

计算机网络

组播路由

遗传算法

延迟敏感

计算机科学2000Vol 27No. 11

基于延迟敏感的组播路由遗传算法的研究^{*}

The Study of Multicast Routing Genetic Algorithm Based on Delay Sensitive

25-28

石 坚 邹 玲 董天临 赵尔墩

(华中理工大学电子与信息工程系 武汉430074)

TP393

Abstract Based on the establishment for multicast routing network model and its mathematic description, a genetic algorithm for multicast routing which is suitable for delay and delay jitter applications is presented. The computer simulation shows that its search speed is faster and its efficiency is higher than other algorithms, furthermore, it is availability strong and robustness.

Keywords Multicast routing, Genetic algorithm, Delay, Delay jitter

1 引言

为了保证多媒体实时业务流的 QoS 传输,所采用的路由技术之一是通过组播路由(multicast routing)为实时应用寻找最佳路径。以往,人们对非受限组播路由问题研究较多,但由于它们都未考虑 QoS 的延迟标准,因此,对多媒体实时传输的要求显得束手无策。Widyono^[1]阐述了受限 Bellman-Ford(CBF)算法,它的性能较优,但其计算时间随着网络规模的增大而呈指数增长,一般在研究中作为评价解决同类问题的参考模型。Kompella 等提出的 KPP 算法^[2,3],首先采用受限斯坦纳树解决组播路由问题,并证明此类问题是 NP 完全问题。KPP 算法存在时间复杂度大或准确度低等问题。Parsa^[4]给出了得到延时限制的最小组播树启发式算法(BSMA),它可计算出近似最佳的延时受限组播路由树,但它的运算时间开销巨大且未考虑延时抖动的因素。Rouskas^[5]虽同时考虑了延时与延时抖动因素的影响,但他并没有给出优化了的斯坦纳树。

本文重点研究如何求解基于延迟敏感(即满足延迟与延时抖动要求)的最小费用组播路由树问题,称之为多受限斯坦纳最小树 MCSMT(Multiple Constrained Minimum Steiner Tree)。文中首先建立了基于延迟敏感的组播路由算法网络模型,然后提出了一种基于遗传算法的求解满足延迟与延时抖动要求的路由算法,最后给出了应用实例并对其性能进行了分析与评价。

2 基于延迟敏感的组播路由算法网络模型

包交换的计算机网络可表示为无向赋权图 $G =$

(V, E) , 其中 $V = \{v_1, v_2, \dots, v_n\}$ 为网络中所有交换节点的集合; $E = \{e_1, e_2, \dots, e_m\}$ 为任意两相邻节点间通信链路的集合。设 G 中每条边 $e = [v_i, v_j]$ 均有两个正实数加权值 (D_e, C_e) , D_e 定义为边 $e \in E$ 的信息传送延时(包括排队延时、传播延时和交换延时等); C_e 定义为 $e \in E$ 的费用,其值与链路 e 的资源使用情况有关,本文采用链路利用率的函数为该边的费用,以便评价算法管理网络资源的能力,不失一般性,假定网络是对称的且任意两节点间的链路为一。

在组播应用中,信息从发送信息的节点 $u(u \in V)$ (以下称源节点)出发,经组播路由树 $T = (V_T, E_T)$ 到达各接收信息节点 v (以下称目的节点,其集合为组播成员集合 $M \subseteq V - \{u\}$), 其中 $V_T \subseteq V, E_T \subseteq E, v \in M$ 。考虑到延时与延时抖动对路由选择的影响,本文给出如下定义:

定义1 设 $P_T(u, v)$ 为图 G 中从源节点 u 到目的节点 v 的路径,则:

• $P_T(u, v)$ 上的延时为:

$$D(P_T(u, v)) = \sum_{e \in P_T(u, v)} D(e)$$

延时的最大值为 Δ 。

• 延时抖动为从源节点 u 到各目的节点 v 的两条路径延时之差:

$$|\sum_{e \in P_T(u, v)} D(e) - \sum_{e \in P_T(u, v')} D(e)|, \quad \forall v, v' \in M$$

延时抖动的最大值为 τ 。该参量定义了目的节点的同步要求。

• $P_T(u, v)$ 上的费用为:

$$C(P_T(u, v)) = \sum_{e \in P_T(u, v)} C(e)$$

* 国家“九五”重点科技攻关项目(96-743-01-04-02)

通过构造图 G 的斯坦纳最小树使其满足多受限斯坦纳最小树问题,可归结为寻求 G 的子图 $T=(V_T, E_T)$,使得在任意一目的节点与源节点之间存在通路,且总费用(各边费用总和)最小,用公式可表示为:

$$\min(\sum_{e \in E_T} C(e)) \quad (1)$$

$$\text{约束条件: } \sum_{e \in P_T(u,v)} D(e) \leq \Delta \quad \forall v \in M \quad (2)$$

$$|\sum_{e \in P_T(u,v)} D(e) - \sum_{e \in P_T(u,v')} D(e)| \leq \tau, \quad \forall v, v' \in M \quad (3)$$

其中式(1)为费用限制,式(2)为延时限制,式(3)为延时抖动限制。此类问题是 NP-完全问题^[3-7],可采用多受限斯坦纳最小树(MCMST)算法解决。具体如下:

设已知源节点获悉整个网络的拓扑结构,则采用 Dijkstra 第 k 最短路径算法^[8]可计算出每对节点 (u, v) (其中 $v_i \in M$) 满足式(2)的路径集合 Ω ,然后从 Ω 中选取满足式(3)的有效路径集合 $\Psi_v = \{\Psi_{v_1}, \dots, \Psi_{v_n}\}$,作为下一步遗传算法的备选路径,通过遗传算法模型求解斯坦纳最小树。

3 组播路由问题的遗传算法模型

3.1 问题空间编码

采用遗传算法(GA)就必须先对问题空间进行编码,建立问题空间和 GA 空间的对应关系。由有效集合 Ψ_{v_i} 各路径组合求解斯坦纳最小树,可采用下述组合原则:

(1)对每一节点集合 Ψ_{v_i} ,最多只能从 $n_i = |\Psi_{v_i}|$ 条备选路径中选择一条;

(2)保证在每一集合 Ψ_{v_i} 中,尽可能选中一条路径,即各节点请求尽可能被接纳;

(3)当两条或多条路由通过同一链路时,这个边的费用只计算一次,即计算 $\sum_{e \in E_T} C(e)$ 时,若有重复的边,则只计算一次;

(4)按式(1)要求,求出最优斯坦纳树。

针对本节问题采用非负整数多维编码方式,既能排除后面遗传操作产生致死基因(无解),又使问题的目标函数容易转换成适应度函数。

假设网络中共有 n 个(其中 $n = |M|$)节点有组播需求,每个节点 $v_i \in M$ 对应一个有效集合 Ψ_{v_i} ,每一个集合 Ψ_{v_i} 有 n_i 条备选路径,对每个 Ψ_{v_i} 的路径按顺序分别进行非负整数编码扩充,设 Ψ_{v_i} 的路径编码为 $\{0, 1, 2, \dots, n_{v_i}\}$, Ψ_{v_j} 的路径编码为 $\{0, 1, 2, \dots, n_{v_j}\}$, $n_{v_1}, n_{v_2}, \dots, n_{v_n} \in Z^+$ (Z^+ 为正整数集),则从各 Ψ_{v_i} 中选中的路径组合可表示为多维非负整数编码的个体,其编码

方式为: $(a_{v_1}, a_{v_2}, \dots, a_{v_n})$, 其中,码长为 n , 各码取值:

$$a_{v_i} \in [0, n_{v_i}], a_{v_2} \in [0, n_{v_2}], \dots, a_{v_n} \in [0, n_{v_n}];$$

$$a_{v_i} = \begin{cases} 0 & \text{代表 } \Psi_{v_i} \text{ 的备选路径都未选} \\ a_{v_i} & \text{代表选中 } \Psi_{v_i} \text{ 的某条备选路径} \end{cases}$$

3.2 适应度函数的定义

为解决上述优化问题,首先需将目标函数映射成适应度函数,确定一个非负适应度函数表达式,它是遗传操作的依据,适应度函数设计直接影响到遗传算法的性能。将式(1)定为目标函数,即: $g(x) = \min(\sum_{e \in E_T} C(e))$,即求代价的最小值。

考虑到上节组合求解斯坦纳最小树的原则(1), (2)的限制,即 $a_1, a_2, \dots, a_n$ 不能同时为0且斯坦纳树尽量连接各组播成员,采用罚函数法,对组合原则(1)和(2)的限制条件进行转换,则可获得 MCMST 问题的适应度函数定义。

定义2 MCMST 问题的适应度函数

$$f(x) = \frac{n'}{n} [\sum_{i \in E} C(i) - \sum_{i \in E_T} C(i)] \quad (4)$$

其中, n 为组播节点数, n' 为斯坦纳树 E_T 中所包含的组播节点数, E 为赋权图的各边。

3.3 遗传操作

遗传操作采用赌轮选择、一点交叉和自适应一点变异,交叉概率为 P_c ,自适应变异概率为: $P_m = C' \frac{f_{max} - f}{f_{max} - f_{min}}$,式中 f_{max} 与 f_{min} 分别为本次群体中的最大适应度值和最小适应度值, C' 为一常数,一般可选 $C' = 0.003$,以便 $0 \leq P_m \leq C'$ ^[9]。

4 实验分析

4.1 应用实例

图1为实验网络模型,其中网络节点数 $N = |V| = 15$,源节点为 U ,组播集合 $M = \{B, S, G, L, W, J, V\}$, $n = |M| = 7$ 各边编号如图1所示,各边的费用如表1。

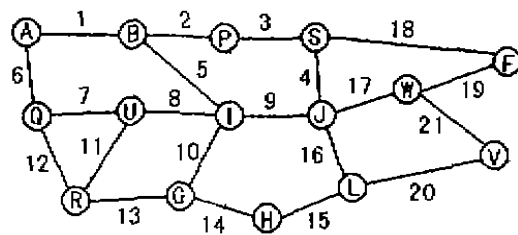


图1 实验采用的网络模型

表1 三种情况下各边的费用表

边的编号	1	2	3	4	5	6	7	8	9	10	11
费用(a)	0.2	0.33	0.2	0.35	0.43	0.3	0.32	0.5	0.28	0.22	0.5
费用(b)	0.5	0.2	0.3	0.5	0.2	0.2	0.5	0.2	0.2	0.4	0.25
费用(c)	0.22	0.2	0.32	0.62	0.2	0.25	0.2	0.21	0.25	0.15	0.26
边的编号	12	13	14	15	16	17	18	18	20	21	
费用(a)	0.2	0.12	0.42	0.35	0.22	0.2	0.32	0.2	0.2	0.2	
费用(b)	0.64	0.82	0.13	0.36	0.45	0.2	0.3	0.2	0.2	0.12	
费用(c)	0.2	0.12	0.72	0.2	0.2	0.5	0.2	0.2	0.5	0.2	

设从源节点U到各个目的节点的备选路径均已求得,且满足条件式(2),式(3)。这里分别给出三种情况下用MCMST算法的优化结果,各树的组成分别为: {4,5,8,9,10,16,17,20}, {2,3,5,8,9,10,16,17,20}和{2,3,5,8,9,10,16,18,19,21},在这三种情况下,当遗传群体大小为120,代数不超过70代时,均得到斯坦纳最小树,且其开销值分别为:2.29、2.25和2.13。

4.2 性能分析

我们对算法进行了计算机仿真,分别从组播路由树的费用和算法的运行时间两个方面,对MCMST算法的性质进行了测试。实验采用Waxman网络拓扑生成器^[10],随机产生一些具有实际网络特性的拓扑模型;首先随机产生一系列节点,按照一定的概率生成机制建立链路的连接,节点的平均度在2~4之间变化;链路费用定义为当前线路所占用的带宽,其变化范围为(0,1)间的实数。组播通信规模(组播节点数 n 与网络总节点数 N 的比)从10%到40%之间变化。图2给出了MCMST算法的运行时间在各种情况下随网络规模的变化情况。

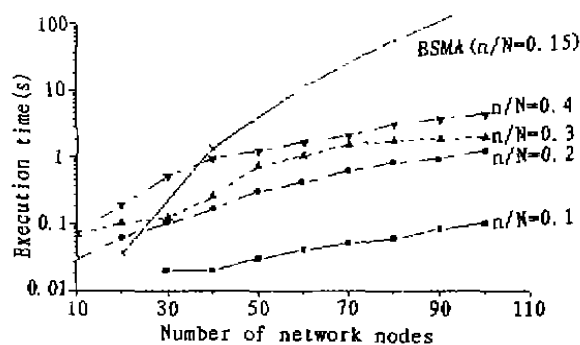


图2 MCMST算法的运行时间在各种组播通信规模情况下随网络规模的变化情况

由图2可以看出,当 $n/N=0.1$ 时,遗传算法寻优的计算时间最短,这是由于组播节点较少,节点的度也较小,使得备选路径不多,算法能很快找到最优解树;随

着网络节点和组播节点的增加,备选路径集合元素急剧膨胀,算法求得最优解的时间相对增长;尽管如此,在网络节点规模为100,组播节点数为40时,MCMST算法的计算时间仍不超过10秒,而且MCMST与BSMA算法^[4]比较,在大型网络中,BSMA计算时间明显高于MCMST算法的求解时间,且增长幅度随着网络节点的增多而急剧增长。这说明在大规模网络的组播路由树求解中,从运算时间上考虑本算法比BSMA优越很多。

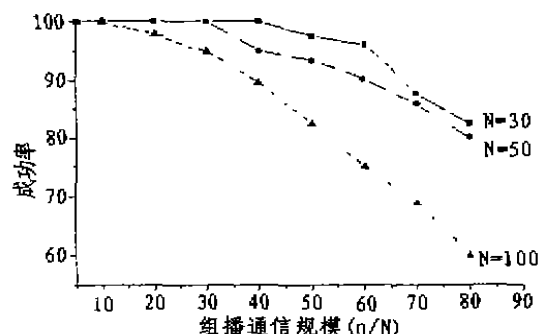


图3 算法求解成功率随组播通信规模变化的情况

通过图3和表2可对MCMST算法的稳定性作如下评价:当组播通信规模不大于30%时,MCMST算法有较好的稳定性,可适用于各种规模的网络中,特别是当组播通信规模不超过20%时,算法的求解成功率很高,其稳定性受网络节点规模的影响较小;但算法的稳定性受组播成员数的影响较大,当组播通信规模增大时,MCMST算法求解的成功率降低,尽管如此,在网络节点为100,组播成员数为80时,仍有大于60%的成功率,说明算法是有效的。图3表示了当网络总节点数分别为 $N=30$ 、 $N=50$ 及 $N=100$ 时,MCMST算法求解的成功率随组播通信规模变化的情况。由于在实际网络中组播通信规模一般小于40%,因此,MCMST算法能够满足实际网络应用的需求。

表2 算法求解成功率随网络节点变化的情况($n/N=20\%$)

网络总节点数	10	20	30	40	50	60	70	80	90	100
成功率	100%	100%	100%	100%	100%	100%	100%	100%	95%	93.3%

结论 基于延迟敏感的最小费用问题是典型的优化问题。由于此类问题属于 NP 完全问题,目前多数采用非确定性的启发式方法来求解。其中最具代表性的是 Parsa 提出的延时限制的只小组播路由树启发式算法(BSMA),它可计算出近似最佳的延时限制组播路由树,但它的运算时间开销巨大且未考虑延时抖动的因素。本文在 CMST 算法的基础上,分析了在高速网络中具有端-端延时和延时抖动的组播路由算法问题,提出了一种适合延迟与延迟抖动要求的组播路由遗传算法模型,并进行了实验分析,结果表明,本文提出的方法能求解适合延迟与延迟抖动要求的最小组播路由树,而且在相对大型的的网络中,本文提出的 MCMST 算法与 BSMA 相比,具有搜索速度快、效率高,计算性能较稳定的特点,避免了采用启发式搜索技术对问题知识的要求,具有较强的通用性和鲁棒性,能够满足多媒体应用对时延和时延抖动的要求并可适用于大规模的网络中。

参考文献

- 1 Widyono R. The Design and Evaluation of Routing Algo-

- gorithms for Real-Time Channels [Tech. Rep. ICSI TR-94-024]. University of California at Berkeley, International Computer Science Institute, June 1994
- 2 Kompella V, et al. Multicast Routing for Multimedia Communication. IEEE/ACM Transactions on Networking, 1993, 1(3): 286~292
- 3 Kompella V, et al. Multicasting for Multimedia Applications. In: Proceedings of IEEE INFOCOM'92 1992 2078~2085
- 4 Parsa M, et al. An Iterative Algorithm for Delay-Constrained Minimum-Cost Multicasting. IEEE/ACM Transactions on Networking, 1996, 6(4): 461~474
- 5 Rouskas G N, Baldine I. Multicast routing with end-to-end delay and delay variation constraints. IEEE Journal on Selected Areas in Comm, 1997, 15(3): 346~356
- 6 Garey M R, Johnson D S. Computers and Intractability: A Guide to the Theory of NP-Completeness. New York: W H Freeman and CO., 1979
- 7 Zhu Q, et al. Garcia-Luna-Aceves. A source-based algorithm for near-optimum delay-constrained multicasting. In: Proceedings of IEEE Infocom'95, March 1995
- 8 Lawler E. Combinatorial Optimization: Networks and Matroids. Holt, Rinehart and Winston, 1976
- 9 陈国良,王熙法,庄镇泉,等.遗传算法及其应用.人民邮电出版社,1996
- 10 Waxman B M. Routing of Multipoint Connections. IEEE Journal of Selected Areas in Communications, 1988, 6(9): 1617~1622

(上接第16页)

```

...
Dim ISA- WHILE1 As Integer
ISA- WHILE1=1
Call CountRP(ISA-local,2)
While Cond(x>0 And y>0,-1)
ISA- WHILE1=0
Call CountRP(ISA-local,3)
Dim ISA- IF2 As Integer
ISA- IF2=1
Call CountRP(ISA-local,4)
If Cond(x>y,0) Then
ISA- IF2=0
Call CountRP(ISA-local,5)
x=x-y
Call CountRP(ISA-local,7)
y=y-x
End If
Call CountRP(ISA-local,8)
Wend
...

```

'Count WHILE-NODE
'Has been in WHILE-BODY
'Count WHILE-BODY

'Count IF-NODE
'Count IF BODY
'Count ELSE-BODY
'Count IF-CONJ

结束语 产生式的设计与源语言是密切相关的,但是,完成测试语义动作的子程序与源语言并没有多大的联系。也就是说,只要修改产生式,使其面向另外的编程语言,我们可以方便地实现我们的测试语义,使我们开发的软件测试工具平滑地向其他编程语言过渡,如 Delphi。

参考文献

- 1 陈火旺,钱家骅,孙永强.程序设计语言编译原理.北京:国

- 防工业出版社,1984
- 2 郑人杰.计算机软件测试技术.北京:清华大学出版社,1992
- 3 孙鑫.软件测试工具的研究和实现.[硕士学位论文]杭州:浙江大学,1999
- 4 Donnelly C, Stallman R. Eison Information. Cambridge, Free Software Foundation, 1992
- 5 Better B. Software Testing Techniques. New York: Van Nostrand Reinhold Company, 1990