

基于蚁群算法和遗传算法融合的 QoS组播路由问题求解

孙力娟¹, 王汝传^{1, 2}

(1 南京邮电学院计算机科学与技术系, 江苏南京 210003; 21 南京大学计算机软件新技术国家重点实验室, 江苏南京 210093)

摘 要: 包含延迟、延迟抖动、带宽、丢包率和最小花费等约束条件在内的服务质量(QoS)组播路由问题, 是一个 NP 完备问题, 传统方法很难求得全局最优解. 本文将遗传算法和蚁群算法融合, 提出了用遗传蚁群算法(GAACS)求解 QoS 组播路由问题的解决方案. 仿真实验表明, 采用新算法比遗传算法具有更好的性能.

关键词: 蚁群算法; 遗传算法; QoS 组播路由

中图分类号: TP309 **文献标识码:** A **文章编号:** 0372-2112 (2006) 0821391-05

Solving QoS Multicast Routing Problem Based on the Combination of Ant Colony Algorithm and Genetic Algorithm

SUN Lijuan¹, WANG Ruochuan^{1, 2}

(1 Department of Computer Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing, Jiangsu 210003, China;
2 State Key Laboratory for Novel Software Technology at Nanjing University, Nanjing, Jiangsu 210093, China)

Abstract The least cost QoS multicast routing problem with delay, delay jitter, bandwidth, packet loss constraint belongs to NP-complete problem. It's hard to get the global solution using the traditional algorithm. In this paper, we propose a new algorithm GAACS to solve QoS multicast routing problem, whose cores are using the genetic algorithm to optimize the control parameters of the ant colony algorithm. Simulation results show that GAACS is superior to the genetic algorithm in quality and efficiency.

Key words ant colony algorithm; genetic algorithm; QoS multicast routing

1 引言

随着 Internet 网络广泛应用, 各种各样的网络服务争相出现, 先进的多媒体业务层出不穷. 一方面由于实时业务对网络传输时延、延时抖动等特性比较敏感, 当网络上有突发性业务时, 实时业务会受到很大影响; 另一方面, 多媒体业务占去了大量的带宽, 对现有的传输网络提出了挑战. 如何在网络上提供各种业务的服务质量(QoS)保证变得越来越重要. QoS 路由的目的就是在网络中寻找满足用户对线路的带宽、延迟、延迟抖动、费用要求的路由, 即向用户提供端到端的服务质量保证. 而基于多个不相关可加度量的 QoS 路由问题是 NP 完全问题. 目前一般采用启发式算法来求解, 如遗传算法, 蚁群算法等.

蚁群算法(Ant Colony Algorithm)^[1, 2]的特点是: 本身是一个增强型学习系统, 具有分布式的计算特性, 具有很强的鲁棒性. 虽然蚁群算法获得了一些成功的应用^[3~6], 但也存在一些问题. 如: 与其他算法相比, 该算法需要较长的搜索时间; 而且该方法容易出现停滞现象, 不能对解空间进行全面的搜索. 此外, 蚁群算法的参数选择缺乏理论的支持.

遗传算法(Genetic Algorithm)^[7]的特点在于: 具有全局搜索能力, 具有潜在的并行性, 具有较强的鲁棒性, 计算过程简单, 能很好地解决开发最优解和探寻搜索空间的矛盾. 但 GA 算法对于系统中的反馈信息利用不够, 当求解到一定范围时往往做大量无谓的冗余迭代, 求解精确率低.

本文将遗传算法和蚁群算法两种算法融合, 提出了一

收稿日期: 2005208212 修回日期: 2005211216

基金项目: 国家自然科学基金(No. 60573141, No. 70271050); 江苏省自然科学基金(No. BK2005146); 江苏省自然科学基金预研项目(No. BK2004218); 江苏省高技术研究计划(No. BG2006001, No. BG2005038); 南京市高技术项目(2006软资 105)江苏省计算机信息处理技术重点实验室基金(No. kjs050001, N. kjs06); 江苏省高校自然科学研究计划(No. 04KJB520095)

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. <http://www.cnki.net>

种遗传蚁群算法 (GAACS), 采用遗传算法对蚁群算法的四个控制参数 (A B Qq₀) 进行编码、优化操作, 为参数的选择提供了依据, 扩大了系统对解空间的搜索, 采用优化得到的参数组合能更好、更快地引导蚁群系统找到全局最优解。文中将其应用于求解 QoS 组播路由问题, 仿真结果显示遗传蚁群算法的求解性能要优于常规的遗传算法。

2 蚁群算法的基本原理

实验观察表明: 蚂蚁在觅食过程中会在路径上释放出一种特殊的信息素 (pheromone), 后面的蚂蚁根据遗留下来的信息素值选择下一步要走的路径。路径上的信息素值越高, 蚂蚁选择这条路径的概率就越大, 构成一个学习信息的正反馈过程。蚁群算法正是在对蚁群行为的研究基础上提出的一种启发式算法。

在蚁群算法中, 每个蚂蚁根据状态转移规则来选择下一跳节点, 而信息素值依据全局更新和局部更新两种规则来进行更新。下面介绍这三种规则^[1, 2]:

(1) 信息素全局更新规则

当 m 只蚂蚁成功地完成一次寻径行为后, 选择出目标函数值最小的路由, 来进行全局信息素更新。若 i, j 是两个相邻的节点, 则有

$$\tau(i, j) = (1 - Q) \tau(i, j) + Q \Delta \tau(i, j) \quad (1)$$

其中, Q 是信息素的消逝因子 (0 < Q < 1), L_k 是目标函数, L_{best} 是最佳目标函数值,

$$\Delta \tau(i, j) = \begin{cases} (Q / L_{\text{best}}), & \text{if } (i, j) \in \text{global_best_tour} \\ Q, & \text{otherwise} \end{cases}$$

Q 为常数系数, 用以调整 $\Delta \tau$ 的大小。

(2) 信息素局部更新规则

对于第 k 只蚂蚁, 如果节点 i, j 是所选择路径上的两个相邻节点, 则有

$$\tau(i, j) = (1 - Q) \tau(i, j) + Q \Delta \tau(i, j) \quad (2)$$

否则, 不更新。其中, 0 < Q < 1, 取 $\Delta \tau(i, s) = \tau_0$, τ_0 是各条链路上信息素的初始值。

(3) 状态转移规则:

在节点 i 的第 k 个蚂蚁选择下一节点 j 的规则是:

$$s = \begin{cases} s, & \text{otherwise} \\ \arg \max_{j \in J_k(i)} [\tau(i, j)]^A [G(i, j)]^B, & \text{if } (q \leq q_0) \end{cases} \quad (3)$$

$$p_k(i, j) = \begin{cases} \frac{[\tau(i, j)]^A [G(i, j)]^B}{\sum_{u \in J_k(i)} [\tau(i, u)]^A [G(i, u)]^B}, & \text{if } [s \in J_k(i)] \\ Q, & \text{otherwise} \end{cases} \quad (4)$$

其中, $\tau(i, j)$ 是两节点 i, j 间的信息素值。A 是残留信息的相对重要程度; B (B > 0) 是能见度的相对重要程度; q₀ 是蚂蚁状态转移方式的选择因子。J_k(i) = {0, 1, 2, ..., n-1} 表示第 k 个蚂蚁下一步可以选择的节点集; G(i, j) =

$$\frac{1}{\text{cost}(i, j)}, \text{cost}(i, j) \text{ 表示节点 } i, j \text{ 间的费用大小。}$$

蚁群算法中的四个控制参数 (A B Qq₀) 的选取与网络本身有关, 取较大的值可加快算法的收敛速度, 但是过快的收敛又会导致得到的是局部最优解, 因此不同的网络对应着不同的参数组合。通常算法中 A B 的取值在 0.05 ~ 0.2 之内, Q 应根据分泌物初始强度 τ_0 及 Q 进行取值, 保证 Q 与 Q₀ 之商和分泌物初始强度 τ_0 相差不多。本文的工作是在一定的网络拓扑结构下, 采用遗传算法求得蚁群算法的最佳参数组合, 并将其用于求解 QoS 组播路由。

3 QoS 组播路由问题的建模

QoS 路由的目的是向用户提供端到端的服务质量保证。它的质量指标包括延迟 (Delay)、延迟抖动 (Delay jitter)、带宽 (Bandwidth)、丢包率 (Packet loss) 和费用 (Cost) 等, 这些因素构成了 QoS 路由问题的约束条件。QoS 组播路由的目的就是在网络中寻找最优路径, 要求从源节点出发, 历经所有的目的节点, 并且满足所有的约束条件, 达到花费最小或达到特定的服务水平。为了方便, 在分析路由问题时可将网络看成无向带权的连通图。

网络模型表示为赋权图 G = (V, E)^[8], 其中 V 是图中所有网络节点 (交换机、路由器和主机等) 组成的集合, E 是网络双向链路的集合, 每一条边表示两节点间的直达通信路径, 假设网络是对称的, s ∈ V 为源点, M ⊆ V - {s} 为终点。对于任一链路 e ∈ E, 定义四种属性, 延迟函数 delay(e), 费用函数 cost(e), 带宽函数 bandwidth(e), 延迟抖动函数 delay_jitter(e)。对于任一网络节点 n ∈ V, 也定义四种属性, 分别为延迟函数 delay(n), 费用函数 cost(n), 延迟抖动函数 delay_jitter(n), 丢包率函数 packet_loss(n)。则对于给定的源点 s ∈ V, 终点集合 M, 节点 t ∈ M, s 和 M 组成的组播树 T(s, M) 存在下列关系:

$$(1) \text{delay}(P_T(s, t)) = \sum_{e \in P_T(s, t)} \text{delay}(e) + \sum_{n \in P_T(s, t)} \text{delay}(n)$$

$$(2) \text{cost}(T(s, M)) = \sum_{e \in T(s, M)} \text{cost}(e) + \sum_{n \in T(s, M)} \text{cost}(n)$$

$$(3) \text{bandwidth}(P_T(s, t)) = \min \{ \text{bandwidth}(e), e \in P_T(s, t) \}$$

$$(4) \text{delay_jitter}(P_T(s, t)) = \sum_{e \in P_T(s, t)} \text{delay_jitter}(e) + \sum_{n \in P_T(s, t)} \text{delay_jitter}(n)$$

$$(5) \text{packet_loss}(P_T(s, t)) = 1 - \prod_{e \in P_T(s, t)} (1 - \text{packet_loss}(e))$$

其中, P_T(s, t) 为组播树 T(s, M) 上源点 s 到终点 t 的路由路径。

以下给出 QoS 组播路由问题中约束条件的定义: 进行 QoS 路由的目的是寻找一棵组播树 T(s, M) 满足:

$$(1) \text{延迟约束: } \text{delay}(P_T(s, t)) \leq D_t$$

$$(2) \text{带宽约束: } \text{bandwidth}(P_T(s, t)) \geq B_t$$

- (3) 延迟抖动约束: $\text{delay_jitter}(P_T(s, t)) \leq DJ_i$;
- (4) 包丢失率约束: $\text{packet_loss}(P_T(s, t)) \leq PL_i$;
- (5) 费用约束: 在满足上述 4 个约束条件下, $\text{cost}(T(s, M))$ 最小。

其中, B, D_i, DJ_i, PL_i 分别代表业务对网络带宽、延迟、延迟抖动、包丢失率的约束限制。在本模型中假设所有的组播终点的带宽约束相同, 而延迟、延迟抖动和包丢失率约束可以互不相同。

4 遗传蚁群算法设计

在进行网络路由之前, 需精简处理网络拓扑结构。本文中, 除去不满足带宽约束的链路, 把网络滤成一个新的简单网络, 如果源点和终点都位于同一连通分量, 就将此连通分量作为算法研究的基图。下面的研究不再考虑带宽约束, 只考虑延迟、延迟抖动和费用度量。

QoS 限制的引入是在全局信息素更新的时候, 即用是否满足限制条件来确定惩罚因子的数值, 调整目标函数值, 从而达到调整 $f_k(t+1)$ 函数大小的目的。在 m 只蚂蚁完成一次寻径后, 计算它们的目标函数 L_k , 该函数由下式决定^[8]:

$$L_k = \frac{1}{\text{cost}[T(r, s)]} (A f_d + B f_{dj} + C f_{pl}) \quad (5)$$

其中:

$$f_d = 5_d \{ \text{delay}[T_k(r, s)] - D \};$$
$$f_{dj} = 5_{dj} \{ \text{delay_jitter}[T_k(r, s)] - DJ_i \};$$
$$f_{pl} = 5_{pl} \{ \text{packet_loss}[T_k(r, s)] - PL_i \};$$
$$5_d(Z) = \begin{cases} 1, Z \leq 0 \\ r_d, Z > 0 \end{cases} \quad 5_{dj}(Z) = \begin{cases} 1, Z \leq 0 \\ r_{dj}, Z > 0 \end{cases}$$
$$5_{pl}(Z) = \begin{cases} 1, Z \leq 0 \\ r_{pl}, Z > 0 \end{cases}$$

其中, A, B, C 分别为 f_d, f_{dj}, f_{pl} 的正加权函数, 分别表示延迟、延迟抖动和包丢失率在目标函数中所占的比重, 其值由系统根据具体应用设定。

设 $5_d(Z)$ 是延迟度量的惩罚函数, 当该路由满足延迟约束时 ($\text{delay}(P_T(s, t)) \leq D_i$), 值为 1; 否则等于 r_d ($0 < r_d < 1$)。同理, $5_{dj}(Z)$ 是延迟抖动度量和包丢失率度量的惩罚函数, 当路由满足延迟抖动约束 ($\text{delay_jitter}(P_T(s, t)) \leq DJ_i$) 其值为 1, 否则为 r_{dj} ($0 < r_{dj} < 1$); $5_{pl}(Z)$ 是包丢失率度量的惩罚函数, 当路由满足包丢失率约束时, 值为 1, 否则等于 r_{pl} ($0 < r_{pl} < 1$)。 r_d, r_{dj}, r_{pl} 的值大小决定惩罚的程度, 在本文的仿真实验中, 选择 $r_d = r_{dj} = r_{pl} = 0.5$ 。

遗传蚁群算法的基本流程:

- 1 初始化参数。

假定网络中有 m 只蚂蚁, n 个节点, 给出各个节点的 (d, dj, pl, c) 取值, 以及每条存在边的 (d, dj, b, c) 取值; 给出约束条件的 D, DJ, B, PL 的值; 随机生成 20 条 28 位编码的染色体。

° 精简网络。根据系统对网络的需求, 精简网络拓扑

结构中不符合要求的链路或节点。

» 染色体的选取。

随机地在 20 条染色体中选取 4 条进行操作。

¼ 计算适应度值。

每一条染色体对应一组 (A, B, Q, q_0) 参数组合, 进行蚁群算法的寻径。将 m 只蚂蚁放置在源节点 s 上, 并设网络拓扑中各条链路的相应信息素值 $\tau = \text{常数 } \text{const}$ 。通过状态转移规则, 信息素局部更新规则和全局更新规则, 寻找一条既满足 QoS 又整体花费最小的组播路径。将 m 只蚂蚁寻得的路径进行比较, 选取最佳的适应度值。同时, 设定一个变量 number 记录蚁群选中的所有路径中, 获得最佳路径 (即适应度值最大) 的次数。重复该步骤, 直至 4 条染色体全都得到适应度值。

½ 选择、交叉、变异操作。

通过排序, 比较每条染色体的最大适应度值和 number 值的大小, 对选中的四条染色体用 / 冒泡法 0 进行排序。若适应度值大, 表示选中的路径好, 所以此染色体较好。若适应度值相等, 表示选中的最好路径相同, 这时候就比较选中最好路径的次数, 即 number 的值, 若 number 大, 表示选中最好路径的次数多, 即命中率大, 此条染色体较好。

染色体排序后, 进行遗传算法的交叉和变异。在本算法中, 保留两条好的染色体作为父代, 以一定的几率进行交叉。对交叉后新产生的两条子代染色体, 以一定的几率对其进行变异。用两条新的子代染色体替代原来四条染色体中性能较差的两条, 进行重组, 再把重组后的四条染色体放到原来的 20 条染色体中, 完成一次循环。

¾ 重复 » ¼ ½ 直至达到循环结束条件。

通过多次循环, 就能够找到性能最好的染色体, 也就找到了蚂蚁算法的一组最佳参数。同时, 在循环的过程中, 寻找既满足 QoS 花费又最小的路径。

5 仿真实验及结果

为说明算法性能, 将遗传蚁群算法与遗传算法求解结果进行比较, 本文采用 8 网络节点的网络结构模型^[8]进行实验仿真, 如图 1 所示。

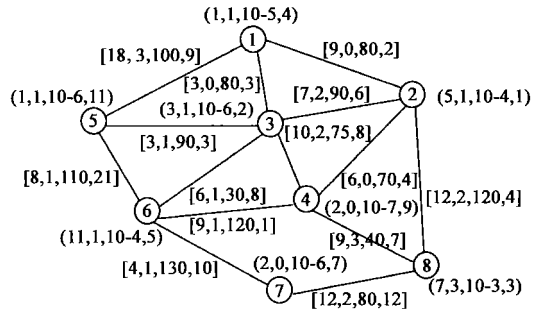


图 1 8 节点网络拓扑图

图中每个顶点用 $3d, dj, pl, c$ 4 表示, 其中的元素分别代表节点延迟、延迟抖动、丢失率和费用; 每条边用 $3d, dj, b, c$ 4

表示,其中的元素分别代表链路延迟、延迟抖动、边的带宽和费用。实验中,设染色体数目 $\text{Population}=20$ 个,每一代从 20 条染色体中随机选取 4 条,蚂蚁数 $m=20$ 同时,限定蚂蚁寻径循环次数 $\text{NC}_{\text{MAX}}=500$ 遗传迭代次数 $\text{Generation}=100$,交叉因子 $p_c=0.6$ 变异因子 $p_m=0.05$, $t=1$ 。

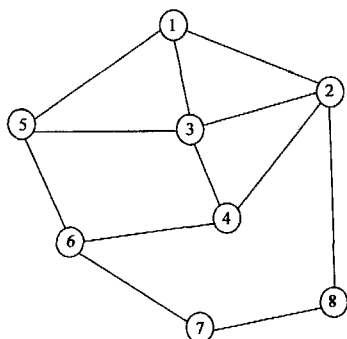


图2 经过带宽限制筛选后的网络图

在仿真实验中,取 $D=46$, $DJ=8$, $PL=0.001$, $B=70^{[8]}$ 。为了验证实验效果,对其值进行了适当调整。按照遗传蚁群算法的基本流程,先根据系统对带宽的要求 ($\text{bandwidth}(P_T(s,t)) \setminus B$) 简化网络拓扑图 1,得到图 2。由于节点 8 的丢包率为 $0.001 (= PL)$,所以节点 8 被删掉,得到图 3。下面的实验数据都由图 3 计算得到。源点为 1,目的节点集合为 $\{2, 4, 5, 7\}$ 。QoS 需求决定加权系数 A、B、C 的大小。在本实验中,设定 $A=B=C=1$,即在满足 QoS 限制下,考虑费用函数的制约作用。

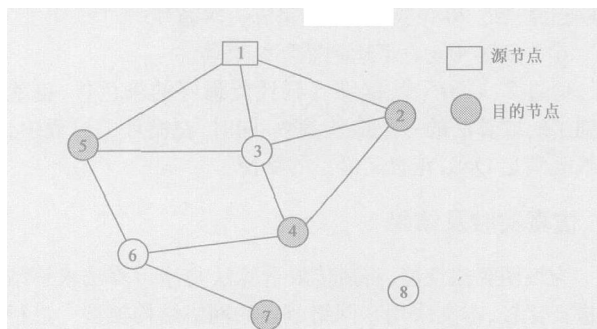


图3 经过丢包率筛选后的网络图

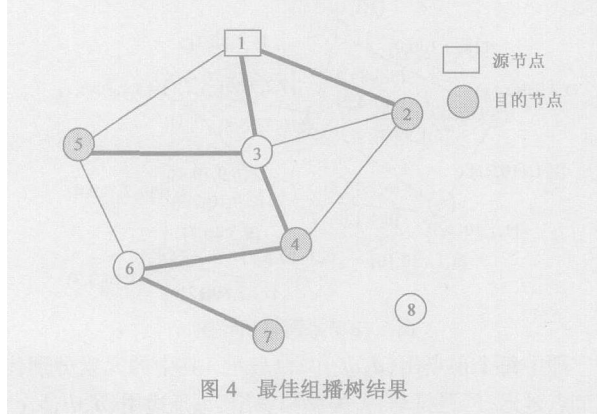


图4 最佳组播树结果

经过求解得到最佳组播树结果如图 4 所示。

文献 [8] 求解结果如图 5 所示。应用 GAACS 求解 QoS 组播树,得到结果如图 6 所示。与遗传算法相比,GAACS 的代价、延迟、延迟抖动曲线都比较平稳,几乎每一代的遗传操作均能得到最佳组播树。说明遗传蚁群算法有较好的收敛性。但在仿真实验中,遗传蚁群算法操作的时间复杂度较大,耗费的计算时间较长,影响算法的性能。

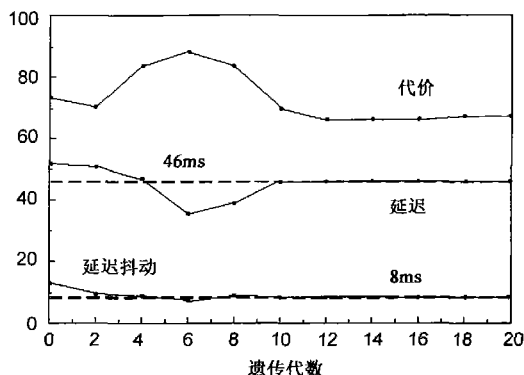


图5 遗传算法在搜索中组播树的延迟、延迟抖动、代价随代数变化曲线

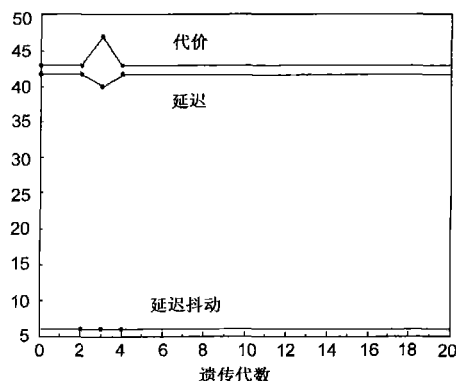


图6 遗传蚁群算法在搜索中组播树的延迟、延迟抖动、代价随代数变化曲线

用遗传蚁群算法所得的最佳参数组合,来指导基本的蚁群算法求解组播树,得到的结果如图 7 所示。其代价曲线比遗传算法的代价曲线波动要略大,说明寻径刚开始

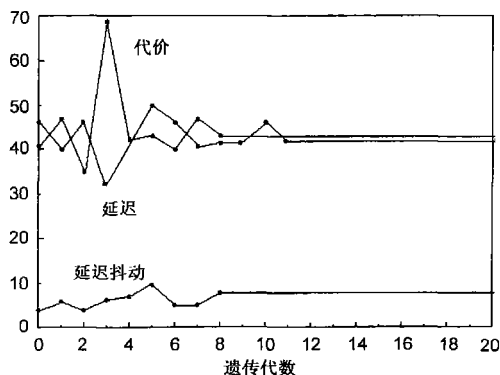


图7 由最优参数组合指导下的蚁群算法在搜索中组播树的延迟、延迟抖动、代价随代数变化曲线

时, 链路上的信息素信息比较匮乏, 蚁群进行随机地全局搜索. 但是经过几代的信息累积后, 能很快收敛于最优解. 延迟曲线相对平稳, 表示能较快地找到最优解 (或次优解). 延迟抖动曲线起伏不大, 表示在最优解附近波动.

6 结束语

本文基于蚁群算法和遗传算法的融合求解包含延迟、延迟抖动、带宽、丢包率和最小花费等约束条件在内的 QoS 组播路由问题, 实验仿真表明, 遗传蚁群算法在求解性能上优于遗传算法, 可用于求解 NP 完备问题.

参考文献:

- [1] Marco Dorigo, Vittorio Maniezzo, Alberto Colorni The ant system: Optimization by a colony of cooperating agents [J]. IEEE Transactions on Systems Man, and Cybernetics 1996, 26(1): 1- 13.
- [2] Marco Dorigo, Luca Maria Gambardella Ant colony system: A cooperative learning approach to the traveling salesman problem [J]. IEEE Transactions on Evolutionary Computation, 1997, 1(1): 53- 66
- [3] Tony White, Bernard Pagurek, Franz Oppacher ASGA: Improving the ant system by integration with genetic algo-

ritm [A]. Proceedings of the 3rd Conference on Genetic Programming (GP/SGA. 98) [C]. University of Wisconsin Madison Wisconsin USA, 1998 610- 617.

- [4] Marco Dorigo, Gianni Di Caro Ant algorithms for discrete optimization [J]. Artificial Life, 1999, 5(3): 137- 172
- [5] Thomas Stutzle, Holger H. Hoos MAX-MN ant system [J]. Future Generation Computer System, 2000, 16(8): 889 - 914
- [6] 李生红, 刘泽民, 周正. ATM 网上基于蚂蚁算法的 VC 路由选择方法 [J]. 通信学报, 2000, 21(1): 22- 28
- [7] 王小平, 曹立明. 遗传算法) 理论、应用与软件实现 [M]. 西安: 西安交通大学出版社, 2002
- [8] 王征应, 石冰心. 基于启发式遗传算法的 QoS 组播路由问题求解 [J]. 计算机学报, 2001, 24(1): 55- 61

作者简介:

孙力娟 女, 1963 年生于江苏南京, 南京邮电学院计算机科学与技术系教授, 在读博士, 研究方向是计算机网络、计算机软件在通信中应用和智能优化方法等.

王汝传 男, 1943 年出生于安徽合肥, 教授、博士生导师, 主要研究方向是计算机软件、计算机网络、信息安全、移动代理和虚拟现实技术等. Email wangrc@njupt.edu.cn