

一个有效的时延约束最小代价多播路由算法

陈月云, 刘亲亲

(北京科技大学信息工程学院, 北京 100083)

摘要 基于时延约束多播路由问题考虑链路代价, 提出一种新的时延约束最小代价路径(DCM-CA)算法, 作为搜寻节点间最短路径的算法; 在此基础上又改进了基于代价-时延比率(CDR)函数的有效中心节点选择算法; 基于CBT树, 应用上述2种算法提出一种基于中心选择的时延约束最小代价多播路由(CS-DCMCMR)算法, 该算法在搜寻路径和中心节点选择的问题上同时考虑路径的时延和代价。仿真证明CS-DCMCMR算法的时间复杂度为 $O(m \log n)$, 与CSDVC算法和CCLDA算法相比, 该算法在没有增加复杂度和满足时延及时延抖动约束的条件下, 较大程度地减小了最终多播树的总代价。

关键词 时延; 时延抖动; 多播树; 最小代价

DOI 10.3969/j.issn.1009-3516.2011.03.015

中图分类号 TP 301.6 **文献标识码** A **文章编号** 1009-3516(2011)03-0068-05

空天信息系统是以外层空间的各类卫星、临近空间的各种飞行器为平台, 利用网络技术把平台上的侦察监视、导弹预警、通信中继、导航定位和气象观测等载荷设备有机地连接在一起, 组成的空天一体的信息系统网络^[1-2]。空天网络环境的复杂性和功能的多元性要求更高的通信质量。如何保障不同用户的QoS需求和高效利用空间资源成为通信网络路由算法需要解决的问题。

多播通信技术是一种节省网络资源, 减少数据冗余的有效解决方案, 其往往有严格的QoS要求, 最终的端到端延迟和时延抖动是重要的QoS参数。多播路由算法的关键就是如何构建一棵正确而有效的多播路由树去满足不同多播通信业务应用的QoS要求^[3-4]。许多研究致力于多播时延和时延抖动约束问题, 该问题已被正式定义为DVBMT^[4](Delay and delay Variation Bounded Multicast Tree, DVBMT)问题, 并证明是NP-完全问题。时延和时延抖动约束算法^[5](Delay and Delay Variation Constraint Algorithm, DDVCA)是已知的解决DVBMT问题的最佳方案。

1 系统模型

1.1 问题定义

DDVCA基于CBT(Core Based Tree, CBT)和最小时延路径算法。CBT构造的共享树可以大幅减少路由器上存储的多播转发状态信息, 但是由于所有的多播信息都要通过中心路由器进行转发, 所以CBT的缺点实际上是需要选择一个适当的中心路由器^[5]。中心节点的选择是构造多播树的关键, 它影响到最终的多播树的链路结构和链路代价。DDVCA选择具有最小时延抖动的节点作为中心节点, 然而本文认为更应该考虑时延抖动约束。另外, 最小时延路径算法针对单一度量, DDVCA在搜索路径时仅仅考虑了时延这个可加性参数, 而没有考虑到链路的代价, 这有可能导致算法为了满足时延和时延差异的限制而使用代价过大的路径, 从而使得最终得到的多播树代价过大。

* 收稿日期: 2010-10-12

基金项目: 北京市自然科学基金资助项目(4102041); 博士后专项基金资助项目(20090006110014)

作者简介: 陈月云(1966-), 女, 河北景县人, 副教授, 主要从事无线和移动通信理论及无线通信新技术等研究。

E-mail: chenyy2@sina.com

所以,本文同时考虑时延和代价,提出基于核心选择的时延约束最小代价多播路由(Core Selection for Delay Constrained and Minimum Cost Multicast Routing, CS-DCMCMR)算法。该算法在较小程度增加时延的情况下有效减小了多播树的总代价。

1.2 模型构建

本文用 $G=(V,E)$ 表示无向加权网络图,其中 V 表示网络中的所有节点的集合, E 表示网络中节点间链路的边的集合。对于每一条边 e ,定义边上的权重函数为:每条链路 $e \in E$ 上有一个代价函数 $C(e):E \rightarrow R^+$ 和一个延迟函数 $D(e):E \rightarrow R^+$ 与之关联;定义函数 $C(\cdot)$ 为链路上传输一条信息所需要占用带宽或消耗功率的开销,是要最小化的度量;定义函数 $D(\cdot)$ 为链路上传输一条信息所需要的时延,是要约束的度量。

对于已知的网络 $G(V,E)$ 和多播源 $s \in V$,目的节点集 M ,时延上限和时延抖动约束 δ 。若一棵多播树覆盖 $s \cup M$ 的生成树 $T = \{T_1, T_2, \dots, T_N\}$,使得树代价和时延满足如下条件:

$$\min \text{cost}[T_i] = \sum_{\substack{e \in \text{Path}(s,t) \\ t \in M}} C(e), \quad (1 \leq i \leq M)$$

$$\text{s. t.} \quad \text{Delay}[s, v_i] = \sum_{e \in P(s, v_j)} D(e) \leq \Delta \quad (\forall v_i \in M) \quad (1)$$

$$|\max \text{Delay}[s, v_i] - \min \text{Delay}[s, v_j]| \leq \delta \quad (\forall v_i, v_j \in M)$$

2 时延约束最小代价路径算法(DCMCA)

文献[3]和文献[8]都采用 Dijkstra 算法作为最短路径算法,而 Dijkstra 算法针对单一度量,不适用于本文。MCP-IA 算法考虑了代价和时延这 2 个可加性参数约束,它利用一个参数 a 将链路上的代价和时延这 2 个权值形成一个合成权值 $w = w_1 + aw_2$, w_1 和 w_2 分别指链路上的代价和时延^[7]。所以,本文基于 MCP-IA 算法的原理对 Dijkstra 算法进行改进后提出时延约束最小代价路径算法(Delay Constraint Minimum Cost Algorithm, DCMCA)。DCMCA 采用 Dijkstra 算法来寻找最短路径的,假设要寻找的最终路径的代价要在满足时延约束的上限 C_2 时尽量优化到最小。因此,只要选择合适的参数 a ,就能找到一条最优的路径,既能满足时延的约束又能使代价最小且满足代价约束上限。参数 a 越大, w_1 的值越大,同时 w_2 的值越小。这个关系的正确性以及 a 的值取多大可以参看文献[9]。

DCMCA 的基本思想是:假设 w_1 和 w_2 分别代表链路的代价和时延。首先采用 Dijkstra 算法以代价函数 $C(e)$ 为权值搜寻节点间的最小代价路径 p ,并计算路径 p 上的端到端时延 $d(p)$,验证 $d(p)$ 是否满足时延约束上限 C_2 。如果路径 p 上的时延满足时延约束,那么这条路径 p 就是所求路径,否则就采用 Dijkstra 算法以时延函数 $D(e)$ 为权值搜寻节点间的最小时延路径 q 。同样计算路径 q 上的时延 $d(q)$,验证 $d(q)$ 是否满足 C_2 。如果满足,则 q 就是所求路径。否则 DCMCA 程序就开始执行下面的循环,在每次循环中,路径 p 被一个拥有更低代价或更小时延的路径 r 更新,路径 r 是最短路径。循环进行直至找到更好的路径来更新。

3 中心节点选择算法

3.1 中心节点选择算法模型

中心节点的选择问题是 CBT 多播路由算法中的一个重要问题,选择不同位置的中心节点必定生成不同的 CBT,进而具有不同的性能指标。因此如何选择一个最佳的中心节点实际上是决定多播性能指标的关键。本文采用一个代价-时延比率函数 $R_{cd}(e)$ 作为选择中心节点的权重函数,用来捕获因时延约束造成的风险水平和最小代价之间的权衡^[9]。该函数定义为一条链路 $e=(u,v)$ 上的权重函数, $e=(u,v)$ 为节点 $u \in V$ 到节点 $v \in V$ 的链路,每一条链路却是双向的:

$$R_{cd}(e) = \frac{C(e)}{D(e)} \quad (2)$$

文献[10]的 CSDVC(Core Selection Delay and Delay Variation Constrained)算法是从目的节点集 M 中选择具有最小 Average $R_{cd}(e)$ 值的节点作为中心节点。本文认为这样缩小了中心节点的选择范围,可能会遗漏最佳的中心节点,使整棵多播树的代价未必优化到最小。所以,CS-DCMCMR 算法采用从多播节点集 V

中选择候选中心节点的方法,选择具有最小 $\text{Sum } R_{\text{CD}}(e)$ 值的节点作为中心节点。

CS-DCMCMR 算法计算 $\text{Sum } R_{\text{CD}}(e)$ 值的过程是:首先在多播节点集 V 中任选一节点 v_i ,利用计算每一多播节点 $v_i \in V$ 到每一目的节点 $v_j \in M$ 间 DCMCA 路径上 $\text{Sum } R_{\text{CD}}[v_i]$ 值。这里,本文只计算 $\text{Sum } R_{\text{CD}}[v_i]$,不采用文献[10]中的方法计算最短路径上 $\text{Average } R_{\text{CD}}[v_i]$ 。

$$\text{Sum } R_{\text{CD}}[v_i] = \sum_{j=1, j \neq i}^{|V|-1} R_{\text{CD}}[v_i, v_j] \quad (v_i \in V, v_j \in M, v_i \neq v_j) \quad (3)$$

3.2 中心节点选择算法的 2 个阶段

本文的中心选择算法过程包含 2 个阶段:中心节点预选阶段和中心节点选择阶段。

3.2.1 中心节点预选阶段

循环计算每个节点到所有目的节点间的 DCMCA 路径上的 R_{CD} 值之和 $\text{Sum } R_{\text{CD}}[v_i]$,然后按 $\text{Sum } R_{\text{CD}}[v_i]$ 值的大小优先级升序地存入序列 Q 中,并记录节点标号。序列 Q 就是一个递增顺序排列的候选中心节点序列,序列中的第一个节点具有最小值,将其作为中心节点选择程序的第一个被检测的节点。

3.2.2 中心节点选择阶段

检验第一个节点与所有目的节点间 DCMCA 路径上的时延和时延抖动是否满足约束上限。具体过程如下^[10]:首先从 Q 序列里挑选出第一个节点 v_i 作为中心节点的候选节点,这个候选节点具有最小的 $\text{Sum } R_{\text{CD}}[v_i]$ 值,验证从源节点 s 经过候选中心节点 v_i 到达目的节点间 DCMCA 路径上的最大总时延满足时延约束,并且这时多播树的最大时延抖动满足时延抖动上限 δ ,我们选择这个候选节点为中心节点($v_c = v_i$)。否则重新从 Q 序列中挑选下一个候选节点,此时再检验经过该节点的 DCMCA 路径上的总时延和最大时延抖动是否满足约束上限。执行该程序直到找到合适的中心节点结束循环。

4 算法的实现与仿真

4.1 算法实现

输入:给定一个网络图 $G=(V,E)$,一个目的节点集 M ,一个源节点 s ,一个端到端的时延约束上限 Δ 和一个时延抖动约束 δ 。

输出:一棵时延和时延抖动约束下的最小代价多播树 T 。其中,从源节点到各个目的节点间的最大端到端时延 $\leq \Delta$,从中心节点到各个目的节点间的最大多播时延差异 $\leq \delta$,并且最终的多播树的代价优化到最小。

CS-DCMCMR 算法的操作过程是:首先初始化变量、设置时延和时延抖动约束参数;然后采用时延约束最小代价算法建立最小代价-时延比率树;再在 DCMCA 的基础上采用有效的中心节点选择算法选择一个有效的中心节点 v_c 。检验中心节点 v_c ,如果 v_c 为 \emptyset ,算法返回到该程序的开始,重新设置时延和时延抖动的阈值,继续执行程序直到选择一个合适的中心节点;否则继续执行程序构建最终的多播树。首先采用 DC-MCA 建立源节点 s 和中心节点 v_c 间的时延约束最小代价路径,接着再采用 DCMCA 链接这个中心节点 v_c 到所有目的节点间的时延约束最小代价路径。

4.2 仿真结果分析

下面用图 1 所示的例子来展示 CS-DCMCMR 算法及文中提到的一些算法的实现过程。在图 1(a)原始网络图的描述中,沿着每一条边上的每一对数字 c/d 表示该边上的代价(c)和时延(d)^[10]链路代价范围为 $[1,3]$,时延范围为 $[1,5]$ 一个单位时延为 10 ms。标号为 0 的节点表示源节点,目的节点集 M 设置为: $M=5,6,7,8$ 。时延约束 Δ 设置为 20,时延抖动 δ 设置为 9。图 1(b)描述的是 CS-DCMCMR 算法在原始图的基础上采用时延约束最小代价路径算法(DCMCA)和有效的中心节点选择算法建立的多播树。该算法采用 DCMCA 计算每一个多播节点 v_i 到所有目的节点 v_j 间路径上的 $\text{Sum } R_{\text{CD}}[v_i]$ 值,且该路径上的时延和时延抖动满足时延和时延抖动约束。最终选择标号为 6 的节点作为中心节点,图中带箭头的黑色粗线代表多播信息的传输链路。图 1(c)描述的是 CSDVC 算法,它采用 Dijkstra 算法来计算每一个目的节点 v_i 到其他目的节点 v_j 间路径上的 $\text{Average } R_{\text{CD}}[v_i]$ 值,最终选择具有最小值的标号为 7 的节点作为中心节点。图 1(d)描述的是文献[7]提出的耗费受限的最短时延路径算法 CCLDA 算法建立的多播树,它采用代价约束最小时延路径算法来建立节点间路径,选择具有最小时延抖动的标号为 2 的节点作为中心节点。图 1(e)描述的是比较经典的 DDVCA 算法建立的多播树,同样计算节点间的最短路径,选择具有最小时延抖动的标号为 2 的节点作

为中心节点。

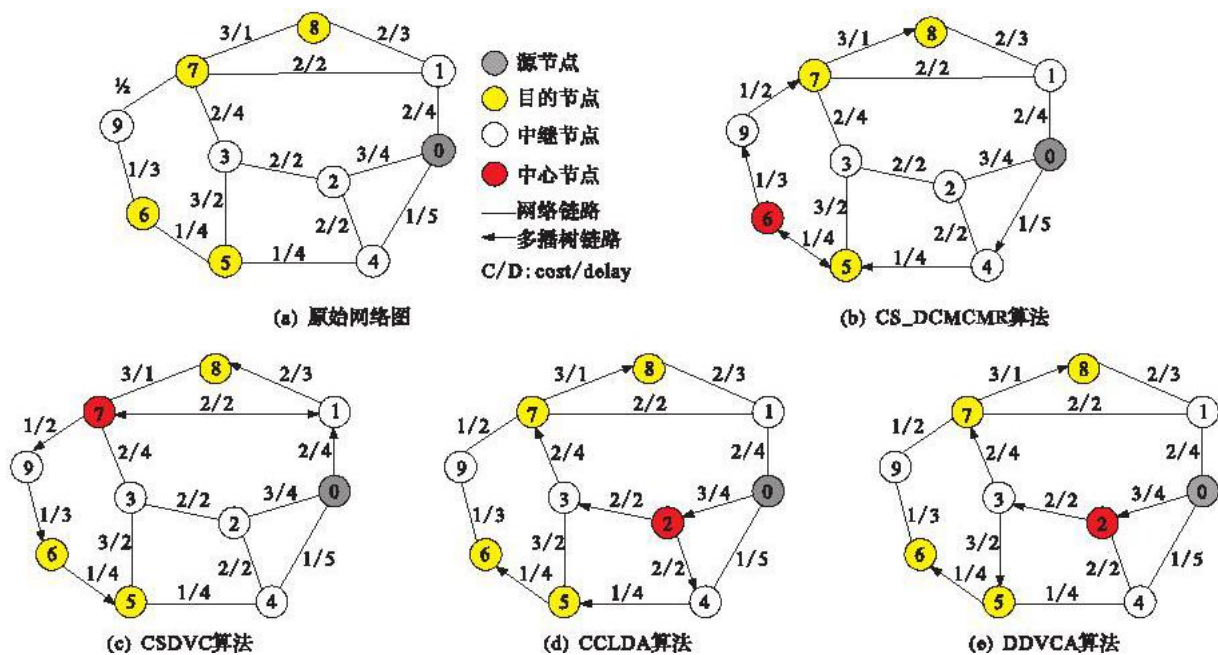


图 1 CS_DCMCMR 算法和其他算法的比较

Fig. 1 Comparison of CS_DCMCMR with other algorithm

4.2.1 算法仿真结果比较

在仿真中,设置时延约束 Δ 为 20,时延抖动 δ 为 9。在上述约束条件下仿真本文提出的算法和文献中提到的其他算法,并比较分析它们的仿真结果。图 1 表示的时延是用公式(4)分别计算源节点 s 到每个目的节点 v_j 间的 DCMCA 路径上的时延确定,而整棵树最大的时延抖动就可以用式(5)计算确定。

$$\text{Delay}[s, v_i] = \sum_{e \in P(s, v_i)} D(e), \quad v_i \in M \tag{4}$$

$$\delta_T = \max_{v_i, v_j \in M} \{ |\text{Delay}[s, v_i] - \text{Delay}[s, v_j]| \} = \max \{ \delta_{5,6}, \delta_{8,6}, \delta_{8,7}, \delta_{7,6} \} \tag{5}$$

从表 1 中可以明显看出,本文的 CS_DCMCMR 算法所产生的多播树的总代价为 8,其代价明显小于其他算法的总代价,而该算法所产生的多播树的总时延只比 DDVCA 算法增加 2 s,同时,时延抖动也比产生最小时延抖动的 DDVCA 和 CCLDA 增加 2 s。可见,CS_DCMCMR 算法以少量增加时延和时延抖动为代价,较大程度减小了多播树的总代价。

表 1 算法仿真结果比较

Tab. 1 Comparison of simulation results

	总代价	总时延/s	最大时延/s	δ
CS_DCMCMR	8	19	19	6
CSDVC	9	24	15	9
CCLDA	4	21	14	4
DDVCA	4	17	12	4

4.2.2 复杂度分析

在 $O(\cdot)$ 的渐进标记法中^[6],由 Dijkstra 算法产生的最短路径树算法的时间复杂度是 $O(n^2)$ ^[5]。DDVCA、CSDVC 都是采用 Dijkstra 算法产生的最短路径树,所以分析 DDVCA 的时间复杂度是 $O(mn^2)$,CSDVC 的时间复杂度是 $O(m \log n)$ 。本文算法采用 DCMCA 算法计算节点间的路径,而 DCMCA 算法是在 Dijkstra 算法的基础上改进的,并没有增加时间复杂度,所以 DCMCA 算法时间复杂度是 $O(n^2)$,那么本文的 CS_DCMCMR 算法的时间复杂度是 $O(m \log n)$ 。通过比较显示本文算法的时间复杂度与 CSDVC 一样,都小于 DCMCA 和 DDVCA 的时间复杂度。因此证明 CS_DCMCMR 算法在不增加时间复杂度的情况下达到多播树代价最小的目的,是一种有效的时延约束最小代价多播路由算法。

5 结束语

本文分析了多播时延和时延抖动约束问题,在构建多播树时考虑了链路的代价,提出以一个链路代价函数为优化目标的建树系统模型。并由此提出了时延约束最小代价路径算法(DDVCA),用该算法来搜寻节点间的最短路径,为了在满足时延约束的条件下尽量优化代价,又采用一种有效的中心选择方法选出最佳的中

心节点使多播树代价达到最小。最终形成的 CS_DCMCMR 算法所构建多播树的代价小于文中提到的其他算法产生的代价,该算法的时间复杂度为 $O(m \log n)$ 。仿真结果说明 CS_DCMCMR 算法在增加时延和时延抖动幅度不大的情况下很大程度上优化了代价,是一种有效的基于中心选择的时延约束最小代价算法。

参考文献:

- [1] 曹裕华,赵玉普,武晓鹏. 空天信息系统发展现状与趋势研究[J]. 装备指挥技术学院学报,2009(4):50-53.
CAO Yuhua,ZHAO Yupu,WU Xiaopeng. Research on the actual status and development trend of space and air information system[J]. Journal of the academy of equipment command & technology,2009(4):50-53. (in Chinese)
- [2] 李轶璋. 空天网络接入算法设计与仿真[D]. 武汉:华中科技大学,2008.
LI Yizhang. Design and simulation of access algorithm in air & space network[D]. Wuhan:Huazhong university of science and technology,2008. (in Chinese)
- [3] Cheng Hui,Cao Jiannong. Constructing delay-bounded multicast tree with optimal delay variation[C]//IEEE international conference on communications (ICC06). [S. l.]:IEEE press,2006:800-805.
- [4] Manas Ranjan Kabat, Manoj Kumar Patel. A heuristic algorithm for delay variation bounded least cost multicast routing [C]//IEEE Advance computing conference (IACC), [s. l.]:IEEE press,2010:261-266.
- [5] Rouskas G N, Baldine I. Multicast routing with end-to-end delay and delay variation constraints[J]. IEEE journal on selected areas in communications, 1997, 15(3): 346-356.
- [6] Hua Wang, Zhao Shi. Multicast routing for delay variation bound using a modified ant colony algorithm[J]. Journal of network and computer applications,2009,32: 258-272.
- [7] 杨春德,杨孝田. DVBMT 问题的一种改进算法[J]. 计算机应用研究, 2009,26(3): 1059-1063.
YANG Chunde,YANG Xiaotian. An improved algorithm for DVBMT issue[J]. Computer application research,2009,26(3): 1059-1063. (in Chinese)
- [8] 李秉智,李智. 一种新的基于 Dijkstra 算法的 QoS 组播树启发式算法[J]. 重庆邮电学院学报:自然科学版, 2006,18(1):115-118.
LI Bingzhi,LI Zhi. A new QoS multicast tree heuristic algorithm based on dijkstra's algorithm[J]. Chongqing university of posts and telecommunications;natural science edition,2006,18(1):115-118. (in Chinese)
- [9] FENG G, DOULIGERIS C. An efficient approximate algorithm for finding paths with two additive constraints[J]. IEICE transactions on communications, 2002, E85B(6):1143-1151.
- [10] Mohamed Aissa, Abdelfettah Belghith. An efficient core selection algorithm for delay and delay-variation constrained multicast tree design[C]//Proceedings of the IEEE. [S. l.]:IEEE press,2009:201-207.

(编辑:徐敏)

An Effective Delay Constrained and Minimum Cost Multicast Routing Algorithm

CHEN Yue-yun, LIU Qin-qin

(Department of Information Engineering, University of Science and Technology, Beijing 100083, China)

Abstract: Based on the problem of considering the link cost into delay constraint, this paper proposes a new delay constrained minimum cost path algorithm (DCMC) which is used for searching the shortest path between nodes. On this basis, an efficient center node selection method is further modified based on the cost-delay ratio (CDR). Based on application of CBT a center selection for delay constrained minimum cost multicast routing (CS_DCMCMR) algorithm is proposed by using the above two algorithms. In the use of this algorithm, both cost and delay are taken into consideration simultaneously in path search and central node selection. The simulation results show that the time complexity of CS_DCMCMR algorithm is $O(m \log n)$, compared with CSDVC and CCLDA algorithms, the use of this algorithm can largely reduce the total cost of the final multicast tree in the case of meeting delay and delay variation constraints without the increase of complexity.

Key words: delay; delay-variation; multicast tree; minimum cost