止时的总退火次数 K_{fin} 、终止温度 T_{fin} 和 Δ , 据此即可依式(24)理论上估计算法运行的马尔科夫链总长度, 即算法总的时间复杂性. 在下节的实验与分析中, 将结合具体的测试问题进行详细讨论.

5 实验与分析

5.1 马尔科夫链长度动态设置

本小节主要对由弛豫模型导出的马尔科夫链长度动态设置方法的有效性进行实验分析与验证, 同时为验证时间复杂性分析的理论结果提供基本的实验方法与依据, 也对上述算法停止准则进行初步的运用和检验.

5.1.1 测试问题与参数设置

选取四个典型的测试问题进行实验分析与验证. 前三个是数值函数优化问题, 其中第一个是简单的单峰函数, 即经典的 Sphere 函数, 记为 f_1 , 其最优值为 0; 第二和第三个是多峰函数, 为广泛使用的测试函数集 CEC2014^[29] 中的函数 F_4 (Shifted and Rotated Rosenbrock's Function) 及 F_9 (Shifted and Rotated Rastrigin's Function), 通过旋转和平移变换出诸多复杂的特性, 分别记为 f_2, f_3 , 其最优值分别为 400, 900. 表 1 给出了它们的表达式和自变量取值范围.

表 1 数值优化测试问题

函数	表达式	变量取值范围
f_1	$\sum_{i=1}^n x_i^2$	$[-100, 100]$
f_2	$f_4 \left(M \left(\frac{2.048(x-o_4)}{100} \right) + 1 \right) + F_4^*$	$[-100, 100]$
f_3	$f_8 \left(M \left(\frac{5.12(x-o_9)}{100} \right) \right) + F_9^*$	$[-100, 100]$

第四个是非数值优化问题, 以旅行商问题(TSP), 作为非数值优化测试问题, 它是最典型的组合优化问题, 也是著名的 NP 难题. 简单描述如下: 设有 N 个城市和距离矩阵 $\mathbf{D}=[d_{ij}]$, 其中 d_{ij} 表示城市 i 到城市 j 的距离, $i, j=1, \dots, N$, 问题是要找到遍访每个城市恰好一次的一条回路, 且其路径长度最短, 此即目标函数, 记为 f_4 (TSP).

实验的方法是采用 3.2 小节描述的马尔科夫链长度动态设置模拟退火算法(SAAwithSET(L_k))与马尔科夫链长度固定设置模拟退火算法(SAA)进行对比实验. 其相关参数设置根据具体问题不同分别设置如下: f_1 的初始退火温度为 $T_0=8000$, f_2 为

$T_0=30000$, f_3 设置为 $T_0=1000$, f_4 为 $T_0=10000$; 固定设置模拟退火算法的马尔科夫链长引用文献[28]根据大量实验和实践给出的经验设置方法, 估计值与问题的维度相关, $L_k=100n$, n 为问题维数. 动态设置模拟退火算法中, 每次退火的马尔科夫链长度由算法的预退火过程动态产生, 其预退火链长度也与问题维度关联, 设为 $2n$, 但由于式(12)或式(14)有时可能产生较大的数值舍入误差, 会造成计算失真, 沿用上述固定设置法, 此处设置最大动态链长阈值为 $L_{\max}=100n$; 所有测试问题的退火速率均设置为 $\beta=0.95$, 即 $T_{k+1}=\beta T_k$.

对于固定设置模拟退火算法, 根据经验总结的停止准则是当退火温度达到与问题目标函数最小值的数量级相当时即可停止, 下面的实验中取 $T_{\min}=0.1$. 对于动态设置模拟退火算法, 运用 4.2 小节综合式(19)和式(21)给出的停止准则, 对函数 f_1 取 $n=10, 30$ 分别进行实验, 而对于函数 f_2 和 f_3 , 由于它们比较复杂, 而模拟退火算法在求解这类复杂函数优化问题时的效率不高, 故只选取 $n=10$ 进行测试分析. 对三个数值优化问题, 算法的新解随机扰动产生方式均采用文献[30]中提出的策略, 每次随机扰动一个分量产生新解.

对于 TSP 问题, 分别取 50 个和 100 个城市进行实验, 城市的坐标在 $[0, 1000] \times [0, 1000]$ 的正方形内随机产生. 对每个测试问题均在上述参数设置下算法独立运行 30 次, 取 30 次运行结果的平均值作为求解结果.

5.1.2 实验结果与分析

首先进行的是一个基本的对比实验, 先用固定设置模拟退火算法按上述参数设置进行运行求解, 得到各个测试问题的求解结果(即目标函数的近似最优值); 再运用动态设置模拟退火算法进行求解, 不使用 4.2 小节提出的停止准则, 而以达到固定设置模拟退火算法的求解结果为停止准则; 然后, 分别统计它们的马尔科夫链总长度, 进行对比分析. 如表 2 所示, 即是两者在每个测试问题上所需的总马尔科夫链长 $\overline{L_{tot}}$ 的对比以及动态设置总链长的标准差 σ . 表中可以看出, 对于四个问题取不同的维数和不同的规模进行测试, 结果表明动态设置模拟退火算法在达到同样求解精度时, 比固定设置模拟退火算法所需的计算时间少 30% 以上且标准差比均值要小 1~2 个数量级. 这说明了提出的马尔科夫链长度动态设置方法的有效性和稳定性, 同时也为时间复杂性分析的理论结果提供了基本的实验依据.

表 2 达到同样求解精确度时固定设置与动态设置总马尔科夫链长对比

函数	固定设置 $\overline{L_{tal}}$	动态设置 $\overline{L_{tal}}$	动态设置链长标准差 σ	精确度均值
$f_1(n=10)$	2.21×10^5	8.825×10^4	3.845×10^3	$2.715E-01$
$f_1(n=30)$	6.63×10^5	1.678×10^5	5.736×10^3	$1.354E+00$
$f_2(n=10)$	2.46×10^5	1.302×10^5	1.116×10^4	$2.919E+00$
$f_3(n=10)$	1.8×10^5	1.013×10^5	6.749×10^3	$7.296E+00$
$f_4(n=50)$	1.13×10^6	5.360×10^5	2.369×10^4	$5.703E+03$
$f_4(n=100)$	2.25×10^6	1.226×10^6	2.978×10^4	$8.385E+03$

其次,对动态设置模拟退火算法运用 4.2 节给出的停止准则(那里要求的“连续若干次”设为 $2n$, n 为问题的维数,关于停止准则的实验分析将在下一小节详细讨论)进行多方面的对比实验,固定设置模拟退火算法的参数设置如上.表 3 显示的是它们求解各个测试问题时的马尔科夫链总长度 $\overline{L_{tal}}$,得到的求解结果精确度.从表中可以看出,对每个测试问题动态设置模拟退火算法的求解精确度与固定设置模拟退火算法的求解精确度基本相当.但是,动态设置模拟退火算法所需的马尔科夫链总长度 $\overline{L_{tal}}$ 比固定设置模拟

退火算法要少 30% 以上,大部分情况下要少 50% 以上,同时动态设置总链长的标准差 σ 比链长均值要小 1~2 个数量级,说明算法具有较好的稳定性.同时,从动态设置模拟退火算法求解问题精确度的标准差也能看出算法对问题求解时的稳定性.一方面,进一步验证了马尔科夫链长度动态设置方法的有效性;另一方面,也验证了本文提出的停止准则具有实用价值,也有较好的普适性.后面还将结合时间复杂性分析对停止准则进行更加细致的实验分析,下面继续从不同的角度来检验马尔科夫链长度动态设置方法.

表 3 动态与固定设置总马尔科夫链长及求解精确度对比

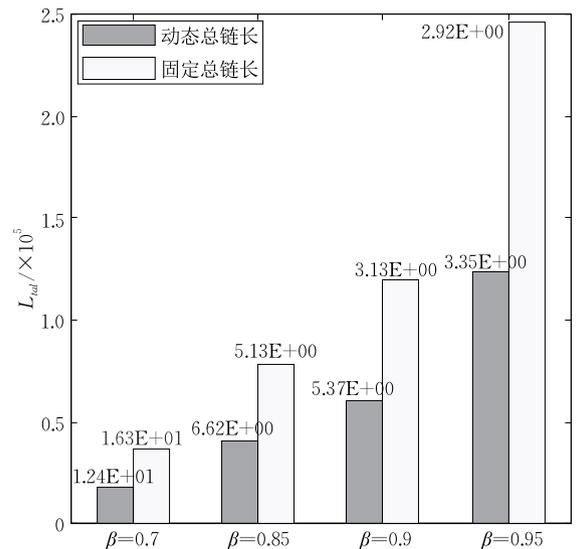
函数	固定设置 $\overline{L_{tal}}$	动态设置 $\overline{L_{tal}}$	动态设置链长标准差 σ	固定设置精确度均值	动态设置精确度均值	动态设置精确度标准差 σ
$f_1(n=10)$	2.21×10^5	8.547×10^4	3.321×10^3	$2.715E-01$	$4.287E-01$	$3.356E-01$
$f_1(n=30)$	6.63×10^5	1.132×10^5	3.812×10^3	$1.354E+00$	$3.046E+00$	$1.528E+00$
$f_2(n=10)$	2.46×10^5	1.270×10^5	1.004×10^4	$2.919E+00$	$3.351E+00$	$1.816E+00$
$f_3(n=10)$	1.8×10^5	9.124×10^4	6.362×10^3	$7.296E+00$	$1.143E+01$	$4.701E+00$
$f_4(n=50)$	1.13×10^6	5.648×10^5	3.528×10^4	$5.703E+03$	$5.667E+03$	$2.731E+02$
$f_4(n=100)$	2.25×10^6	1.03×10^6	2.536×10^4	$8.385E+03$	$8.715E+03$	$4.028E+02$

前文已述及,模拟退火算法的马尔科夫链长度与算法的初始温度以及退火率都有密切的关联,以下就是不同温度和不同退火率设置的情形下,进行马尔科夫链长度动态设置模拟退火算法的实验验证与分析.依然如前,与固定设置模拟退火算法作对比分析,此处为了对比分析的确定性,仅针对较复杂的测试函数 $f_2(n=10)$ 进行实验.

首先测试退火率变化的情形,将初始温度固定为 $T_0 = 30000$,然后实验统计在退火率分别为 $\beta = 0.7, 0.85, 0.9, 0.95$ 时,固定设置模拟退火算法以及动态设置模拟退火算法在求解测试函数 $f_2(n=10)$ 时,各自运行的总马尔科夫链长度.需要说明的是,当 $\beta < 0.7$ 时,退火速率过高,算法会出现过早收敛,故不考虑此种情况.其次测试初始温度变化的情形,将退火率固定为 $\beta = 0.95$,然后实验统计在初始温度分别设置为 $T_0 = 30, T_0 = 300, T_0 = 3000$ 和 $T_0 = 30000$ 时,固定设置模拟退火算法和动态设置模拟退火算法求解测试问题时各自运行的总马尔科

夫链长度.

图 1 显示的即是固定设置模拟退火算法以及动态设置模拟退火算法在初始温度固定退火率不同

图 1 测试函数 $f_2(n=10)$ 初始温度固定为 $T_0 = 30000$ 时,不同退火率下总马尔科夫链长对比图

时,求解测试函数运行的总马尔科夫链长度的柱状对比图。

从图中可以看出在不同退火率时,动态设置模拟退火算法的总马尔科夫链长度都小于固定设置模拟退火算法的总马尔科夫链长度,约少于 50%。

图 2 显示的是在退火率固定为 $\beta=0.95$ 时,设置不同初始温度的情况下,固定设置模拟退火算法以及动态设置模拟退火算法求解测试函数运行的总马尔科夫链长度的柱状对比图。从图中也可以得出上述相同结论。柱子上端给出的是算法按各自的停止准则终止时所得的求解结果,数值表明,相同条件下它们搜索到的最优解基本相当。

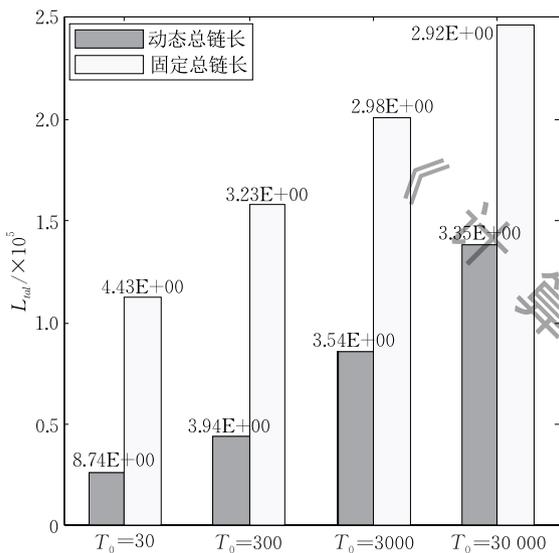


图 2 测试函数 f_2 ($n=10$) 退火率固定为 $\beta=0.95$ 时,不同初始温度下总马尔科夫链长对比图

本小节的实验从不同方面多角度地验证了基于弛豫模型导出的马尔科夫链长度动态设置方法的实用性和有效性,相较于依据经验总结给出的马尔科夫链长度固定设置方法,动态设置模拟退火算法在相同条件下进行问题求解时,能达到精度相当的求解结果,而所需的求解时间大大缩短,提升了算法的运行效率,此其一。其二,在算法运行时可以较准确地给出每次退火时有有效的马尔科夫链长度或搜步数,减少无效搜索,明确了模拟退火算法整个搜索过程的搜索性态。其三,马尔科夫链长度的动态估计为模拟退火算法时间复杂性分析提供了理论基础,而动态设置方法的实验验证为进一步进行时间复杂性的实验分析提供了实验依据和可靠的支撑。

5.2 时间复杂性分析

由弛豫模型导出的算法停止准则和理论分析得

到的总体时间复杂性结论是否适用于模拟退火算法,也必须进行实验验证。实验验证的目的其一是看实验结果与理论分析的结论是否吻合,以表明理论模型、理论分析方法和结论的适用性,其二是希望通过实验获得进一步改进和推广的思路,包括对理论模型和理论方法的改进,以及对算法的改进,理论模型与方法进一步推广的可能性。前面的结果从实验上验证了马尔科夫链长度动态设置方法的有效性,也为从实验上分析模拟退火算法的时间复杂性和实验检验停止准则的有效性提供了实验依据和可以借鉴的实验方法。

准确分析模拟退火算法的总体时间复杂性,首先需要实验验证停止准则的有效性。由前文式(21)、式(19)和式(10)可知,算法的停止准则体现的是终止温度与初始温度在数量级上的差别,从而根据具体问题给定一个适合的 Δ 值,但遗憾的是这个“适合的 Δ 值”难以做出恰当的选取。由式(19)可知,当算法运行至后期时,退火过程的马尔科夫链长度实际上已是一个不太大的常数。但是由于问题的复杂性和算法运行中的随机性,根据某次退火过程具有较短的马尔科夫链链长估计,不足以评估此次退火是否已进入算法运行的后期阶段。于是,从概率论的大数定理中自然会想到用多次连续出现小常数搜索链长的退火过程为判断依据,这就是 4.2 小节综合式(19)和式(21)提出的停止准则。这个停止准则可以在算法运行时灵活运用,实验中取连续 $2n$ (n 为问题的维数或规模)个退火过程的马尔科夫链长几乎为小常数时之前或之后的某次退火过程作为算法运行的终止时刻 K_{fin} ,由式(21)可知相应的退火温度 T_{fin} 就是 Δ 。据此,便可估算算法的总体时间复杂性。

5.2.1 停止准则实验结果与分析

图 3~图 8 分别描绘了使用停止准则的动态设置模拟退火算法(算法各参数设置与 5.1 节相同)求解各个测试问题的收敛曲线,显示了算法在各个测试问题上的停止时刻,图中纵轴表示函数值,横轴表示退火次数,即外层循环次数。图中标记的时刻值即为算法求解相应测试问题的停止时刻,也即是 t 表示的算法终止时的外层循环次数,其中 ϵ 为一个小的正整数。 ϵ 的作用是:一方面,判断是否出现了连续的小常数马尔科夫链长;另一方面也表示,这样的连续 $2n$ 次小常数链长退火过程出现后,可以选取该时刻之前或之后的某次退火作为最终停止时刻。

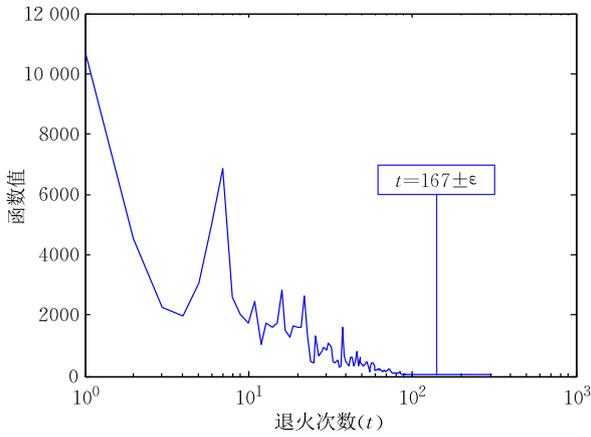


图 3 $f_1(n=10)$ 算法停止时刻

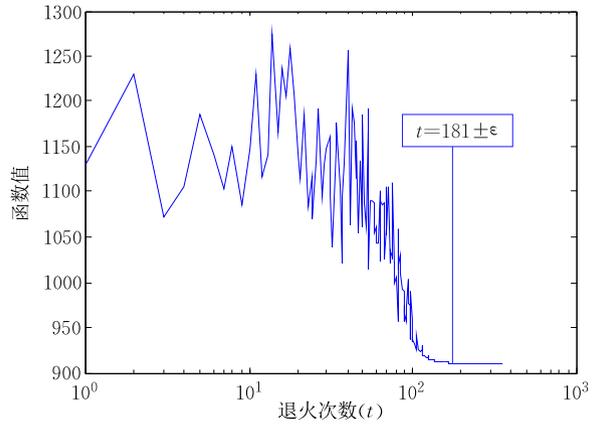


图 6 $f_3(n=10)$ 算法停止时刻

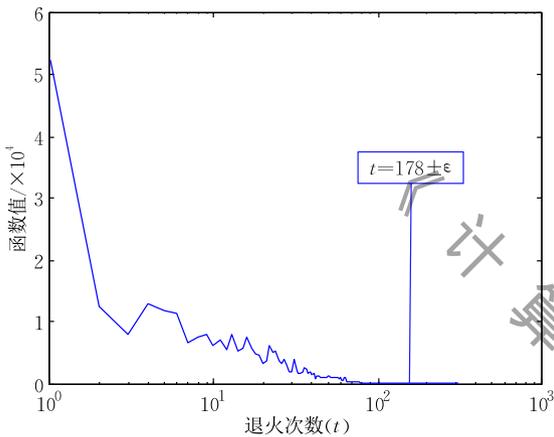


图 4 $f_1(n=30)$ 算法停止时刻

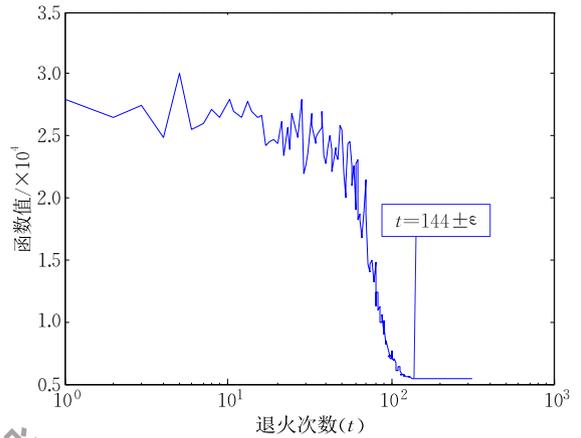


图 7 $f_4(n=50)$ 算法停止时刻

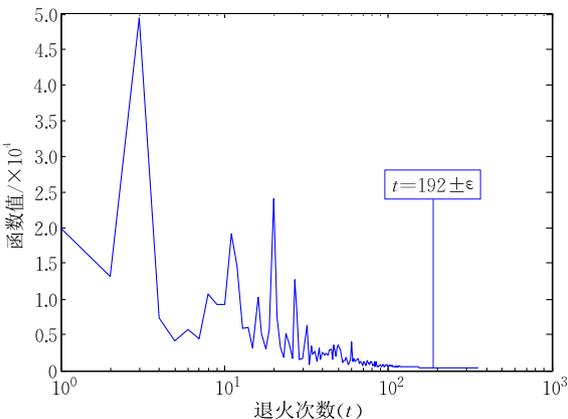


图 5 $f_2(n=10)$ 算法停止时刻

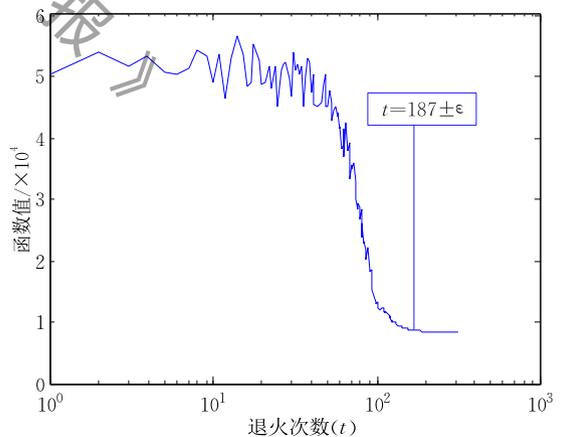


图 8 $f_4(n=100)$ 算法停止时刻

从图 3~图 8 可以看出,对于每个测试问题,依照本文提出的停止准则均能给出恰当的算法停止时刻,表明了该停止准则的有效性和较好的普适性.此外,从表 3 的“动态设置函数值”一栏中也能看出,模拟退火算法采用此停止准则,在搜索到测试函数最优值附近时即终止,表明对于中等规模和中等复杂度问题的求解,当模拟退火算法趋于全局收敛性时,停止准则能够有效检测,及时终止算法,避免过多的

无效搜索.然而,由于模拟退火算法固有的单点搜索的局限性,对于复杂和大规模问题的求解往往难以搜索到全局最优解,通常在退火温度降低到一定程度时,会陷入局部最优的陷阱之中.对于这种情况,停止准则也应能有效地检测和判定,及时终止局部搜索,避免无效搜索过多而消耗计算资源.为此,选取较复杂的测试函数 f_2 和 f_3 ,并将维数增加到 $n=50$ 进行实验分析.

图 9 和图 10(图例如上说明)分别描绘了使用停止准则的动态设置模拟退火算法(参数设置同上)求解较复杂的测试函数 $f_2(n=50)$ 和 $f_3(n=50)$ 的收敛曲线,图中标明了算法终止时刻 t . 实验结果表明,对于较复杂问题的求解,模拟退火算法陷入局部最优的陷阱,难以收敛到全局最优解时,停止准则能够有效地检测和判定,及时终止局部搜索,以利于算法跳出“局部最优的陷阱”. 一方面,避免无效搜索过多地消耗计算资源,另一方面,提供改变寻优方法和更新搜索策略的判据,如回火策略等,算法能够不失时机地变换搜索方式和方向.

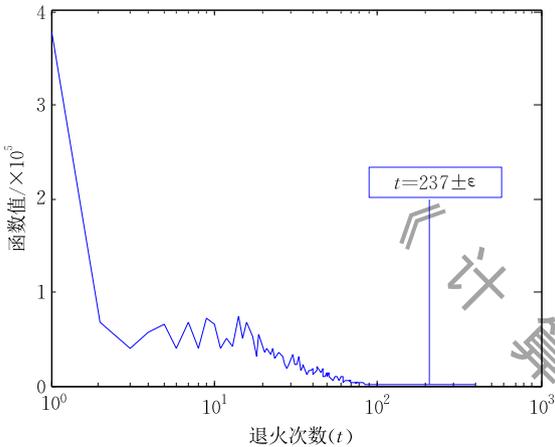


图 9 $f_2(n=50)$ 算法停止时刻

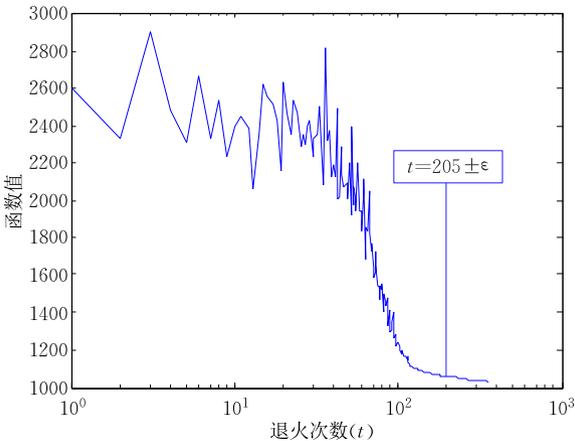


图 10 $f_3(n=50)$ 算法停止时刻

5.2.2 总体时间复杂性分析

模拟退火算法的总体时间复杂度主要取决于两个因素,即退火温度及其下降的方式和退火过程的时间或马尔科夫链的长度,但停止准则是一个必不可少的条件,它给出算法的终止时刻或总退火次数 K_{fin} . 然后,可得到终止温度 T_{fin} 或 Δ 的值,理论上由式(23)或者式(24)估计算法运行的马尔科夫链总长度,即总体时间复杂性. 在实际进行的实验中,可

以由每次退火时动态设置的马尔科夫链长度,直接统计算法运行的马尔科夫链总长度.

基于弛豫模型方程(4),从理论上得出了模拟退火算法总体时间复杂性的上界,即不等式(16). 事实上由上面的阐述,综合前面两小节关于马尔科夫链长度动态设置方法的实验验证和停止准则的实验验证,已经从实验上验证了时间复杂性分析的理论结果. 但为了量化地验证理论分析结果与实验分析结果的吻合性,本小节设计了一个更直接的实验验证方案.

选取测试函数 f_2 和 f_3 ,均在维数 $n=10$ 和 $n=30$ 进行单独测试,即将两个测试函数变为四个测试问题. 运用使用停止准则的动态设置模拟退火算法(算法各参数设置与 5.1 节相同),对每个测试问题求解过程运行 30 次,统计每次运行的马尔科夫链总长度,即为实验获得的总体时间复杂度. 然后,对比分析每个测试问题在多次求解中总体时间复杂度的变化情况.

实验结果绘于图 11 和图 12 之中,它们分别显示的是对 $f_2(n=10, 30)$ 和 $f_3(n=10, 30)$ 30 次求解

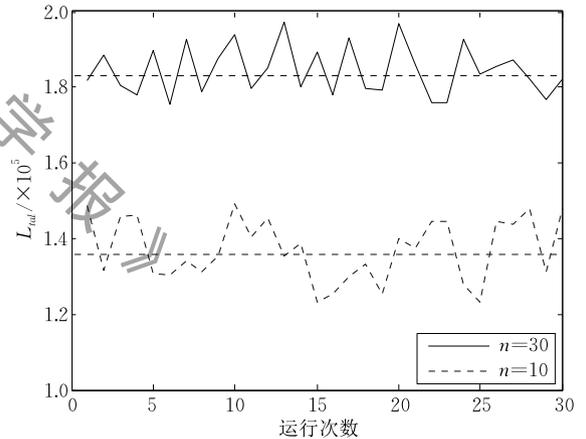


图 11 $f_2(n=10, 30)$ 总体时间复杂度波动图

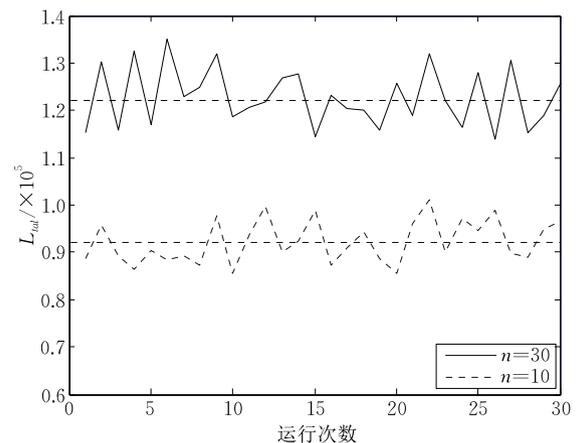


图 12 $f_3(n=10, 30)$ 总体时间复杂度波动图

运行的马尔科夫链总长度,横轴表示运行次数,纵轴表示马尔科夫链长度。

从图中可以看出,每个测试问题进行 30 次求解运行的马尔科夫链总长度不完全一致,但是相差并不大,基本上在一个平均值(虚线所示)上下波动,也即实验验证了模拟退火算法的总体时间复杂性有上界,这里的实验表明了实验分析结果与理论分析结果的一致性。

6 结论与展望

由于模拟退火算法的核心思想源于统计热力学,因此本文继续深入挖掘模拟退火算法的物理学基础,从统计物理和非均匀气体的输运理论之中汲取思想理论和分析方法。首先,基于非均匀气体的输运理论和弛豫方程建立了模拟退火算法的弛豫时间模型,作为时间复杂性分析的基础和出发点。其次,通过对弛豫模型的整体与局部分析,从理论上明确了时间复杂性与退火温度和目标函数值之间的关系,并据此提出了一个动态设置马尔科夫链长度的模拟退火算法,它也是对时间复杂性进行实验分析的基础。然后,进一步导出了时间复杂性的上界(与目标函数值相关),并与前期建立的动力系统模型相结合提出了一个有实用价值的算法停止准则。最后,从不同的方面多角度地进行实验分析和验证,实验分析表明实验结果与理论分析结果相吻合,也表明了本文建立的弛豫时间模型,运用的理论分析方法和得出的结论适用于模拟退火算法,同时也具有推广意义。

虽然在理论推导过程中做了一些假设和近似计算,但是,那些假设一方面有统计物理学基础,另一方面与模拟退火算法的运行性态相适应,具有合理性,近似计算也在可控的范围之内,因此,不失一般的理论意义。通过对测试问题的求解,实验结果也从多方面验证了这些假设的合理性、理论分析结果的正确性和实用意义。

此外,本文导出的一些理论结果还可以进一步用于指导算法的改进,如马尔科夫链长度的动态设置和退火温度的动态设置还有进行深入研究的必要。其一,理论上做更精确的推导和设计合适的正则化项,实验上做更全面的分析,以期获得最佳的改进效果;其二,与前期建立的动力系统模型相结合,导出停止准则与弹性系数和阻尼系数的关系,并对它们做出合理的修正,一方面提出更精细的停止准则,

另一方面给出更精准的回火时刻判据,当算法求解复杂问题陷入局部搜索时,能及时回火。这些为本文的后续研究提供了思路;其后,将结合动力系统模型与弛豫时间模型进行更加深入的研究,并发展这里的模型和理论应用于自然与演化计算的理论分析和算法设计之中。

诚然,模拟退火算法是一个简单的启发式拟物算法,还受固有单点搜索的局限,但它的启发式策略简单有效,物理学背景清晰,便于建立理论分析模型,挖掘其数学物理基础。然而,在演化算法研究领域,早已融入了模拟退火算法及其物理学基础,如运用热力学与统计力学的理论建立群体演化的动力学模型,并运用于算法设计,分析群体演化的动力学特征和多样性保持策略^[19-21]。本文提出的弛豫模型也是源于统计物理和气体运动论,与上述的融合思想一脉相承;再者,以群体搜索为特征的群智能算法,如演化算法等,它们的群体本身就可以比拟为气体运动系统,适合于用统计物理的模型和理论描述,因此,本文建立的弛豫模型及其建模方法能够自然地推广到演化计算领域,用于分析一类群智能算法的时间复杂性。

现在,模拟退火思想从狭义和广义上已全面融入以群体搜索为特征的群智能启发式算法之中,模拟退火的启发式策略与群智能、进化型算法的启发式策略相互融合。故而,这里的模型和理论,以及前期的动力系统模型和理论,可顺理成章地推广应用于分析一般的自然与演化计算的算法。因此,这也是本文的目的所在,希望以模拟退火算法为典型例子,深入地探究其数学物理基础,以利借鉴推广,发展自然与演化计算的算法设计与分析方法,建立更广义的数学物理模型,指导算法的设计和改进,探讨自然计算的数学物理基础和分析方法。这些将是我们后续研究工作的重点。

参 考 文 献

- [1] Dowsland K A, Thompson J M. Simulated annealing// Handbook of Natural Computing. Berlin, Germany: Springer, 2012: 1623-1655
- [2] Kirkpatrick S, Gelatt C D, Vecchi M P. Optimization by simulated annealing. Science, 1983, 220(4598): 671-680
- [3] Zhan S, Lin J, Zhang Z, et al. List-based simulated annealing algorithm for traveling salesman problem. Computational Intelligence and Neuroscience, 2016, 2016: 1712630
- [4] Fu Wen-Yuan, Ling Chao-Dong. Brownian motion based simulated annealing algorithm. Chinese Journal of Computers,

- 2014, 37(6): 1301-1308(in Chinese)
(傅文渊, 凌朝东. 布朗运动模拟退火算法. 计算机学报, 2014, 37(6): 1301-1308)
- [5] Xavier-de-Souza S, Suykens J A K, Vandewalle J, et al. Coupled simulated annealing. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 2010, 40(2): 320-335
- [6] Geng X, Chen Z, Yang W, et al. Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search. *Applied Soft Computing*, 2011, 11(4): 3680-3689
- [7] Lundy M, Mees A. Convergence of an annealing algorithm. *Mathematical Programming*, 1986, 34(1): 111-124
- [8] Granville V, Krivánek M, Rasson J P. Simulated annealing: A proof of convergence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1994, 16(6): 652-656
- [9] Jansen T, Wegener I. A comparison of simulated annealing with a simple evolutionary algorithm on pseudo-Boolean functions of unitation. *Theoretical Computer Science*, 2007, 386(1-2): 73-93
- [10] Meer K. Simulated annealing versus metropolis for a TSP instance. *Information Processing Letters*, 2007, 104(6): 216-219
- [11] Sasaki G H, Hajek B. The time complexity of maximum matching by simulated annealing. *Journal of the ACM*, 1988, 35(2): 387-403
- [12] Javidrad F, Nazari M. A new hybrid particle swarm and simulated annealing stochastic optimization method. *Applied Soft Computing*, 2017, 60: 634-654
- [13] Smith K I, Everson R M, Fieldsend J E, et al. Dominance-based multiobjective simulated annealing. *IEEE Transactions on Evolutionary Computation*, 2008, 12(3): 323-342
- [14] Mafarja M M, Mirjalili S. Hybrid whale optimization algorithm with simulated annealing for feature selection. *Neurocomputing*, 2017, 260: 302-312
- [15] Torkaman S, Ghomi S M T F, Karimi B. Hybrid simulated annealing and genetic approach for solving a multi-stage production planning with sequence-dependent setups in a closed-loop supply chain. *Applied Soft Computing*, 2018, 71: 1085-1104
- [16] He J, Yao X. Drift analysis and average time complexity of evolutionary algorithms. *Artificial Intelligence*, 2001, 127(1): 57-85
- [17] He J, Yao X. Towards an analytic framework for analysing the computation time of evolutionary algorithms. *Artificial Intelligence*, 2003, 145(1-2): 59-97
- [18] Yu Y, Qian C, Zhou Z H. Switch analysis for running time analysis of evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 2015, 19(6): 777-792
- [19] Mori N, Yoshida J, Tamaki H, et al. A thermodynamical selection rule for the genetic algorithm//*Proceedings of the IEEE International Conference on Evolutionary Computation*. Perth, Australia, 1995, 1: 188
- [20] Li Yuanxiang, Zou Xiufen. A new dynamical evolutionary algorithm from statistical mechanics. *Journal of Computer Science and Technology*, 2003, 18(3): 361-368
- [21] Cornforth T W, Lipson H. A hybrid evolutionary algorithm for the symbolic modeling of multiple-time-scale dynamical systems. *Evolutionary Intelligence*, 2015, 8(4): 149-164
- [22] Li Yuan-Xiang, Xiang Zheng-Long, Xia Jie-Ning. Dynamical system models and convergence analysis for simulated annealing algorithm. *Chinese Journal of Computers*, 2018, 42(6): 1161-1173(in Chinese)
(李元香, 项正龙, 夏界宁. 模拟退火算法的动力系统模型及收敛性分析. 计算机学报, 2018, 42(6): 1161-1173)
- [23] Chen Ren-Lie. *Introduction to Statistical Physics*. Beijing: Higher Education Press, 1978(in Chinese)
(陈仁烈. 统计物理引论. 北京: 高等教育出版社, 1978)
- [24] Huang Zu-Qia, Ding E-Jiang. *Transport Theory*. Beijing, China: Science Press, 2008(in Chinese)
(黄祖洽, 丁鄂江. 输运理论. 北京: 科学出版社, 2008)
- [25] Chapman S, Cowling T G. *The Mathematical Theory of Non-Uniform Gases*. 3rd Edition. London, UK: Cambridge University Press, 1970
- [26] Papoulis A, Pillai S U. *Probability, Random Variables and Stochastic Processes*. New York, USA: McGraw-Hill Education, 2002
- [27] Zhang Bo, Zhang Jing-Xiao. *Application of Stochastic Processes*. Beijing: Tsinghua University Press, 2004(in Chinese)
(张波, 张景肖. 应用随机过程. 北京: 清华大学出版社, 2004)
- [28] Aarts E, Korst J. *Simulated Annealing and Boltzmann Machines*. New York, USA: John Wiley and Sons Inc., 1988
- [29] Liang J J, Qu B-Y, Suganthan P N. Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. Technical Report 201311, Zhengzhou University, Zhengzhou, China and Technical Report, Nanyang Technological University, Singapore, 2013
- [30] Jin Li-Xia, Tang Huan-Wen, Li Bin. A simulated annealing algorithm for continuous functions and its convergence properties. *Mathematica Numerica Sinica*, 2005, 27(1): 19-30(in Chinese)
(靳利霞, 唐焕文, 李斌等. 一类连续函数模拟退火算法及其收敛性分析. 计算数学, 2005, 27(1): 19-30)



LI Yuan-Xiang, Ph. D. , professor, Ph. D. supervisor. His main research interests include evolutionary computation and parallels computation.

XIANG Zheng-Long, Ph. D. candidate. His main research interests include evolutionary computation and search-based software engineering.

ZHANG Wei-Yan, born in 1981, Ph. D. candidate. Her main research interest is evolutionary computation.

Background

Simulated annealing algorithm, proposed by Metropolis in 1953, has been achieved many results in algorithm design and application researches. Currently, with the decades of research and development, the theoretical researches for the algorithm on performance analysis and time complexity analysis are based primarily on the Markov chain theory in the random process, which obtains the total time complexity of specific problems. It hardly to get the interpretation of the operating mechanism and deduce instructional improved strategy. Due to mathematics and physics which have robust theories and rich methods, and modern complex dynamical systems such as dynamical systems and statistical mechanics are compatible with the evolutionary dynamics of evolutionary algorithms. These methods can be implemented to the theoretical analysis of nature inspired algorithms.

In this paper, the relaxation model, which built on the classical gas transport theory, especially the BGK relaxation equation, is proposed and established to model and analyze the simulated annealing algorithm. Thus, by comparing the

searching process of the algorithm to the non-uniform gases, the dynamical system models are established for the algorithm. Analyzing the model by using the properties of simulated algorithm, the time complexity of the algorithm and the dynamic Markov chain setting mechanism are obtained. Additionally, a novel stop criterion for stopping the algorithm process is given. The numerical experiments show that the relaxation models established in this paper are appropriate for analyzing time complexity of the simulated annealing algorithm in theory.

The work is supported by the National Natural Science Foundation of China (No. 61672391). The project is intended to study the mathematical-physical models for nature inspired computation and to design algorithms of nature inspired computation by using theory and methods of mathematics and physics. Up to now, the project team has proposed some useful models which derived meaningful results for analyzing evolutionary algorithms. And progress is partly made for this project.