

求解连续函数最大值的蚂蚁优化算法

张玉兰, 朱庆保

(南京师范大学 数学与计算机科学学院, 江苏 南京 210097)

[摘要] 求解连续函数最大值的优化算法已有多种, 但都不同程度地存在一定的局限性. 为此, 提出了一种用于求解连续函数最大值的蚂蚁优化算法——基于图的蚂蚁算法. 该方法将问题抽象为一个有向图, 模拟蚂蚁的觅食行为, 由一组蚂蚁反复地在有向图上移动, 最终得到最优解. 在阐述了该算法的具体步骤后, 从理论上对该算法的收敛性进行了分析, 证明了该算法可较快地收敛到最优解.

[关键词] 连续函数, 最大值, 基于图的蚂蚁算法, 收敛性

[中图分类号] TP311.5 [文献标识码] B [文章编号] 1672-1292(2005)03-0061-03

Solving the Maximization Problem of the Continuous Functions Based on Ant Optimization Algorithm

ZHANG Yulan ZHU Qingbao

(School of Mathematics and Computer Science, Nanjing Normal University, Jiangsu Nanjing 210097, China)

Abstract Many ways have been given to solve the maximization problem of the continuous function, however, there are some drawbacks more or less. So an ant optimization algorithm—GBAS(Graph-Based Ant System) is proposed, in which the problem is firstly abstracted as a directed graph, a group of ants then traverse on the directed graph repeatedly simulating the behavior of foraging, and the optimization solution can be obtained in the end. The procedure of the algorithm is described in detail and its convergence is analyzed theoretically, and thus this algorithm is proved to converge to the optimization solution rapidly.

Key words continuous function, maximization, GBAS, convergence

0 引言

求解函数最大值优化问题一直受到广大研究者的关注, 并提出了一些有效的方法, 大体可以分为两类: 传统方法(如梯度下降法、迭代法等)和智能方法(如神经网络法、遗传算法等). 尽管它们都能够求解函数(包括凹凸函数以及离散和连续函数)的最大值, 但不同程度地存在一定的局限性, 如: 迭代法耗时大以及效率低; 基于下降的方法容易丢失全局最优解^[1]; 模拟退火算法的精度和效率低^[2]; 遗传算法的收敛速度慢^[2]等等. 针对求解函数最大值优化问题的研究现状及其不足, 受基于图的蚂蚁系统的启发^[3,4,5], 本文提出了求解连续函数最大值的蚂蚁优化算法.

蚂蚁算法是一种广泛应用于解决 NP-hard 组

合优化问题的仿生算法, 它模拟自然界蚂蚁的觅食行为: 蚂蚁在觅食的过程中会在所经过的路径上留下一一种称为信息素的化学物质, 它能在觅食过程中感知信息素的存在, 并倾向于朝该物质强度高的方向移动. 因此, 由大量蚂蚁组成的集体觅食行为就表现为一种信息正反馈现象: 某一路径越短, 该路径上走过的蚂蚁就会越多, 留下的信息素就越多, 下一个蚂蚁选择该路径的概率就会越大. 因此, 蚁群通过这种协作觅食过程, 可找到一条最短的觅食路径.

本文提出的算法首先将被求解连续函数抽象为有向图, 然后由一组蚂蚁反复地在图上运动, 最终找到最优路径即最优解.

收稿日期: 2005-01-03

作者简介: 张玉兰(1982-), 女, 硕士研究生, 主要从事智能控制等方面的学习与研究. E-mail: kulan-njnu@sohu.com;

通讯联系人: 朱庆保(1955-), 教授, 主要从事人工智能与智能控制等方面的教学与研究. E-mail: zhuqingbao@njjnu.edu.cn

1 问题描述

设求解连续函数最大值的优化问题如 (1) 式所示:

max f(x) (1)

其中 min ≤ x ≤ max x ∈ R

定义 1 设 f(x) 是实数空间的连续函数. 对于 x ∈ R 可与一个长度为 l 的二进制对应, 于是求解 max f(x) 的问题可以转化为求解 max {F(X), X ∈ {0, 1}^l}, 其中 F(X) = E(f(e^{-1}(X))), e 为实数到 {0, 1}^l 上的编码映射 (如 e^{-1} 可取 (max - min) × X / 2^{l-1} + min), E 为实数空间的增函数.

定义 2 设函数变量所对应的二进制串 X 为 {b_l, b_{l-1}, ..., b_1}, 其中 b_i ∈ {0, 1}, i = 1, 2, ..., l 任意选择一点 (记为 V_s), 则构造一个有向图 G = (C, L), 其中 C 为节点集合 (C 的个数为 2l + 1), L 为边集, C = {C_0(V_s), C_1(b_l^0), C_2(b_l^1), C_3(b_{l-1}^0), C_4(b_{l-1}^1), ..., C_{2l-3}(b_2^0), C_{2l-2}(b_2^1), C_{2l-1}(b_1^0), C_{2l}(b_1^1)}, b_j^0, b_j^1 分别代表 0, 1, j = 1, 2, 3, ..., l, L = {(V_s, b_l^0), (V_s, b_l^1), (b_l^0, b_{l-1}^0), (b_l^0, b_{l-1}^1), (b_l^1, b_{l-1}^0), (b_l^1, b_{l-1}^1), ..., (b_2^0, b_1^0), (b_2^0, b_1^1), (b_2^1, b_1^0), (b_2^1, b_1^1)}. 图 1 是 l = 3 的示例, 其中, V_{j0}, V_{j1} (j = 1, 2, 3) 分别对应 b_j^0 和 b_j^1, 箭头表示蚂蚁运动的方向.

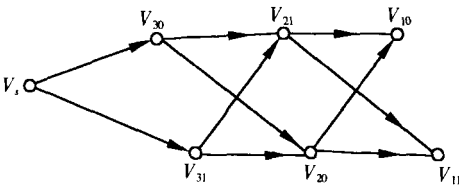


图 1 N=3 的构造图

定义 3 如果 ω_i 满足条件 (i) 到 (iii), 则 ω_i 称为有向图中的一条路径, 对应着定义 1 中的 X:

- (i) ω_i 从 C 中的起始点出发;
- (ii) ω_i 至少包含 C 中的每一个节点一次;
- (iii) ω_i 的最后一个点在 C 中没有后继点.

所有路径的集合记为 ω_s. 路径的长度为二进制的长度 l, 其中下标 i 为整数, 表示某一次循环.

2 算法及其步骤

本文的算法, 先根据第 1 节的方法构造一个有向图, 再由一组蚂蚁反复地在图上运动, 算法步骤如下:

Step 1 初始化: 将 S 只蚂蚁放在图中的起始点 V_s 处, 并设置循环次数为 M, 图中每条边上的信息素为 τ_0 = 1/N (N 为图中边的数目, 若二进制串

长度为 l 则 N 为 4l - 2);

Step 2 对于第 m 次循环, m ∈ [1, M], 每一只蚂蚁按 (2) 式给出的转移概率转移到下一个节点: ∀ v ∈ ω_m 且 (k, v) ∈ L 时:

p_{kv}(m, ω_m) = \frac{\tau_{kv}(m)}{\sum_{r \in \omega_m, (k,r) \in L} \tau_{kr}(m)} (2)

否则: p_{kv}(m, ω_m) = 0

式中, k, v ∈ C, p_{kv}(m, ω_m) 是任一只蚂蚁在第 m 次循环中 t 步之前已经经过路径 ω_m 并转移到下一节点 v 的概率, 其中 ω_m = (V_s, b_{m1}, ..., b_{m, t-1} = k), b_{mi} ∈ C, i = 1, 2, ..., t - 1, τ_{kv}(m) 表示第 m 次循环边 (k, v) 上的信息素.

Step 3 当所有的蚂蚁都完成一次循环后, 按 (3) 式进行全局信息素更新;

\tau_{kv}(m+1) = (1 - \rho)\tau_{kv}(m) + \rho\Delta\tau_{kv} (3)

式中, 当蚂蚁 A_s 已经经过了边 (k, v), 则:

\Delta\tau_{kv}^{(s)} = f(x_m) + 1 (4)

否则 \Delta\tau_{kv}^{(s)} = 0

令 D = \sum_{(k,v) \in L} \sum_{s=1}^S \Delta\tau_{kv}^{(s)}, 则有 \Delta\tau_{kv} = \frac{1}{D} \sum_{s=1}^S \Delta\tau_{kv}^{(s)} (5)

式中, ρ 为蒸发因子; f(x_m) = \max_{\omega_m \in \omega_s} \{f(e^{-1}(\omega_m))\}, x_m 表示 ω_m 所对应的变量的值, ω_s 为第 m 次循环中所有可能路径的集合; S 为蚂蚁的数目, 上标 s 表示某一只蚂蚁.

Step 4 计算 \max_{\omega_i \in \omega_s} f(e^{-1}(\omega_i)), i = 1, ..., M;

Step 5 从 1 ~ n, 计算 |\max_{\omega_i \in \omega_s} f(e^{-1}(\omega_i)) - \max_{\omega_j \in \omega_s} f(e^{-1}(\omega_j))|, 其中, i ≠ j, i, j = 1, ..., n, n 为整数, 根据具体的问题由实验来确定其取值, 一般取几到几十之间, 且 n < M.

Step 6 对于给定的常数 λ 如果

|\max_{\omega_i \in \omega_s} f(e^{-1}(\omega_i)) - \max_{\omega_j \in \omega_s} f(e^{-1}(\omega_j))| < λ 则算法结束, 退出; 否则, 循环步骤 1 ~ 5M 次, 其中 λ 由精度来确定.

3 算法收敛性分析

显然, 当算法连续运行 n 次后, 并且满足步骤 6 的终止条件: |\max_{\omega_i \in \omega_s} f(e^{-1}(\omega_i)) - \max_{\omega_j \in \omega_s} f(e^{-1}(\omega_j))| < λ 时, 算法是收敛的. 下面分析如果不满足上面的终止条件, 算法进行 M 次后的收敛性.

命题 执行基于图的蚂蚁算法的过程是一个 Markov 随机过程 [6].

引理 在第 m 次循环中至少有一只蚂蚁走过

最优路径 (即找到优化解) 的概率为 p , 且 $p \geq 1 - \prod_{i=0}^{l-1} (1 - \rho)^{(m-1)l} \prod_{i=0}^{l-1} \tau_{b_{m,i}, b_{m,i+1}}(1) J^S$, 其中 $\tau_{b_{m,i}, b_{m,i+1}}(1)$ 为第一次循环后边 $(b_{m,i}, b_{m,i+1})$ 上的信息素, 其中 $i = 0, 1, \dots, l-1$

证明 因为 $\Delta \tau_{kv} \geq 0$, $\rho > 0$ 则由式 (3) 有 $\tau_{kv}(m+1) \geq (1-\rho) \tau_{kv}(m)$ (6)

则经过迭代有 $\tau_{kv}(m) \geq (1-\rho)^{(m-1)} \tau_{kv}(1)$ (7)

因为 $\sum_{(k,v) \in \omega_m} \tau_{kv}(m) = 1^{[5]}$, 所以

$$\sum_{r \in \omega_m, (k,r) \in L} \tau_{kr}(m) \leq 1, p_{kv}(m, \omega_m) = \frac{[\tau_{kv}(m)]^\alpha}{\sum_{r \in \omega_m, (k,r) \in L} [\tau_{kr}(m)]^\alpha} \geq \tau_{kv}(m) \quad (8)$$

由式 (7) 和式 (8), 在第 m 次循环中蚂蚁 s 经过优化路径的概率 p_1^s 满足

$$p_1^s = \prod_{i=0}^{l-1} p_{b_{m,i}, b_{m,i+1}}(m, (b_{m,0}, \dots, b_{m,i})) \geq \prod_{i=0}^{l-1} \tau_{b_{m,i}, b_{m,i+1}}(m) \geq \prod_{i=0}^{l-1} (1-\rho)^{(m-1)} \tau_{b_{m,i}, b_{m,i+1}}(1) \geq (1-\rho)^{(m-1)l} \prod_{i=0}^{l-1} \tau_{b_{m,i}, b_{m,i+1}}(1)$$

其中 $b_{m,0}$ 即为 V_s , 则蚂蚁 s 不经过优化路径的概率小于等于 $1 - (1-\rho)^{(m-1)l} \prod_{i=0}^{l-1} \tau_{b_{m,i}, b_{m,i+1}}(1)$, 所有蚂蚁都没有找到优化解的概率小于等于 $[1 - (1-\rho)^{(m-1)l} \prod_{i=0}^{l-1} \tau_{b_{m,i}, b_{m,i+1}}(1)]^S$, 则至少有一只蚂蚁找到优化解的概率大于等于 $1 - [1 - (1-\rho)^{(m-1)l} \prod_{i=0}^{l-1} \tau_{b_{m,i}, b_{m,i+1}}(1)]^S$.

定理 经过 M 次循环后, 至少有一只蚂蚁找到最优解的概率趋于 1

证明 由引理的证明过程可知: 在第 m 次循环中, 所有蚂蚁都没有找到优化解的概率小于等于 $[1 - (1-\rho)^{(m-1)l} \prod_{i=0}^{l-1} \tau_{b_{m,i}, b_{m,i+1}}(1)]^S$, 则 M 次循环后, 所有蚂蚁都没有找到优化解的概率小于等于 $[1 - (1-\rho)^{(m-1)l} \prod_{i=0}^{l-1} \tau_{b_{m,i}, b_{m,i+1}}(1)]^{MS}$, 则经过 M 次循环后, 至少一只蚂蚁找到最优解的概率大于等于 $1 - [1 - (1-\rho)^{(m-1)l} \prod_{i=0}^{l-1} \tau_{b_{m,i}, b_{m,i+1}}(1)]^{MS}$.

显然, 当挥发系数 ρ 越小, 蚂蚁数目 S 越大时,

经过 M 次循环后, 至少一只蚂蚁找到最优解的概率就会越来越趋近于 1 所以通过适当的取值 (即既要考虑到精度也要兼顾效率), 可以使该算法以趋于 1 的概率收敛, 本文取 $\rho = 0.2$, $S = 50$, $l = 10$, $M = 10$ 时, 算法进行到第 5 次循环后, 即 $m = 5$ 时, 至少一只蚂蚁找到最优解的概率近似为 0.999 999 999 98

4 结语

基于图的蚂蚁算法是新兴起的仿生算法, 与遗传算法是完全不同的, 在求解不同的问题时各有特长. 本文在基本蚂蚁算法的基础上, 借鉴了遗传算法的编码和解码方法, 提出了求解连续函数优化问题的基于图的蚂蚁算法并由理论分析知, 经过 M 次循环后, 至少有一只蚂蚁找到最优解的概率趋于 1 由多只蚂蚁进行并行搜索, 提高了搜索多样性和收敛速度, 相对于遗传算法具有独特的特点. 因此, 今后可以考虑将两者结合起来, 得到混合算法, 以加快收敛速度并避免陷入局部最优点.

[参考文献]

- [1] Yang Shiyou, Ni Guangzheng, Li Yan, et al. An universal tabu search algorithm for global optimization of multimodal functions with continuous variables in electromagnetics [J]. IEEE Transactions on Magnetics, 1998, 34(5): 2901-2904
- [2] Salhi S, Queen N M. A hybrid algorithm for identifying global and local minima when optimizing functions with many minima [J]. European Journal of Operational Research, 2004, 155: 51-67
- [3] Xing Weiqing, Wei Ping. A kind of ant colony algorithm for function optimization [J]. IEEE Proceedings of the first International Conference on Machine Learning and Cybernetics, 2002(1): 4-5
- [4] Walter J. Gutjahr. A graph-based ant system and its convergence [J]. Future Generation Computer Systems, 2000 (16): 873-888
- [5] Thomas Stütz, Marco Dorigo. A short convergence proof for a class of ant colony optimization algorithms [J]. IEEE Transactions on Evolutionary Computation, 2003(6): 358-365
- [6] 周明, 孙树栋. 遗传算法原理及应用 [M]. 北京: 国防工业出版社, 1999. 27-28
- [7] 张文修, 梁怡. 遗传算法的数学基础 [M]. 西安: 西安交通大学出版社, 2000. 14-15

[责任编辑: 刘健]