

多播路由算法 MPH 的时间复杂度研究

蒋廷耀^{1,2}, 李庆华²

(1. 三峡大学电气信息学院, 湖北宜昌 443002; 2. 华中科技大学计算机学院, 湖北武汉 430074)

摘 要: 多播通信是从一个源点同时向网络中的多个成员发送分组的通信服务, 一个最小代价的多播路由算法是 NP 完全的, 在时间敏感的应用中其运行时间是一个关键问题. MPH (Minimum Path Cost Heuristic) 算法是一个著名的启发式最小代价多播路由算法, 本文对该算法进行了理论分析和证明, 并做了广泛的仿真实验, 结果表明其时间复杂度是 $O(m^2n)$ 而不是过去文献中所给出的 $O(m^2n+e)$.

关键词: 多播路由; 最小成本; 时间复杂度

中图分类号: TP393

文献标识码: A

文章编号: 0372-2112 (2004) 10-1706-03

A Corrigendum to the Time Complexity of Multicast Routing Algorithm MPH

JIANG Ting-yao^{1,2}, LI Qing-hua²

(1. College of Electrical Engineering and Information Technology, China Three Gorges University, Yichang, Hubei 443002, China;

2. College of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China)

Abstract: Multicasting is a communication service that allows a source to efficiently transmit copies of data packet to a set of destination nodes. The run time of multicast routing algorithm is a key problem in real time applications since finding a minimum cost multicast tree is NP-completeness. MPH (Minimum Path Cost Heuristic) algorithm is a famous heuristic solution to multicast routing. In this paper, we show that the $O(nm^2+e)$ time complexity of MPH in previous literatures is not correct by theory analysis and extensive simulation, where n is number of network nodes, m is number of destination nodes and e is number of network edges. Furthermore, a precise time bound $O(nm^2)$ is proposed.

Key words: multicast routing; minimum cost; time complexity

1 引言

多播通信 (multicast) 技术是网络中的一个源点同时向网络中的多个成员节点 (端节点) 发送分组的通信服务. 它有利于节省网络资源, 在现代计算机通信中有着重大的应用价值. 多播路由 (又叫多播树) 是从源点到各端节点的路径, 最小成本的多播路由的确定是 NP 完全的. MPH 是一个出色的启发式算法, 在文献 [1~6] 中都有描述, 文献 [2, 3] 提出了改进的 MPH 算法以减少多播树的构造时间. 文 [1] 在将 MPH 算法与其它算法进行性能比较时, 认为 MPH 算法的时间复杂度在端节点到其它节点的最短路径已知的条件下 (后文都是基于这个条件) 是 $O(m^2n+e)$, n 是网络节点数, m 是端节点数, e 是网络中的边数. 文 [2, 3] 运用文 [1] 的分析方法和结论也进行了有关算法的性能分析和比较. 本文认为文 [1~3] 对 MPH 算法和有关算法的时间复杂度量度是不正确的, 理论分析和仿真结果表明 MPH 的时间复杂度应该是 $O(m^2n)$.

2 MPH 算法

通信网络可模型为一个带权无向图 $G=(V, E, c)$, V 是节点集合, E 是边的集合, 边代表节点间的通信链路. G 中节

点个数为 n , 节点标记为 $v_i (i=1, 2, \dots, n)$, 边标记为 $\langle v_i, v_j \rangle$, $(1 \leq i < j \leq n)$, $\text{cost}(\langle v_i, v_j \rangle)$ 表示节点 v_i, v_j (或 v_j, v_i) 间的边的成本, G 的成本 $c = \sum_{\langle v_i, v_j \rangle \in E} \text{cost}(\langle v_i, v_j \rangle)$.

定义 1 节点 v_i, v_j 之间的最短路径是连通 v_i, v_j 的路径中各边成本之和最小的一条路径. 节点到树的最短路径是指该节点到树中各节点的最短路径中成本最小的那一条路径.

定义 2 最小成本多播树. 给定一个无向图 $G=(V, E, c)$, 源节点 s , 一个端节点集合 $Z \subseteq V$, 寻找 G 的一个子图 (树) $T=(V_t, E_t, c_t)$, 若满足源节点到其它任一端节点存在一条路径, Steiner 节点 (源节点和端节点以外的节点) 的度大于或等于 2, $T \subseteq G, V_t \subseteq V, E_t \subseteq E, Z \subseteq V_t$, 则 T 是关于 s 和 Z 的一棵多播树. 若 T 的成本 $c_t = \sum_{\langle v_i, v_j \rangle \in E_t} \text{cost}(\langle v_i, v_j \rangle)$ 是有关于 s 和 Z 的多播树中最小的, 则 T 是最小成本多播树.

MPH 算法是一个启发式最小成本多播树算法, 其基本思想如下:

(1) 部分多播树 T 开始只包括源节点 s .

(2) 从余下的端节点中选取到树 T 的最短路径 (如有多个则随机选择一个), 端节点及路径上的所有节点并入 V_t , 路径上的边并入 E_t .

(3)重复步骤(2), (3)直到 Z 中所有的节点包含到 V_t 中, 此时的 T 即为近似最小成本多播树.

3 算法时间复杂度分析

文[1~3]认为 MPH 算法中, 对每一个即将加入到部分多播生成树的端节点的选择, 最多需进行 mn 次最短路径的比较, m 个端节点一个一个地加入, 共需进行 m^2n 次比较. 另外, 将选定的端节点到树 T 的最短路径上的节点和边加入到多播生成树的次数最多是 e . 因而, MPH 算法的时间复杂度为 $O(m^2n + e)$.

我们把 MPH 算法的操作分为三部分: 端节点的加入所需的最短路径的比较次数; 最短路径上的边的加入次数; 最短路径上的节点的加入次数.

(1) 最短路径的比较次数

MPH 算法中, 端节点到部分多播生成树的最短的最短路径的确定需进行的最大比较次数为:

$$O\left(\sum_{i=0}^m (n-m+i)(m-i)\right) \\ = O\left(\frac{3m^2n + 2m^3 - 3m^2 + 3mn - m}{6}\right) = O(m^2n)$$

文[3]基于 MPH 算法介绍了一种改进的算法 FMPH, 有利于减少最短路径的比较次数. 但文[3]关于比较次数为 $O(m^2k)$, k 为多播生成树中任意一个端节点的路径节点个数的结论是值得商榷的. 因为根据原文定义 2, k 可能是 0, 最坏情况下也可能是 $O(n)$. 作为对算法复杂性的度量, 任意的 k 是不妥当的.

(2) 边的加入次数

当从余下的端节点确定了到部分多播生成树的最短路径后, 还需将该路径上的边并入部分多播生成树. 为了确立这部分操作所需考察的边的数目, 我们先证明下面的定理.

定理 1 设 $T' = (V', E')$ 是网络图 $G = (V, E)$ 的一棵子树(可扩展为子图), 节点 $v_m \in V, v_m \notin V', v_m$ 到树 T' 的最短路径 $\text{path}(v_m, T') = (V'', E'')$, 则 $E' \cap E'' = \emptyset$.

证明 用反证法. 由定义 1, 设 v_m 到树 T' 的最短路径是 v_m, v_i 节点间的最短路径, $v_i \in V'$. 如果 $E' \cap E'' \neq \emptyset$, 则至少有一条边 $\langle v_i, v_j \rangle \in E', \langle v_i, v_j \rangle \in E''$, 即有 $v_i \in V', v_j \in V'$, 则 v_i, v_j 两个节点至少有一个节点不是 v_i , 不妨设该节点是 v_j . 由于 v_j 是 v_m 到 v_i 的最短路径上的节点, 则 v_m 到 v_j 的距离小于 v_m 到 v_i 的距离, 这说明 v_m 到树 T' 的最短路径是 v_m, v_j 节点间的最短路径, 这同 v_m 到树 T' 的最短路径是 v_m, v_i 节点间的最短路径的假设是矛盾的, 所以定理 1 成立.

由定理 1, MPH 算法中向部分多播生成树加入选定的最短路径上的边时所需处理的边的最大数目应是 $n-1$. 因为最终生成的多播树最大是包括 n 个节点的一棵生成树, 且该树在构造时所处理的边是没有重复的. 而文[1~3]认为处理的边的最大数目是 $O(e)$, 对稀疏图, 这部分计算量对算法总的时间复杂性的影响不是太大, 但是对稠密图却可能使对算法的性能分析得到错误的结论, 因为边的总数 e 是 $O(n^2)$ 条. 例如, 稠密图中当 $m \ll n$ 时边的加入的计算量成为算法的主要

部分, $O(m^2n + e) = O(e) = O(n^2)$, 是实际计算时间的 $O(n)$ 倍.

另外, 文[2]在对 LSMPH 算法、文[3]在对 FMPH 算法的时间复杂度的分析上对边的度量存在同样的问题, 这里不再赘述.

(3) 最短路径上的节点的加入次数

多播生成树的节点数最多是 n , 故节点的加入所需的最大次数是 $O(n)$.

综上所述, MPH 算法的时间复杂度应是 $O(m^2n + n + n) = O(m^2n)$, 而 $O(m^2n + e)$ 的结论是不正确的.

4 仿真实验

为了更好的说明文[1~3]的结论 $O(m^2n + e)$ 中的 e 相对于 n 的偏差程度, 我们进行了仿真实验. 实验中采用 Waxman 在文献[7]中介绍的方法生成随机网络图. n 个节点中任意一对节点 u, v 之间是否存在一条边由连通概率 $p(\langle u, v \rangle)$ 确定, $p(\langle u, v \rangle) = \beta e^{-d(u,v)/L\alpha}$, $\alpha, \beta \in (0, 1)$. 边长即成本 $d(u, v)$ 随机分布在 $(0, L)$ 中, L 表示边长的变动期间. α, β 是调整网络特性的参数, α 增加, 长边相对短边的比增加; β 增加, 节点的度增加, 即边越稠密. 共进行了 3 组对照实验, $n = 300, L = 100, \alpha = 0.2, \beta$ 分别为 0.9, 0.6, 0.3. 每一组参数生成 300 个随机网络图. 对每一个网络图, 记录网络图的总边数, 300 次实验的平均值记为 e ; 随机选择指定数目的端节点, 记录 MPH 算法构造多播生成树时所实际处理的边数, 平均值记为 $real$.

比值 $e/real, n/real$ 分别示于图 1. 从图 1(a) 可以看出用 e 来作为 MPH 算法的边的处理次数是实际处理次数的 10^2 倍数, 而且边越稠密, 偏差越大; 端节点数越少, 偏差越大. 而从图 1(b) 可以看出以 n 来作为 MPH 算法的边的处理次数是合理的, 并且几乎不受边的稠密程度的影响.

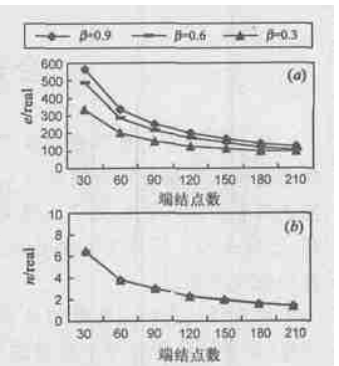


图 1 (a). $O(e)$ 相对实际处理边数的偏差程度; (b). $O(n)$ 相对实际处理边数的偏差程度

5 结论

理论分析和仿真结果表明多播路由算法 MPH 的时间复杂度应该是 $O(m^2n)$, 而文[1~3]对 MPH 算法和有关算法的时间复杂性度量是不正确的.

参考文献:

- [1] S Ramanathan. Multicast tree generation in networks with asymmetric links[J]. IEEE ACM Trans. On Networking, 1996, 4(4): 558-568.
- [2] 李汉兵, 喻建平, 谢维信. 局部搜索最小路径费用算法[J]. 电子学报, 2000, 28(5): 92-95.

- [3] 胡光岷, 李乐民, 安红岩. 最小代价多播生成树的快速算法[J]. 电子学报, 2002, 30(6): 880—882.
- [4] Pawel Winter. Steiner problem in networks: a survey [J]. Networks, 1987, 17(2): 129—167.
- [5] F K Hwang. Steiner tree problems [J]. Networks, 1992, 22(1): 55—89.

作者简介:



蒋廷耀 男, 1969 年 4 月出生于湖北宜昌, 副教授, 目前在华中科技大学计算机学院、国家高性能计算中心(武汉)攻读博士学位, 主持或参与过多项国家自然科学基金和 863 项目的研究, 当前研究方向: 移动计算, 计算机网络及信息安全, 高性能计算. Email: tingyao-jiang@263.net

- [6] H F Salama, D S Reeves. Evaluation of multicast routing algorithms for real-time communication on high speed networks [J]. IEEE Journal on selected areas in communication, 1997, 15(3): 332—349.
- [7] M Waxman. Routing of multipoint connections [J]. IEEE Journal on Selected Areas in communication, 1988, 6(9): 1617—1621.



李庆华 男, 1940 年 7 月出生于湖北武汉, 教授, 博士生导师, 研究方向: 网络计算, 高性能计算. Email: liqh@263.net

2005 软件过程技术国际研讨会征文通知

(2005 年 5 月 24—26 日 中国·北京)

<http://www.cnsqa.com/~spw2005>

<http://www.iscas.ac.cn/~spw2005>

会议主题: 统一软件过程宏观与微观研究体系

随着软件与信息系统在世界上发挥越来越重要的作用, 如何确保在可接受的时间与成本下, 按照可预测的进度, 开发出能随着应用环境的变化而确定、迅速地改进的高质量的系统, 已经成为人们日益关注的焦点, 并为软件过程技术提供了新的挑战与机遇。

本次研讨会的主要目的是为整合这两条互补的软件过程研究(微观过程研究与宏观过程研究)路径提出建议, 使软件过程工程成为规范、有序的学科。这种整合将给我们一些有益的启示, 如: 如何通过分析、优化低层的过程行为, 对高层的过程行为进行预测与改进。这种整合也将探索如何更好地将基于明确知识的客观的微观过程, 同基于隐含知识的主观的协作过程整合起来。

这次研讨会将提供一个论坛, 系统展示当今软件过程研究成果, 共同洞察软件过程未来方向, 朝着统一软件过程宏观与微观研究体系的目标迈进。研讨会内容将包括: 世界最顶尖软件过程研究者与使用者的特邀报告; 针对软件过程挑战与解决方法的论文报告; 工具演示; 和关于软件过程研究方向的专题讨论会。

会议论文计划收录在 Springer 出版的 Lecture Notes in Computer Science 中。

征文范围:

欢迎有关软件过程的经验、描述和方法等各相关研究领域的论文。例如, 过程内容(文档驱动的、变化驱动的、体系结构驱动的、风险驱动的、涉众驱动的……); 过程表示与分析; 过程工具和度量; 过程中的人为因素; 等等。

重要日期:

论文提交: 2005 年 1 月 17 日

录用通知: 2005 年 3 月 21 日

最终论文: 2005 年 4 月 25 日

程序委员会主席:

中国科学院软件研究所 李明树研究员

美国南加州大学 Barry Boehm 教授

美国马萨诸塞大学 Leon Osterweil 教授

论文提交:

论文需用英文书写, 长度为十页或十页以内, 格式为 PDF、LaTeX 或 MS Word, 并送到以下地址: (建议通过 Email 投寄电子版论文)

100080 北京中关村南四街 4 号, 中国科学院软件研究所

2005 软件过程技术国际研讨会组织委员会

E-mail: spw2005@iscas.ac.cn