

Bellman-Ford算法的改进研究

陈华容¹, 张崇富²

(1. 西南科技大学计算机学院 四川 绵阳 621010; 2. 电子科技大学通信与信息工程学院 成都 610054)

【摘要】针对传统距离矢量路由(Bellman-Ford)算法, 将多约束路由问题简化, 并改进路由问题的限制条件, 提出了改进的Bellman-Ford算法, 应用于基于集中式控制路由协议网络模型。仿真结果表明, 采用改进算法的网络路由计算时间随网络节点数有明显的改善, 且网络阻塞率随路由请求数增加而下降。验证了改进算法比传统算法更加有效。

关键词 距离矢量路由; 多约束路由; 集中控制路由协议; 时延-费用; 阻塞率
中图分类号 TP393 **文献标识码** A

Study on Amelioration of Bellman-Ford Algorithm

CHEN Hua-rong¹, ZHANG Chong-fu²

(1. College of Computer Science and Engineering, SWUST Sichuan Mianyang 621010;
2. School of Communication and Information Engineering, UEST of China Chengdu 610054)

Abstract Amelioration of Bellman-Ford algorithm is deduced in the letter based on traditional Bellman-Ford algorithm and applied to CCRP networks. The simulation results demonstrate that calculating time of networks adopted amelioration of Bellman-Ford algorithm is better than that of adopted Bellman-Ford algorithm with node number increasing and block ratio of networks is declining with request number increasing.

Key words Bellman-Ford; multi-constrained problem; central control routing protocol; block ratio

路由选择方法可归属于距离矢量路由算法与链路状态路由算法^[1]。距离矢量路由算法也称Bellman-Ford算法, 该算法有路由协议算法简单, 路由器配置简易等优点; 缺点是网络中的过时路由不能及时被删除造成网络不稳定, 大规模网络需要很长的收敛时间, 最大跳数限制网络规模, 数据库没有变化时需要周期发送距离向量表等。本文针对上述缺点改进了Bellman-Ford算法, 并在集中控制路由协议(Central Control Routing Protocol, CCRP)网络中得到了应用。

1 改进的Bellman-Ford算法

时延-费用约束路由和多约束路由问题(Multi-Constrained Problem, MCP)都属NP-complete问题^[1], 要解决这类问题首先要简化NP-complete问题。文献[1-5]对解决这类问题有所研究, 并取得了一定的进展, 尤其是文献[4]提出的MCP定义和定理, 使多约束路由问题在一定程度上得到了简化。

定义 1 对于多约束路由问题, 给定一个有向图 $G(V, E)$, 一个源节点 s 和一个目标节点 t , 两个度量函数 $w_1: E \rightarrow R_0^+$ 和 $w_2: E \rightarrow R_0^+$, 两个常数 $c_1 \in R_0^+$ 和 $c_2 \in R_0^+$, MCP即 $MCP(G, s, t, w_1, w_2, c_1, c_2)$ 是找到一条从 s 到 t 的满足 $w_1(p) \leq c_1$ 和 $w_2(p) \leq c_2$ 的路径 p , 其中 w_1 和 w_2 都是加性度量; R_0^+ 代表非负实数集; I 代表非负整数集。

定义 2 建立一个新的度量函数 $w_2': E \rightarrow I$, 有:

$$w_2'(u, v) = \left\lceil \frac{w_2(u, v)x}{c_2} \right\rceil \quad (1)$$

式中 x 是一个给定的正整数。通过函数 w_2' 可以把 $MCP(G, s, t, w_1, w_2, c_1, c_2)$ 归结为一个比较简单的 $MCP(G, s, t, w_1, w_2', c_1, c_2)$ 。

定理 1 $MCP(G, s, t, w_1, w_2', c_1, c_2)$ 的解必然是 $MCP(G, s, t, w_1, w_2, c_1, c_2)$ 的解。

定理 2 假设路径 p 是 $MCP(G, s, t, w_1, w_2, c_1, c_2)$ 的解, 而 l 是路径 p 的长度, 如果 $w_2(p) \leq (1 - \frac{l-1}{x})c_2$, 则路径 p 也是问题 $MCP(G, s, t, w_1, w_2', c_1, c_2)$ 的一个解。

定理 3 假设 P 是问题 $MCP(G, s, t, w_1, w_2, c_1, c_2)$ 的解集, 则当 $P \neq \emptyset$, 且 $\exists p \in P, w_2(p) \leq (1 - \frac{l-1}{x})c_2$ 成立时, $MCP(G, s, t, w_1, w_2, c_1, c_2)$ 的解可以通过简化 $MCP(G, s, t, w_1, w_2', c_1, c_2)$ 求出。

根据以上结论, 可以把NP-complete的时延-费用路由问题简化, 然后使用改进的Bellman-Ford算法在多项式时间内解决。

时延-费用路由与带宽/跳数路由的定义相类似^[5], 网络是一个有向图 $G=(V, E)$, 其中 V 是节点集, $v \in V$; E 是连接节点的边集, $e \in E$ 且 $e=(a, b)$, $(a, b) \in V$; D_i 和 C_i 是边集 E 的权, $i=1, 2, \dots, |E|$ 。时延/费用路由问题就是要在 G 中找到从源节点 s 到目标节点 d 的路径 $r=\{a_1, a_2, \dots, a_m\}$, $a_1=s, a_m=t, a_i \in V$, 且限制条件为:

$$\sum_1^{m-1} D_{(a_i, a_{i+1})} \leq D_{\text{elay}} \quad (2)$$

$$\sum_1^{m-1} C_{(a_i, a_{i+1})} \leq C_{\text{ost}} \quad (3)$$

式中 D_{elay} 和 C_{ost} 分别是给定的时延要求和费用要求。

由式(1)可看出, 解决问题的关键是如何确定正整数参数 x 来简化问题。选择费用参数作为简化的对象, 需要确定一个新的费用参数 C_x 。实验结果表明, C_x 越大, 实验结果越好, 但路由计算的开销也越大。选定 C_x 后, 路由问题限制条件中式(3)变为 $\sum_1^{m-1} C_{(a_i, a_{i+1})} \leq (1 - \frac{l-1}{C_x})C_{\text{ost}}$, 而时延-费用路由问题就简化为在 G 中找到从源

节点 s 到目标节点 d 的路径 $r = \{a_1, a_2, \dots, a_m\}$, $a_1=s, a_m=t, a_i \in V$, 则限制条件中的式(3)变为 $\sum_1^{m-1} C'_{(a_i, a_{i+1})} \leq C_x$ 和

$C'_{(a_i, a_{i+1})} = \left[\frac{C_{(a_i, a_{i+1})} C_x}{C_{\text{ost}}} \right]$, 式中数组 e 初始化为最大值。算法结束时, 如果满足限制条件式(2), 则解路径存在,

且解存放在数组 r 中。 $r[v, k]$ 存放了节点 v 在路径上的前一个节点, 称之为节点 v 的前缀。因此, 只要从数组 r 中的 d 目标节点出发, 不停地找前缀直到到达源节点 s , 就可得到解路径。设 N 和 M 分别是节点数和边数, 算法的时间复杂性是 $O(C_x NE)$, 算法中for循环最多要执行 $(C_x+1)(N-1)M$ 次, 算法的空间复杂性是 $O(C_x N)$ 。

2 计算机仿真与讨论

2.1 仿真网络模型

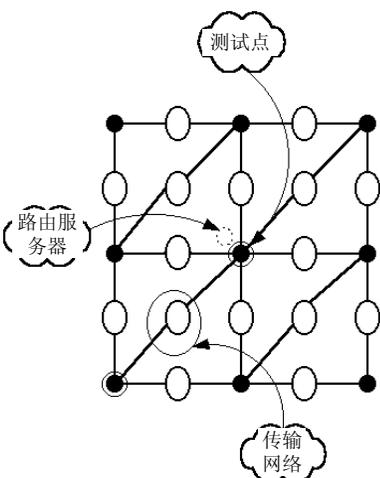


图1 网状网络模型

集中控制路由协议(Central Control Routing Protocol, CCRP)采用了人工、网状拓扑结构, 如图1所示。图中的黑色圆点表示节点, 节点具有路由器的一切功能, 包括发送/接收链路状态公告(Link State Advertisement, LSA)、转发网络、提出路由请求、转送数据等; 椭圆圈表示连接节点的传输网络, 传输网络就像传输链路一样具有一定的度量值, 如带宽、时延、费用等。考虑到路由服务器与其他节点之间只有状态信息、路由请求信息和完整路径信息的交换, 数据量不大。假定当网络规模不大, 网络节点数 ≤ 100 时, 路由服务器与网络中的所有节点都有链路相连, 且链路的带宽统一为64 Kb/s; 当网络规模较大, 网络节点数 > 100 时, 路由服务器位于网络的中心处, 即图1的虚线椭圆圈处。它与离它最近的100个有直接连接的链路节点与网络中的其他节点通过网络中的现存链路进行连接, 这100个节点是以网络中心节点为中心的一个 10×10 网状拓扑结构包含的节点, 链路带宽亦统一为64 Kb/s。从网络建设的安全性和健壮性来考虑, 为一定数量的节点之间建立专门功能的传送线路是可以接受的。

2.2 路由计算的时间开销

图2是CCRP协议下, 测试点分别采用Bellman-Ford算法和改进的Bellman-Ford算法进行带宽-跳数类型的QoS路由计算时间的比较, 实验比较过程中采用了相同的测试数据。路由计算的概念是指完成一次路由请求

所耗费的时间, 包括:

(1) 在CCRP协议中, 节点收到一个路由请求, 查询本地链路状态信息数据库, 调用路由算法进行计算, 找到下一个节点的全过程的时间;

(2) 在CCRP协议中, 节点收到一个路由请求, 向路由服务器发出计算路由的请求LSA, 等待路由服务器调用算法计算路由, 收到路由服务器返回路径的全过程的时间。

从图2可以看出, CCRP的路由计算时间随着网络规模的扩大呈线性增长趋势。考虑到前述的计算时间, 尤其是路由服务器调用算法进行路由计算所需时间随网络规模的变化而增长, 可以认为服务器的计算时间是近似的线性增长, 但并不是说算法的计算时间线性增长, 因为服务器的计算时间包含查询链路状态信息数据库的时间等其他部分。当网络规模增长到350个节点时, 平均的计算时间仍能保持在大约16 s, 即节点可以在16 s内得到QoS路由, 这个时间是可以接受的。即使从服务器和节点比较的结果来看, 使用路由服务器计算一条完整路径所用的时间也不会比用分布式路由策略由各个节点各自计算路径中的下一个节点所需的累计时间长。相反, 路由服务器的计算时间比各个节点的累计计算时间短得多。

2.3 路由协议的性能估计

在一个10×10的网状拓扑结构中, 分别采用Bellman-Ford算法和改进的Bellman-Ford算法, 使用相同的测试数据, 即所有的路由请求都是QoS路由请求, 比较网络的阻塞率, 如图3所示。图中的结果是30次实验得到的平均阻塞率。所有路由请求发生在250 s内, 凡是在该时间段内未能完成的路由请求都被认为是失败的路由请求。由前面的分析可以看出CCRP路由失败既可能是节点转送数据失败, 也可能是路由算法无法找到可行路径。后者既有确实没有可行路径的可能, 也有可行路径存在而路由算法无法找到的可能。从图3可以看出, 采用改进的Bellman-Ford算法使网络的阻塞率比采用Bellman-Ford算法有明显的下降, 有力地证明了本文所提出的改进算法处理QoS路由问题比传统路由算法更加有效。

3 结论

本文根据集中控制的思想, 研究了在模拟软件上实现一个集中式控制的路由协议CCRP, 并改进了QoS路由问题的Bellman-Ford算法。CCRP协议的实现采用了硬件厂商目前所广泛支持的一种Java语言开发接口, 能够有效地解决在不同的软硬件环境中移植的问题。此外, Java语言的开放性和灵活性有利于开发的进行。

参考文献

- [1] Wang Z, Crowcroft J. Quality-of-service routing for supporting multimedia applications[J]. IEEE Journal on Selected Areas in Communications, 1996, 14(7): 1 228-1 234
- [2] Andrew S T. Computer network (third edition)[M]. 北京: 清华大学出版社, 1996.
- [3] Garey M, Johnson D. Computers and intractability: a guide to the theory of NP-completeness[M]. New York: W. H. Freeman and Co., 1979.
- [4] Chen S, Nahrstedt K. On finding multi-constrained paths[R]. Tech. Report UIUCDCS-R-97-2026, Dept. of Computer Science, UIUC, 1997.
- [5] Cormen T H, Leiserson C E, Rivest R L. Introduction to algorithms[M]. Cambridge: MIT Press and McGraw-Hill Book Co., 1990
- [6] Moy J. OSPF version 2[S]. RFC2328, 1998.

编辑 熊思亮

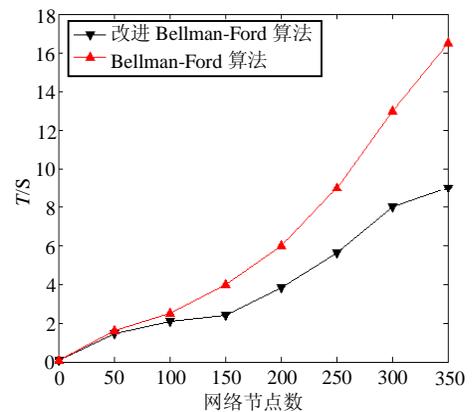


图2 网络节点数 N 与计算时间 T 关系曲线

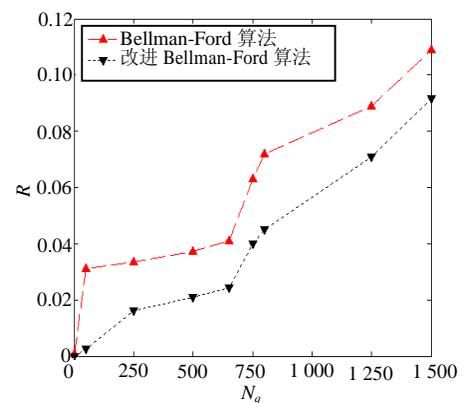


图3 路由请求数 N_q 与网络阻塞率 R 关系曲线