

有限级信息素蚁群算法¹⁾

柯良军 冯祖仁 冯远静

(西安交通大学系统工程研究所 机械制造系统工程国家重点实验室 西安 710049)
(E-mail: kelj163@163.com)

摘要 提出一种新的蚁群算法, 将信息素分成有限个级别, 通过级别的更新实现对信息素的更新, 并且信息素的更新量独立于目标函数值. 文中采用有限马氏链的理论证明算法可以线性地收敛到全局最优解. 针对 TSP 问题, 通过与 MMAS 和 ACS 等蚁群算法的数值实验结果进行比较, 表明所提出的算法是有效的、鲁棒的.

关键词 蚁群算法, 有限马氏链, 收敛性, TSP 问题

中图分类号 TP18

Ant Colony Optimization Algorithm with Finite Grade Pheromone

KE Liang-Jun FENG Zu-Ren FENG Yuan-Jing

(Systems Engineering Institute, State Key Laboratory for Manufacturing Systems Engineering,
Xi'an Jiaotong University, Xi'an 710049)

(E-mail: kelj163@163.com)

Abstract In the paper, a new class of ant colony optimization algorithm is proposed, in which pheromone is classified into finite grades, pheromone updating is realized by changing the grades, and the updated quantity of pheromone is independent of the objective function values. It is proved by means of finite Markov chains theory that the algorithm converges to the global optimal solutions linearly. Compared with MMAS, ACS and some other ant colony optimization algorithms for the Traveling Salesman Problem, the calculating results demonstrate that the proposed algorithm is effective and robust.

Key words Ant colony optimization, finite Markov chains, convergence, Traveling Salesman Problem

1 引言

蚁群算法 (ACO) 是意大利学者 M. Dorigo^[1] 于 1996 年提出的一种随机优化算法. 近年来, 它已被成功应用于求解诸如 TSP 问题^[1,2]、Job-shop 问题^[1] 等组合优化问题. 目前蚁群算法的研究侧重于其性能优化与理论分析. ACS^[2] 和 MMAS^[3] 是两类实验效果较好的改进蚁群算法, 并且它们是收敛的^[4]. 同时, 结合模拟退火算法^[5]、免疫算法^[6]、遗传算法^[7,8] 等算法, 蚁群算法搜索能力得到很好改善.

1) 国家自然科学基金 (60475023) 资助

Supported by National Natural Science Foundation of P. R. China (60475023)

收稿日期 2004-12-30 收修改稿日期 2005-11-7

Received December 30, 2004; in revised form November 7, 2005

蚁群算法通过信息素引导整个搜索过程. 在信息素更新时利用到目标函数值, 这固然有其合理性, 但目标函数值变化的规律难以预知, 这给算法的参数设置带来很大的困难. [9] 讨论了基于超立方体框架的蚁群算法, 从而实现算法的信息素最大值不依赖于目标函数值, 但是, 它的信息素变化值与目标函数值有关. 有限级信息素蚁群算法 (Ant Colony Optimization algorithm with finite grade pheromone, 简称为 FGACO) 把信息素分成有限个级别, 并且信息素更新量与目标函数值无关, 用完全不同的方式来更新信息素. 数值实验表明 FGACO 是一类性能良好的算法.

2 组合优化问题与 MMAS

组合优化问题是运筹学的一个重要分支, 一般可以描述为:

$$\min_{s \in S} f(s)$$

其中 S 是有限集合, 它的元素由 N 个点按照一定规则排列组合而成, 记结点集为 C , N 为问题的规模, 称任意两点 $i, j \in C$ 组成的有序对 (i, j) 为弧, f 是目标函数, 把 S 中的元素映射到实数集. 典型的组合优化问题有旅行商问题 (TSP)、加工调度问题 (如 Flow-shop 问题、Job-shop 问题) 等. 蚁群算法能用来求解几乎所有的组合优化问题, 一般地, 主要针对 TSP 问题来检验算法性能.

MMAS 算法通过设定信息素的上下界来避免算法的过早收敛 (早熟), 取得很好的效果, 主要流程是:

初始化所有弧的信息素, 设定信息素的上下界等参数;

for 迭代次数 $n = 1, 2, \dots$ {

 for 蚂蚁 $ant = 1, \dots$, 最大蚂蚁数 {

 随机选取一个结点 k 作为蚂蚁的初始结点;

 已经选取的结点数 $m = 1$;

 路径表 $list(m) = k$;

 while(存在可行的后续结点){

 按照路径选择策略选择结点 l ;

$k = l$;

$m = m + 1$;

$list(m) = k$;

 }

 }

 确定当前最优路径 \hat{w} , 按照信息素更新策略更新信息素;

}

路径选择策略

假设蚂蚁在第 k 次选择时, 位于结点 i , 它按照以下式子计算选择弧 (i, j) 的概率:

$$P(c_{k+1} = j | c_k = i, \tau) = \begin{cases} \frac{\tau(c_k, j)^\alpha \cdot \eta(c_k, j)^\beta}{\sum_{\substack{r \in C \\ (c_k, r) \in J_{c_k}}} \tau(c_k, r)^\alpha \cdot \eta(c_k, r)^\beta}, & \text{如果 } (c_k, j) \in J_{c_k} \\ 0, & \text{其它情形} \end{cases} \quad (p1)$$

其中 α, β 是参数, τ 是信息素场, $\tau(i, j)$ 表示弧 (i, j) 上的信息素, J_{c_k} 是与 c_k 相连的所有弧, $P(\cdot|\cdot)$ 是条件概率. $\eta(c_k, r)$ 表征弧 (c_k, r) 上的启发信息, 对于 TSP 问题, $\eta(c_k, r)$ 通常指弧 (c_k, r) 的长度.

信息素更新策略

- 1) $\forall(i, j): \tau(i, j) \leftarrow \rho\tau(i, j)$
- 2) 如果 $f(\hat{w}) > f(w_t)$, 令 $\hat{w} = w_t$
- 3) 对于 \hat{w} , $\forall(i, j) \in \hat{w}$, $\tau(i, j) \leftarrow \tau(i, j) + 1/L(\hat{w})$
- 4) $\forall(i, j): \tau(i, j) \leftarrow \max(\tau_{\min}, \tau(i, j))$
- 5) $\forall(i, j): \tau(i, j) \leftarrow \min(\tau_{\max}, \tau(i, j))$

其中 \hat{w} 是当前最优解, w_t 是本次迭代的最优解, ρ 是挥发因子, τ_{\max} 是最大信息素, τ_{\min} 是最小信息素, $L(\hat{w})$ 是 \hat{w} 对应的目标函数值 (对于 TSP 问题, 它表示路径长度). 在上述信息素更新策略中, 只增加当前最优解对应弧上的信息素, 但 MMAS 算法也常常增加本次迭代最优解对应弧上的信息素^[3].

实验表明, MMAS 和 ACS 具有如下几个共同特点:

- 1) 参数的设置是依赖问题的, 几乎没有规律可循, 而且参数比较敏感;
- 2) 在算法的运行过程中, 大量的信息素相同或相近;
- 3) 在算法的运行过程中, 信息素是按照指数下降的, 这是算法易于早熟的一个原因;
- 4) 相对于信息素的绝对数量, 信息素之间的比例关系对于选路概率的影响更大.

3 FGACO 算法

在分析经典蚁群算法基础上, 本节提出有限级信息素蚁群算法, 它与 MMAS 算法的主要流程一致, 不同的是采用一种新的信息素更新策略.

如果两段弧上的信息素相近, 它们的概率相差非常小, 可以近似地把它们看成是等同的. 如果依据信息素把弧分成有限 M 个级别, 不同的级别按照一定映射关系对应不同实数值, 这样相同级别的选路概率相同, 就不会出现上一节 2) 中的情形. 信息素更新是通过级别的变动来实现的. 对于当前最优弧, 提高其级别, 对其它的弧, 降低其级别. 更新时只用加、减法. 记 $h(i, j)$ 是弧 (i, j) 上的级别, $g(x)$ 是单调增的正的实函数, 它实现从级别到信息素的映射关系. 这种有限级信息素更新的数学表达式如下:

- u1) $\forall(i, j): h(i, j) \leftarrow h(i, j) - r_1$
- u2) 如果 $f(\hat{w}) > f(w_t)$, 令 $\hat{w} = w_t$
- u3) 对于 \hat{w} , $\forall(i, j) \in \hat{w}$, $h(i, j) \leftarrow h(i, j) + r_2$
- u4) $\forall(i, j): h(i, j) \leftarrow \max(1, h(i, j))$
- u5) $\forall(i, j): h(i, j) \leftarrow \min(M, h(i, j))$
- u6) $\forall(i, j): \tau(i, j) = g(h(i, j))$

其中参数 r_1, r_2 是正整数, $r_1 < r_2$, 称 r_1 为惩罚数, $r_1 = 1$, 称 r_2 为奖励数; 参数 M 是最大级别数. 约定 $g(M) = \tau_{\max}$, $g(1) = 1$. 在信息素更新策略中, 利用奖励数、惩罚数来协调控制级别的变化, 从而实现对信息素的控制.

在路径选择策略 (p1) 式中使用了两个参数表征信息素和启发信息的相对重要程度, FGACO 算法把参数 α 内嵌到函数 $g(x)$ 中. 沿用上一节的符号, 假设蚂蚁在第 k 次选择

时, 位于结点 i , 于是在 FGACO 算法中, 相应的路径选择策略为:

$$P(c_{k+1} = j | c_k = i, \tau) = \begin{cases} \frac{\tau(c_k, j) \cdot \eta(c_k, j)^\beta}{\sum_{\substack{r \in C \\ (c_k, r) \in J_{c_k}}} \tau(c_k, r) \cdot \eta(c_k, r)^\beta}, & \text{如果 } (c_k, j) \in J_{c_k} \\ 0, & \text{其它情形} \end{cases} \quad (\text{p2})$$

4 算法的随机模型及其收敛性分析

表 1 第 4 节用到的记号

Table 1 Symbols used in section four

| 记号或约定 | 意义 |
|--|--|
| $\tau(n)$ | 第 n 次迭代时的信息素场, 它是 $N \times N$ 维矩阵, N 是问题的规模 |
| $\underline{w}(n)$ | 第 n 次迭代之前的最优解, ($\underline{w}(1)$ 为任意可行解), 它是 N 维向量 |
| s_n | $s_n = (\tau(n), \underline{w}(n))$, 它表示第 n 次迭代时的状态 |
| X | $X = \{s_n\}_{n=1}^\infty$ |
| Γ | s_n 所有可能取值构成的集合 |
| Z | $\underline{w}(n)$ 所有可能取值构成的集合 |
| $H: \Gamma \rightarrow Z$ | H 把状态集 Γ 中的元素映射到 Z 中, $H(\tau(n), \underline{w}(n)) = \underline{w}(n)$ |
| $G_{(i,j)}: \Gamma \rightarrow \mathfrak{R}$ | \mathfrak{R} 是实数集, $G_{(i,j)}(s_n) = \tau(n)(i, j)$, $(i, j) \in H(s_n)$, $G_{(i,j)}$ 把状态集映射到实数集, 求 $\underline{w}(n)$ 上的弧 (i, j) 对应的信息素 |
| $P_{s_n s_m}$ | 状态 s_n 转移到 s_m 的概率 |
| P | 状态转移矩阵 |
| π_0 | 初始状态的概率分布 |
| π_i | 第 i 次迭代时的概率分布 |
| π^* | 最优状态概率分布 |

下面讨论算法的随机模型, 分析算法与有限马氏链之间的关系. 有关随机过程和有限马氏链的内容见 [10]. 马氏链有如下重要性质 ([10, p. 43] 定理 3.1.1 的直接推论):

引理 1. 对于有限马氏链, 不管从何状态出发, 经过 n 步转移到任一瞬时状态的概率随着 n 无限增大而趋近于 0, 换言之, 离开所有瞬时状态的概率随着 n 无限增大而趋近于 1.

性质 1. 在 FGACO 算法中, s_n 是有限马氏链.

证明. s_n 有限状态的; 由算法的路径选择策略可知, s_n 是马氏链. \square

性质 2. 在 FGACO 算法中, 若 $s_n, s_m \in X$, 关于 s_n 到 s_m 的转移概率 $P_{s_n s_m}$ 有:

1) 若 $f(H(s_n)) < f(H(s_m))$, 则 $P_{s_n s_m} = 0$.

2) 若 $H(s_n) = H(s_m)$, 且存在某个弧 $(i, j) \in H(s_n)$, 使得 $G_{(i,j)}(s_n) > G_{(i,j)}(s_m)$, 则 $P_{s_n s_m} = 0$.

证明. 由 u1), u2), u3), 结论显然. \square

这说明状态转移时, 要么当前最优解改变, 且目标函数值下降; 要么当前最优解不变, 但它对应的每段弧的信息素递增. 如果当前最优解不变, 由 u1), u3) 式, 每段弧的信息素是收敛的 (由单调有界性可知). 经过有限次迭代后, 所有弧的信息素将保持不变, 此时当前最优解对应的弧上信息素都是 τ_{\max} , 其它弧上的信息素都是 1. 按照停留时间可以对有限马氏链的状态进行分类.

定义 1. (滞留态、最优态和正常态). 对于 $s \in \Gamma$, 如果当前最优解对应的弧上信息素都是 τ_{\max} , 其它弧上的信息素都是 1, 则称 s 是滞留态; 特别地, 当 $H(s)$ 是全局最优解时, 则称 s 为最优态; 其它状态称为正常态.

定理 1. (FGACO 的概率特性). 对于 FGACO, 具有状态 $s_n = (\underline{z}(n), \underline{w}(n))$ 的随机过程是有限马氏链, 它具有以下性质:

- 1) 所有状态不是瞬时态, 就是吸收态, 并且吸收态是最优态, 它具有常返性.
- 2) 离开正常态只需一次迭代, 离开每一个滞留态的迭代次数服从几何分布.

证明. 1) 对于最优态 s^* , 由 u1), u2), u3) 可知它只可能转移到自己, 即 $P_{s^*s^*} = 1$, 因此是吸收态, 从而具有常返性. 由性质 2 可知其它状态都是瞬时态.

2) 对于滞留态, 因为在离开该滞留态之前, 每段弧上的信息素保持不变, 从而选路概率也不变, 因此迭代次数服从几何分布. \square

定理 2. (FGACO 的收敛性). FGACO 算法收敛到全局最优解的概率随着 n 无限增大而趋近于 1.

证明. 注意到 FGACO 每次迭代时, 都保留了最优解, 由性质 2、引理 1 及定理 1 的性质 1), 可知结论成立. \square

对 Γ 中的元素按照以下约定排序: 1) 对应目标函数值升序排列; 2) 如果目标函数值相同, 最优解对应信息素降序排列, 其它弧上信息素升序排列.

性质 3. 概率转移矩阵 P 是下三角矩阵, 且 $P = \begin{bmatrix} I_k & 0 \\ A & B \end{bmatrix}$, I_k 是单位矩阵, B 是下三角矩阵, 它的最大特征值 λ 小于 1.

证明. 注意到 Γ 中的元素按照上面的约定排序, 由性质 1 和定理 1 可知结论成立. \square

由于 B 的最大特征值 λ 小于 1, 则 $(I - B)^{-1}$ 是可逆阵. 记 $M_i = I + B + B^2 + \dots + B^{i-1}$, 则 $P^i = \begin{bmatrix} I_k & 0 \\ M_i A & B^i \end{bmatrix}$, $P^\infty = \begin{bmatrix} I_k & 0 \\ (I - B)^{-1} A & 0 \end{bmatrix}$.

定理 3. (FGACO 的收敛速度). π_i 收敛到 π^* 的收敛速度满足 $\|\pi_i - \pi^*\| \leq O(\lambda^i)$, 也就是说, 算法以不大于 λ 的收敛比率线性收敛.

证明. $\|\pi_i - \pi^*\| = \|\pi_0 P^i - \pi_0 P^\infty\| = \left\| \pi_0 \left[\begin{bmatrix} I_k & 0 \\ M_i A & B^i \end{bmatrix} - \begin{bmatrix} I_k & 0 \\ (I - B)^{-1} A & 0 \end{bmatrix} \right] \right\| = \left\| \pi_0 \begin{bmatrix} 0 & 0 \\ (M_i - (I - B)^{-1}) A & B^i \end{bmatrix} \right\| \leq \|\pi_0\| \lambda^i = O(\lambda^i)$ \square

注. 显然地, 当 $M = 1$ 时, FGACO 算法退化为穷举算法, 而 τ_{\max} 充分大时, λ 充分接近于 1, 算法收敛速度最慢, 容易过早收敛, 性能最差. 合理选取适中的 τ_{\max} 是发挥算法性能的关键.

5 参数设置和数值实验分析

本节用 TSPLIB 中的 TSP 算例来测试 FGACO 算法. 沿用 TSPLIB 中的记法, 算例中的数字表示城市的数目 (比如, 算例 ft70 的城市数目是 70, kro124p 是个例外, 它的城市数目是 100). 在实现算法时, 用一张表记录已经选用的城市, 而用另一张表记录未被选用的城市.

5.1 参数设置

算法中涉及的参数有信息素的初始值、奖励数、最大级别数 M 、参数 β 和 τ_{\max} . 下面分析参数对算法的性能的影响, 实验中, 只改变一个参数, 而保持其它参数不变. 在对算

例测试时, $\beta = -5$, 信息素的初始值取为最大值的一半. 算法运行 10 次, 每次 2500 代, 蚂蚁数为 25. 函数 $g(x) = \frac{\tau_{\max} - 1}{M - 1}(x - 1) + 1$, 这是一个线性函数.

5.1.1 最大级别数 M

结果见表 2, 在 M 等于 10000 时, 平均值较大. 这时 M 大而奖励数小导致信息素变化小, 算法表现出盲目随机性. 因此, 在 M 较大时, r_2 应取较大的数. 对于算例 eil51, M 取其它四个数值时, 偏差很小; 对于算例 ry48p, M 取值为 50, 100, 1000 时, 偏差很小. 结果表明, 有限个级别能保证算法收敛到较好的结果, 并且算法具有良好的鲁棒性.

表 2 $r_2 = 3$, $\tau_{\max} = 200$, 最大级别数不同时, 平均值及偏差的百分比比较 (括号内是偏差)
Table 2 Comparison of average and percentage deviation for varying maximal grade, where $r_2 = 3$, $\tau_{\max} = 200$ (percentage deviation is indicated in bracket)

| M | 10 | 50 | 100 | 1000 | 10000 |
|-------|---------------|---------------|---------------|---------------|---------------|
| ry48p | 14571.5(1.0%) | 14534.5(0.7%) | 14545.3(0.8%) | 14544.6(0.8%) | 15758.9(9.2%) |
| eil51 | 429.2(0.8%) | 429.8(0.9%) | 428.1(0.5%) | 428.6(0.6%) | 460.5 (8%) |

5.1.2 最大信息素 τ_{\max}

在上节收敛性分析过程中, 对于 τ_{\max} 的选取有一个定性的认识, 不能太小也不能太大, 它是影响算法性能的一个重要的量. 表 3 的结果说明了这一点. 当 $N \leq \tau_{\max} \leq 6N$ 时, 两个算例的偏差都小于 1%.

表 3 $r_2 = 3$, $M = 50$, 最大信息素不同时, 平均值及偏差的百分比比较 (括号内是偏差)
Table 3 Comparison of average and percentage deviation for varying maximal pheromone, where $r_2 = 3$, $M = 50$ (percentage deviation is indicated in bracket)

| τ_{\max} | 10 | 50 | 300 | 1000 | 3000 |
|---------------|---------------|---------------|---------------|---------------|---------------|
| ry48p | 14728.4(2.1%) | 14491.0(0.5%) | 14547.1(0.9%) | 14646.9(1.6%) | 14713.4(2.0%) |
| eil51 | 431(1.2%) | 428.3(0.6%) | 428.4(0.6%) | 430.1(1.0%) | 429.3(0.8%) |

5.1.3 奖励数 r_2

在前面的分析中已经知道 M 较大时, r_2 应取较大的数, 反之, M 较小时, r_2 应取较小的数. 表 4 比较了 r_2 不同时的实验结果. r_2 较小两个算例结果较好. 但 eil51 表明 r_2 越小, 结果不一定越好.

表 4 $M = 50$, $\tau_{\max} = 50$, r_2 不同时, 平均值及偏差的百分比比较 (括号内是偏差)
Table 4 Comparison of average and percentage deviation for varying r_2 , where $M = 50$, $\tau_{\max} = 50$ (percentage deviation is indicated in bracket)

| r_2 | 2 | 5 | 10 |
|-------|---------------|---------------|---------------|
| ry48p | 14519.1(0.7%) | 14584.5(1.1%) | 14603.6(1.2%) |
| eil51 | 427.3(0.3%) | 426.9(0.2%) | 427.9(0.4%) |

5.1.4 算法收敛性能

图 1 和图 2 是最大信息数 $\tau_{\max} = 50$, 最大级别数 $M = 50$, 奖励数 $r_2 = 3$ 时的实验结果. 由图可见, 算法收敛速度快, 并且表现出很好的搜索能力, 尤其是在图 2 中, “锯齿”多, 这意味着算法能较快地搜索到更好的解.

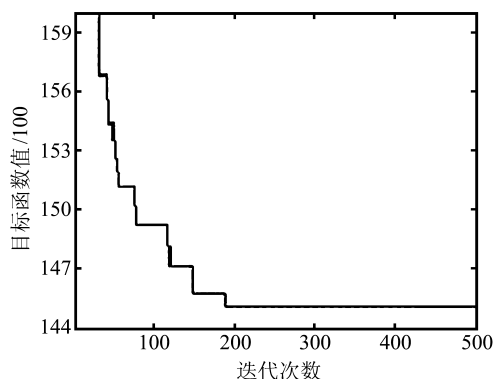


图 1 ry48p 迭代过程示意图

Fig. 1 Evolution of the best results (ry48p)

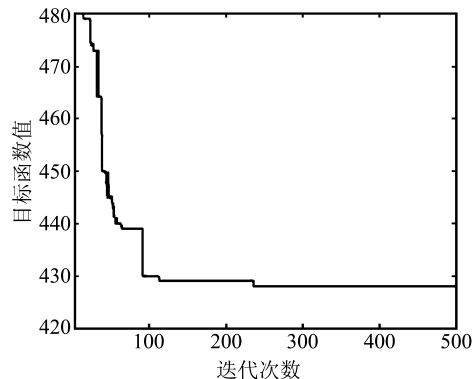


图 2 eil51 迭代过程示意图

Fig. 2 Evolution of the best results (eil51)

5.2 与其它蚁群算法的比较

依照 [11] 中提议的方式进行测试和比较, 每个算例构造 $10000kN$ 次解, 对于 TSP, $k = 1$; 对于 ATSP, $k = 2$; N 表示城市数目. 奖励数 $r_2 = 3$, 最大级别数 $M = 50$. 下面来比较 FGACO 算法与 MMAS, ACS, AS 的性能, 后三类算法的参数设置与数据引自 [3]. 前四个算例中 $g(x) = \frac{\tau_{\max} - 1}{M - 1}(x - 1) + 1$, 在 ft70 和 kro124p 算例中, $g(x) = \sqrt{\frac{\tau_{\max}^2 - 1}{M - 1}}(x - 1) + 1$, 它是一个凹函数. 由表 5 可知, ft70 算例结果与最优结果很接近, 对于其它算例, FGACO 算法比其它算法的平均结果好. 结果表明算法是有效的, 且算法不易于早熟收敛.

表 5 TSP(前三个) 和 ATSP(后三个) 的计算结果比较, 粗体字表示最好的平均结果 (括号内是 τ_{\max})
Table 5 Comparison of computational results for TSP (upper three) and ATSP (lower three), the best average results are indicated in bold face (τ_{\max} is indicated in bracket)

| 算例 | 最短路径长 | FGACO (τ_{\max}) | MMAS | MMAS+pts | ACS | AS |
|---------|-------|-------------------------|---------|----------------|---------|---------|
| eil51 | 426 | 426.8(50) | 427.6 | 427.1 | 428.1 | 437.3 |
| kroA100 | 21282 | 21286.0(400) | 21320.3 | 21291.6 | 21420.0 | 22471.4 |
| d198 | 15780 | 15950.8(800) | 15972.5 | 15956.8 | 16054.0 | 16702.1 |
| ry48p | 14422 | 14498.2(50) | 14553.2 | 14523.4 | 14565.4 | 15296.4 |
| ft70 | 38673 | 38979.9(210) | 39040.2 | 38922.7 | 39099.0 | 39596.3 |
| kro124p | 36230 | 36444.6(300) | 36773.5 | 36573.6 | 36857.0 | 38733.1 |

6 结论

信息素更新策略是蚁群算法的关键之一. 经典蚁群算法的信息素更新量依赖于目标函数值且指数下降, 导致的问题是参数的敏感性和早熟. 本文提出的有限级信息素蚁群算法采用一种新的信息素更新策略, 把信息素分成有限级别, 信息素更新由级别更新来实现, 并且更新量与目标函数值的大小无关. 这种信息素更新策略有如下的优点:

- 1) 有效性. 本文证明了算法是收敛的. 实验表明, 算法的速度较快, 并且与 MMAS、ACS 等算法相比, 不易于早熟.
- 2) 鲁棒性. 参数设置部分的实验表明, 在较大范围里, 参数对算法影响很小.

References

- 1 Dorigo M, Maniezzo V, Colomni A. Ant system: Optimization by a colony of cooperating agents. *IEEE Transaction on Systems, Man, and Cybernetics*, 1996, **26**(1): 29~41

- 2 Dorigo M, Gambardella L M. Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transaction on Evolutionary Computation*, 1997, **1**(1): 53~66
- 3 Stützle T, Hoos H H. MAX-MIN ant system. *Future Generation Computer Systems*, 2000, **16**(8): 889~914
- 4 Stützle T, Dorigo M. A short convergence proof for a class of ant colony optimization algorithms. *IEEE Transactions on Evolutionary Computation*, 2002, **6**(4): 358~365
- 5 Feng Y J, Feng Z R. Ant colony system hybridized with simulated annealing for flow-shop scheduling problems. *WSEAS Transaction on Business and Economics*, 2004, **1**(1): 133~138
- 6 Feng Y J, Feng Z R. An immunity-based ant system for continuous space function optimization. In: The Third International Conference on Machine Learning and Cybernetics (ICMLC 2004), Shanghai: IEEE Press, 2004. 1050~1054
- 7 Wu Q H, Zhang J H, Xu X H. An ant colony algorithm with mutation features. *Journal of Computer Research and Development*, 1999, **36**(10): 1240~1245
- 8 Ding J L, Chen Z Q, Yuan Z Z. On the Markov convergence analysis for the combination of genetic algorithm and ant algorithm. *Acta Automatica Sinica*, 2004, **30**(4): 629~634
- 9 Blum C, Dorigo M. The hyper-cube framework for ant colony optimization. *IEEE Transaction on Systems, Man, and Cybernetic*, 2004, **34**(2): 1161~1172
- 10 Kemeny J G, Snell J L. Finite Markov Chains. New York: Springer-Verlag, 1976, 43
- 11 Bersini H, Dorigo M, Langerman S, Seront G, Gambardella L. Results of the first international contest on evolutionary optimization. In: Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC'96), Piscataway: IEEE Press, 1996. 611~615

柯良军 西安交通大学系统工程研究所博士生. 主要研究领域为智能优化算法、模式识别.

(**KE Liang-Jun** Ph.D. candidate in Systems Engineering Institute at Xi'an Jiaotong University. His research interests include intelligent optimization algorithms and pattern recognition.)

冯祖仁 博士、西安交通大学博士生导师. 主要研究领域为机器人控制、智能系统、机器人视觉导航等.

(**FENG Zu-Ren** Ph. D., professor at Xi'an Jiaotong University. His research interests include robot control, intelligent systems, and robot navigation with computer vision.)

冯远静 西安交通大学系统工程研究所博士生. 主要研究领域为智能系统理论与应用、机器人视觉.

(**FENG Yuan-Jing** Ph.D. candidate in Systems Engineering Institute at Xi'an Jiaotong University. His research interests include intelligent system and robot vision.)