

求解度约束组播路由的新型蚁群算法

葛连升^{1,2}, 王 华¹, 王海洋¹

(1. 山东大学计算机科学与技术学院, 山东济南 250061; 2. 山东大学网络与信息中心, 山东济南 250100)

摘 要: 基于蚁群算法的正反馈机制提出了一种基于树的蚁群算法,并用它来求解度约束组播路由问题.在该算法中,蚂蚁按照一定的概率选择一条链路加入组播子树,然后检查加入点的度约束情况,如果该点的度约束情况达到饱和,则蚂蚁以后不再选取与该点连接的链路.通过计算模拟分析方法证明了该算法的有效性,计算机仿真结果显示,在解决度约束组播路由问题时,该新型蚁群算法的收敛速度大大快于已有的蚁群算法,找到的最优解性能稍好于已有的算法,算法的空间复杂度也得到降低.

关键词: 度约束; 组播路由; 蚁群启发式算法

中图分类号: TP393.01 **文献标识码:** A **文章编号:** 0372-2112 (2009) 07-1447-05

New Ant Heuristic Algorithm for Degree-Constrained Multicast Routing

GE Lian-sheng^{1,2}, WANG Hua¹, WANG Hai-yang¹

(1. School of Computer Science and Technology, Shandong University, Jinan, Shandong 250061, China;

2. Network and Information Center, Shandong University, Jinan, Shandong 250100, China)

Abstract: This paper proposes a Tree-Based Ant Colony Optimization based on the positive feedback mechanism and uses it to resolve the Degree-Constrained Multicast Routing. The ant in the new algorithm chooses some link according to the possibility distribution and adds it into the multicast sub-tree, while it checks the node where the new link is added. If the node has reached its degree constraint, the ant will never choose the adjacent links of the node. Computational simulation and analysis show validity of the new algorithm. The simulation result shows that the convergence of the new algorithm is much faster than the old one, the performance of the resolution is better, and it also has a lower space complexity when used to resolve the degree-constrained multicast routing.

Key words: degree-constrained; multicast routing; ant heuristic algorithm

1 引言

随着网络通信技术的发展,视频点播、视频会议、计算机协同计算等具有点对多点通信要求的业务开始出现,受到了广泛重视^[1,2].点到多点的通信通常使用组播技术,组播路由要寻找连接源节点和目的节点的一棵组播树^[3],信息以并行方式沿这棵树发送到不同的目的节点,信息只需在树的分枝处进行复制,这样网络中需要传送的复制信息最少,从而节省网络资源^[4].上述组播树的寻找和构造问题在数学上归结为 Steiner 树问题,该问题是 NP 完全问题^[5].经典 Steiner 树问题假定,每个节点可以向邻接的所有节点发送组播信息,但实际网络中各节点的组播能力并非如此,有些节点不具备组播能力,具备组播能力的节点也应限制其组播能力.为此,本文通过给节点指定度约束来表示节点的组播能力,并讨

论度约束情况下的组播路由问题.度约束组播路由问题被归结为有度约束的 Steiner 树问题,该问题也是一个 NP 完全问题^[6].

本文在分析研究度约束组播路由问题的基础上,借鉴相关文献的思想和方法,提出了一种新型蚁群算法求解该问题,并采用计算模拟分析方法验证了新算法的有效性.

2 相关工作

人们普遍认为,在多项式计算时间内无法获得度约束组播问题的最优解,只能采用近似方法在一个相对短的时间内获得近似最优解,在非严格意义上称这类算法为启发式算法(heuristics).文献[7]提出的 A29 算法为 SPH 算法^[8]的一种简单变形.它首先通过添加权值为无穷大的边使图连通,然后用 SPH 算法进行求解,由于算

法加入了许多不可行的边,该算法最终求出的解有可能是非可行解.文献[6]给出了求解有度约束组播问题的几个启发式算法,其中有Lagrange松弛法及一个改进Naive算法.实验结果表明改进Naive算法、Lagrange松弛法对大多数网络都能求到满足条件的解,但是Lagrange松弛法求到的解有时不满足约束条件,改进的Naive算法的性能相对较好.文献[6]还提出了一个自己的算法SPHR,即不断变换源节点的SPH算法,最后取结果最好的情况.文献[9]指出文献[6]提出的SPHR算法在很多情况下求不到解,如果其能求到解,那么解的质量好于其它算法.

随着智能优化算法研究的兴起,许多学者开始使用智能优化算法解决组播路由问题,遗传算法是应用较为广泛的一种算法^[10~15].文献[14]提出一种基于遗传算法的度约束组播路由算法,该算法使用了基于Prüfer序列的编码方案表示组播树,得到了性能优越的解.但Prüfer编码有其自身的局限性:在非完全图上,Prüfer序列交叉后得到的子代通常为不可行解,需要使用复杂的算法进行修补.文献[15]也提出一种基于遗传算法的度约束组播路由算法,该算法采用了另一种编码方法.新编码方法使用构造生成树时加入的边的序号作为编码内容.新编码方案可以保证交叉后的编码仍然表示组播树,但不能保证交叉后得到的编码满足度约束.文献[14]和[15]中组播树编码的微小变化会引起对应组播树的巨大变化,这种“不连续性”与遗传算法“基因相似染色体代表性能相近个体”的自然选择规律相矛盾,使这两种算法中的交叉操作失去意义.

蚁群算法是一种求解组合优化问题的新型启发式算法,由Marco Dorigo于1992年在博士论文中引入.研究表明该算法具有正反馈、分布式计算和富于建设性的贪婪启发式搜索等诸多优良性质^[16,17].很多研究者致力研究和改进蚁群算法,用来解组播路由问题^[18,19,21].文献[21]提出一种基于蚁群算法的度约束组播路由算法AH(Ant Heuristic),该算法使用蚂蚁寻找从源节点到目的节点的路径,组合成满足度约束的组播树,然后根据组播树性能更新信息素,通过该过程的不断迭代,最终找到问题的最优解.跟SPH、Naive、Lagrange松弛法等传统解法相比,该算法找到的解更优秀,搜索成功率也大大提高.但是,该算法的收敛速度比较慢,空间复杂度也较高,不适应实际网络情况.基于对以上问题的思考,本文提出了一种新型蚁群算法NAH(New Ant Heuristic)解度约束组播路由问题,该算法每次选择一个满足度约束的边添加到当前树中,重复该过程,直至覆盖所有目标节点,然后剪枝,得到一棵组播树.该算法加快了寻树的速度,且空间复杂度较低.仿真结果显示,新算法能够快速收敛到近似最优解,且大部分解

的性能较AH算法更优秀.

3 度约束组播问题的数学模型

通信网络用带权无向图 $G=(V, E, C)$ 表示, V 表示网络节点集合, E 是网络链路集合,表示节点间通信链路的全体,在 E 上定义的一个实值函数: $c: E \rightarrow R^+$ 称为链路费用,节点 v 的度(degree)是指与 v 关联的边的条数.假设信息由源节点 $s \in V$ 传送到一组目标节点集 $D \subseteq V - \{s\}$, (设 $D = \{dst_1, dst_2, \dots, dst_m\}, |D| = m$), $\{s\} \cup D$ 称为基本节点集;组播树 $T=(V_T, E_T)$,其中 $V_T \subseteq V, E_T \subseteq E$,且 $\{s\} \cup D \subseteq V_T, T$ 中存在由源节点 s 到每个目标节点 $d \in D$ 的通路 $P_T(s, d)$;节点 $u \in V_T$,且 $u \notin \{s\} \cup D$ 被称为Steiner点.

组播树 T 的代价为 $C(T) = \sum_{e \in E_T} c(e)$.度约束组播路由算法就是要构造一棵包含基本节点的最小费用组播树,同时每个节点满足各自的度约束,即考虑

$$\begin{aligned} \min_{e \in E_T} & c(e) \\ \text{s.t. } & d_i \leq k_i, i=1, 2, \dots, |V| \end{aligned} \quad (1)$$

其中 d_i 表示节点 v_i 在组播树中的度, k_i 表示 v_i 的度约束.

4 经典蚁群算法简介

蚁群算法是模仿真实世界的蚂蚁行为提出的.蚂蚁在没有视觉的情况下,能够找到从食物到蚁巢的最短路径;同时,它们能适应环境的改变,当原先最短路径上出现障碍物时,能够再发现一条新的最短路径.这是因为蚂蚁个体之间是通过一种称为信息素的物质进行信息传递的.蚂蚁在运动过程中,在所经路径上留下该物质,同时能够感知这种物质的存在和强度,并朝着强度高的方向移动.因此,大量蚂蚁的集体行为表现为一种信息正反馈现象:某一路径上走过的蚂蚁越多,后来的蚂蚁选择该路径的概率就越大.蚂蚁个体之间就是通过这种信息交流进行路径寻优,从而达到搜索食物的目的.

以寻找端到端最优路径为目的的蚁群算法主要分为以下三个步骤:

(1) 转移概率. 设所有蚂蚁的集合为 $\{a_1, a_2, \dots, a_m\}$. 每只蚂蚁分别根据路径的选择概率随机选取一条从起点到终点的路径,当所有蚂蚁都完成了一次路径搜索后,称算法进行了一个搜索周期.第 t 个周期内,节点 i 到节点 j 的路径选择概率 $p_{ij}^k(t)$ 定义如下:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)] [\eta_{ij}(t)]}{\sum_{s \in allowed_k} [\tau_{is}(t)] [\eta_{is}(t)]}, & J \in allowed_k \\ 0, & \text{其他} \end{cases} \quad (2)$$

式中, $allowed_k = \{0, 1, \dots, n-1\} - tabu_k$ 是第 k 只蚂蚁可以选择的下一跳节点集; $tabu_k$ 为第 k 只蚂蚁本周期已走过的节点集合; $ij(t)$ 为第 t 个周期内节点 v_i 和 v_j 之间的链路 $v_i v_j$ 上的信息素; $\eta_{ij}(t)$ 为启发函数 $\eta_{ij} = \frac{1}{c_{ij}}$, c_{ij} 为节点 v_i 和 v_j 之间的链路的费用值, 对第 k 只蚂蚁而言, c_{ij} 越小, 则 $\eta_{ij}(t)$ 越大, $p_{ij}^k(t)$ 也就越大; τ_{ij} 表示轨迹的相对重要性, 反映了蚂蚁在运动过程中积累的信息在蚂蚁运动时所起的作用, 其值越大, 该蚂蚁越倾向于选择其它蚂蚁经过的路径, 蚂蚁之间协作性越强; τ_{ij} 表示能见度的相对重要性, 反映了启发式因子在蚂蚁选择路径中的受重视程度, 其值越大, 则该转移概率越接近于贪心规则。

(2) 局部更新. 局部更新是每只蚂蚁在建立一个解的过程中进行的. 随着时间的推移, 以前留下的信息逐渐消逝, 经过 h 个时刻, 两个元素(节点) 状态之间的局部信息素数量要根据下式作调整:

$$\eta_{ij}(t+h) = (1-\rho) \cdot \eta_{ij}(t) + \rho \cdot \tau_{ij} \quad (3)$$

$$\tau_{ij} = \frac{1}{nl_{min}} \quad (4)$$

式中, $\rho \in [0, 1]$, l_{min} 表示路径集合中两个节点之间最短的距离.

(3) 全局更新. 只有生成了全局最优解的蚂蚁才能进行全局调整, 调整规则为:

$$\eta_{ij}(t+n) = (1-\rho) \cdot \eta_{ij}(t) + \rho \cdot \tau_{ij}(t) \quad (5)$$

$$\tau_{ij}(t) = \sum_{k=1}^m \eta_{ij}^k(t) \quad (6)$$

$$\eta_{ij}^k(t) = \begin{cases} \frac{Q}{L_k}, & \text{第 } k \text{ 只蚂蚁在本次循环中经过 } ij \text{ 时} \\ 0, & \text{其他} \end{cases} \quad (7)$$

式中, ρ 为挥发系数, $\rho \in [0, 1]$, $\eta_{ij}(t)$ 表示本次循环中路径 $v_i v_j$ 上的信息素数量的增量, $\eta_{ij}^k(t)$ 表示第 k 只蚂蚁在本次循环中留在路径 $v_i v_j$ 上的信息量. Q 是常数, L_k 表示第 k 只蚂蚁在本次循环中所走路程的长度, 初始时刻 $\eta_{ij}(t) = 0$, $\eta_{ij}(0) = A$, ($i, j = 0, 1, \dots, n-1$).

5 解度约束组播问题的 NAH 算法

5.1 NAH 算法构树的核心思想

AH 算法使用基本蚁群算法寻路的原理, 搜索从源节点到组成员节点的路径, 然后将路径组合成树. 与 AH 算法不同, 本文提出的 NAH 算法直接使用链路组合成树. NAH 算法首先生成一棵只包含源节点的树 T ; 然后选取与 T 连接且满足约束的链路, 组成集合 E ; 蚂蚁按照转移概率从 E 中选取一条链路 (i, j) , 其中 $i \in T, j \notin T$, 并将链路 (i, j) 加入组播子树中; 检查节点 i 的度约束情况, 如果该节点的度约束已经达到饱和, 则从 E

中删除所有与 i 相邻的链路; 重复上面过程, 直到组播子树 T 覆盖了所有的组成员节点; 对覆盖所有组成员节点的组播子树 T 进行剪枝操作, 删除所有非组成员节点的叶子节点, 得到一棵组播树 T .

5.2 NAH 算法信息素存储策略

AH 算法将信息素分成多个类别, 在每个节点使用了一个 $m \times n$ 的矩阵来记录链路信息素浓度, m 是节点邻居数目, n 是组成员节点数目, AH 算法的总空间复杂度为 $O(M \times N \times n)$, 其中 M 是拓扑链路数目, N 为拓扑节点数目. NAH 算法不是以寻找路径为基础的, 因而不需要将信息素分类. 在每个节点上使用一个长度为 n 的一维向量来记录链路信息素浓度, n 为节点邻居数目. 所以 NAH 算法的总空间复杂度为 $O(M \times N)$, 其中 M 为拓扑的链路数目, N 为拓扑的节点数目. 因此, NAH 算法的空间复杂度低于 AH 算法, 且组规模越大, 这种优势就越明显.

5.3 NAH 算法步骤

5.3.1 主算法

Step1 蚂蚁数目为常数 N_{ant} ; 当前循环次数用变量 $count$ 来记录, 初始时 $count = 0$; 最大循环次数为常数 N_{max} .

Step2 $count = count + 1$.

Step3 放出总数量为 N_{ant} 的蚂蚁, 每只蚂蚁分别按照寻树算法来寻找一棵组播树.

Step4 按照第 4 节的信息素更新公式更新链路上的信息素, 并记录当前的全局最优解. 更新时, 公式(7) 由公式(8) 代替.

$$\eta_{ij}^k = \begin{cases} \frac{Q}{C_k}, & \text{若第 } k \text{ 只蚂蚁找到的树包含链路 } v_i v_j \\ 0, & \text{否则} \end{cases} \quad (8)$$

式中, η_{ij}^k 为第 k 只蚂蚁在链路 $v_i v_j$ 上留下的信息素的量, Q 为一个常数, C_k 为第 k 只蚂蚁找到的树的总费用.

Step5 若连续若干次找到的树不再发生大的变化, 即算法已经收敛, 或者循环次数已经超过最大循环次数, 即 $count > N_{max}$, 则输出最优解, 算法结束; 否则转第 2 步.

5.3.2 寻树算法

Step1 已知源节点为 s ; 组成员节点集合为 $\{dst_1, dst_2, \dots, dst_n\}$; 网络中每个节点 v_i 的度约束上限为 k_i , 它在组播树中度用变量 d_i 来记录, 初始时 $d_i = 1$.

Step2 建立链路集合 E , 树 T , 初始时 $E = \{sv_i | s \text{ 为源节点}, v_i \in V, sv_i \in E\}$, $T = \{s\}$.

Step3 按照概率公式(9) 从集合 E 中选择一条链路 $v_m v_n$, 其中 $v_m \in T, v_n \notin T$:

$$P_i = \begin{cases} \frac{[\tau_{ij}][\eta_{ij}]}{\sum_v [\tau_{vj}][\eta_{vj}]}, & e_j \in E \\ 0, & \text{others} \end{cases} \quad (9)$$

式中, τ_{ij} 为链路 e_{ij} 上的信息素浓度, η_{ij} 为 e_{ij} 上的启发式函数, 本文取 $\eta_{ij} = \frac{1}{c_{ij}}$, c_{ij} 为 e_{ij} 的费用, α 和 β 是调节因子.

Step4 将链路 $v_m v_n$ 添加到树 T 中, 同时 $d_m = d_m + 1$; 若 $d_m > k_m$, 则令 $E = E - \{v_i v_j | v_i v_j \in E, i = m \text{ 或者 } j = n\}$, 否则, $E = E - \{v_i v_j | v_i v_j \in E, j = n\}$.

Step5 若 T 尚未覆盖所有目标节点, 则跳转至 Step3; 否则, 对 T 进行剪枝, 去除所有非组成员节点的叶子节点, 得到一棵组播树, 算法结束.

6 仿真结果和数值分析

评价算法优劣可以采用三种不同的方法: 第一, 最坏情形分析, 即通过最坏实例来评价算法的复杂性和解的偏离程度; 第二, 概率分析, 即假设实例的数据服从不同的概率分布, 研究算法的效率和效果; 第三, 计算模拟分析, 即通过大量的数据、实例进行模拟和测试, 分析算法的平均复杂性和解的偏离程度, 也可以分析解的稳定性. 前两种分析方法都是理论分析方法, 要求有很强的数学基础和推演能力. 目前, 学者们只对一些特殊和简单的问题采用理论分析方法, 对大量的组合最优化问题都是采用大规模计算模拟分析方法^[20]. 本文采用大规模计算模拟分析方法来验证所提算法的有效性.

本文采用文献[21]中的网络拓扑构造方法和度约束设定方法, 使用 VC++ 6.0 程序实现新算法. 仿真实验得到了下面的实验数据. 其中, AH 代表文献[21]提出的算法, NAH 代表本文提出的算法.

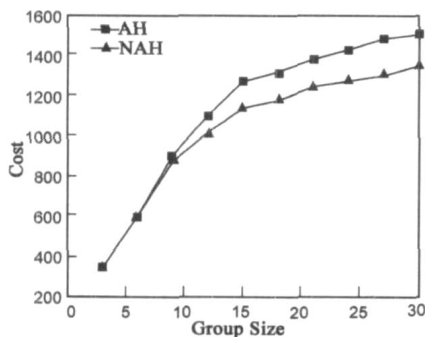


图1 不同组规模下解的费用对比图

图1是节点数为60, 组成员数从3变到30时, AH算法和NAH算法找到的最优解费用的对比图. 图2是两种算法的时耗对比图. 如图所示, NAH算法的解的费用和时耗都低于AH算法, 且这种优势随着组成员数目的增多而更加明显. 这说明在寻找全局最优解和收敛

速度上, NAH算法比AH算法更优秀. 这是因为NAH算法的信息素不用进行区分, 蚂蚁在选择链路时得到的信息更加全面, 有利于算法向最优解快速收敛.

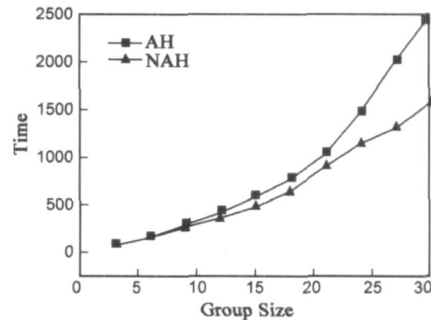


图2 不同组规模下算法收敛时间对比图

为了进一步验证算法的有效性, 本文进行了扩展性实验, 逐步增大网络规模, 节点数从20变到100, 比较搜索到的解的费用和时耗, 得到了如图3和4所示的实验结果.

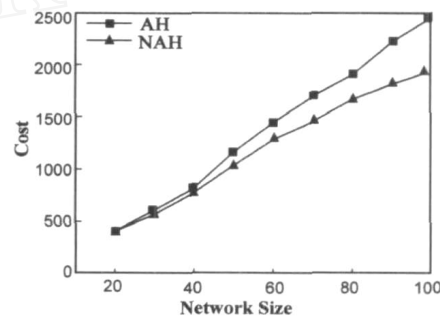


图3 不同网络规模下解的费用对比图

图3是两种算法在不同规模的网络上运行时收敛时间的对比图, 图4是图3所示试验中两种算法的时耗对比图. 实验中, 组成员数量设定为拓扑节点数的15%. 由图3和4可以看出, 在各种规模的网络拓扑上, NAH算法找到的解的费用和时耗都小于AH算法, 且当拓扑规模比较大, 这种优势就越明显. 这进一步证明了NAH算法在搜索全局解和收敛速度上要优于AH算法.

最后, 我们对两种算法进行了寻解成功率的统计和比较, 得到表1.

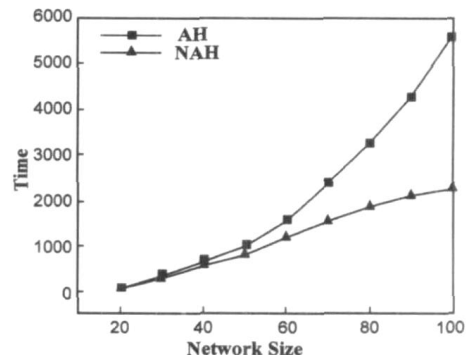


图4 不同网络规模下算法收敛时间对比图

表 1 两种算法成功解决问题的百分比

拓扑规模	组成员占 15 %		组成员占 30 %	
	AH	NAH	AH	NAH
50	95 %	97 %	92 %	95 %
70	94 %	96 %	91 %	95 %
90	94 %	97 %	92 %	96 %
100	95 %	98 %	93 %	95 %

表 1 表示了两种算法特定网络环境下分别解决 100 个度约束组播路由问题实例时,成功解决的问题的数量,用百分比来表示.在随机分配度约束的网络上进行实验.由表 1 的数据我们可以看到,NAH 算法的成功率高于 AH 算法的成功率.这是同样是因为 NAH 算法实际搜索空间更大的缘故.

7 结语

度约束组播路由问题是一个具有现实意义的 NP 完全问题.本文研究了有度约束的最小代价组播树问题,并提出了一个新型蚁群算法来解决这个问题.新算法每次选择一条链路加入组播子树,同时考虑加入点的度约束情况,然后进行适当的调节,既扩大了实际搜索空间又降低了算法的空间复杂度.仿真结果显示,新算法降低了时间开销,提高了求解的成功率,并能以更大的概率找到全局最优解.下一步的工作是研究更加复杂的度约束组播问题,如考虑组播组成员动态变化的度约束组播问题,满足 QoS 要求的度约束组播路由问题等.

参考文献:

- [1] 黄东军,陈松乔.基于源根组播的多点视频会议系统模型及其实现[J].电子学报,2005,33(1):47-51.
- [2] 章森,徐明伟,吴建平.应用层组播研究综述[J].电子学报,2004,32(12A):22-23.
- [3] Martin Zitterbart. Multicast Communication[M]. Morgan Kaufmann,2001.
- [4] L H Sahasrabudde,B Mukherjee. Multicast routing algorithms and protocols:A tutorial[J]. IEEE Network,2000,14(1):90-102.
- [5] Hans-Juergen Proemel,Angelika Steger. The Steiner Tree Problem:A Tour Through graphs,Algorithms,and Complexity[M]. Vieweg Teubner Verlag,2002.
- [6] Fred Bauer,Anujan Varma. Degree-constrained multicasting in point-to-point networks[A]. Proceedings of IEEE INFOCOM[C]. 1995,vol. 1:369-376.
- [7] Stefan Vo. Steiner's problem in graphs:heuristic methods[J]. Discrete Applied Mathematics,1992,40(1):45-72.
- [8] H Takahashi,A Matsuyama. An approximate solution for the Steiner problem in graphs[J]. Math Japonica,1980,24(6):573-577.
- [9] Sung-Jin Chung,Sung-Pil Hong. Algorithms for the degree-constrained multicast trees in packet-switched networks[A].

IEEE GLOBELCOM '98[C]. 1998. 1054-1059.

- [10] 石坚,邹玲.遗传算法在组播路由选择中的应用[J].电子学报,2000,28(5):88-90.
- [11] 王征应,石冰心. QoS 组播路由的启发式遗传算法[J].电子学报,2001,29(2):253-256.
- [12] 王新红,王光兴.基于遗传算法的时延受限代价最小组播路由选择方法[J].通信学报,2002,3(3):112-117.
- [13] 郭伟,席裕庚.基于精确罚函数法的遗传算法求解时延约束组播路由问题[J].电子学报,2001,16(4):75-78.
- [14] 刘莹,刘三阳.多媒体通信中带度约束的多播路由算法[J].计算机学报,2001,24(4):62-65.
- [15] 潘耘,王行刚,冯烟利,余镇危.求解带度约束多播路由问题的启发式遗传算法[J].通信学报,2007,28(1):134-141.
- [16] Marco Dorigo, Thomas Stützle. Ant Colony Optimization[M]. The MIT Press,2004.
- [17] 段海滨.蚁群算法原理及其应用[M].北京:科学出版社,2005.
- [18] 杨云,徐佳,高飞,陆璐,刘凤玉.基于蚁群系统的多 QoS 约束组播路由算法[J].小型微型计算机系统,2006,27(11):2030-2035.
- [19] 孙力娟,王汝传.基于蚁群算法和遗传算法融合的 QoS 组播路由问题求解[J].电子学报,2006,34(8):341-346.
- [20] 邢文训,谢金星.现代优化计算方法(第 2 版)[M].北京:清华大学出版社,2005.
- [21] Yiag Liu,Jiangping Wu,Ke Xu,Mingwei Xu. The degree-constrained multicasting algorithm using ant algorithm[A]. Proceedings of 10th International Conference on Telecommunications[C]. 2003. 370-374.

作者简介:



葛连升 男,副教授,山东大学网络与信息中心副主任,兼任 CNGI-CERNET2 济南核心节点主任,山东省互联网协会副理事长等职.1967 年 5 月出生于山东莒县.1989 年毕业于山东大学数学系,现为山东大学计算机科学与技术学院在职博士生,主要研究兴趣为网络路由与组播算法、网络与系统安全等.
E-mail:lsge@sdu.edu.cn



王 华 男,博士、副教授.1962 年 12 月出生于山东济南.现任山东大学计算机科学与技术学院系统结构研究所副所长.主要研究方向为网络路由与组播协议机制,网络优化模型与算法,网络仿真.

王海洋 男,教授,博士生导师.1965 年 2 月生于山东文登.先后在山东大学、中国科学院计算所获得学士、硕士、博士学位,现任山东大学网络与信息中心主任,兼任山东大学计算机科学与技术学院副院长,主要研究方向为计算机软件与理论、智能 web、工作流.