Performance Analysis of QoS Routing with Network Dependant Delay Cost

Hahnearl Jeon*, Sung-Dae Kim*, Young-Joon Kim**, Hyung-Taek Kim** and Jai-Yong Lee*
*Department of Electronic Engineering, Yonsei University
** Mobile Telecommunication Network Group, LG Electronics
bird@nasla.yonsei.ac.kr

Abstract

This paper investigates the problem of path calcul ation of multiple metric routing. Toda 's Internet routing is based on a single metric path selecting algorithm. Single metric path is a best effort service that can only support path for one requirement. In order to support quality-of-service(QoS) call, path is subjec to multiple constraints on routing metric. In many cases, the path finding problem is NP-complete. This paper proposes widest-least cost routing algorithm that uses cost metric that is based on delay metric and inf uenced by resource bandwidth metric and network state. This algorithm is a multiple metric routing algorithm that has traffic distribution ability to serve traffic engine ring. Our goal is to select shortest path when network link is not loaded, and perform traffic engineering to move traffic to other path when network load is heavy. We have studied the performance through simulation and compared it against other routing algorithms.

1. Introduction

Traditional communication network architectures were designed to support users with simple quality-of-service(QoS). Routing in today's Internet is focused o connectivity has the capability to serve only th "best-effort" service. Multimedia applications such as digital video and audio has increased demand for wide spectrum of network service and broadband integrated service networks are expected to support multiple and diverse QoS requirements.

Current Internet routing protocols, e.g. OSPF, RIP use "shortest path routing", i.e. routing that is optimized for a single arbitrary metric[2]. QoS routing is one of the tools available to accommodate flows which expect certain QoS guarantees from the network. It enables the network to identify the sufficient resource path through the network. QoS routing is seen as a missing piece in Internet and an important factor in order serve various qual ities required from network. QoS routing has been receiving attention in order to overcome IP limitation to serve various qualities of connectio rather then best-effort service. In the past several years, QoS routing has been subject of several studies and proposals ([3]-[6],[10],[12 -[15],[17]-[19],[21],[23], and references therein).

In order to extend current routing, routing algorithms calculate path under multiple constraint s. According t current works, calculating most suitable path using more than one metric is known to be u nfeasible[5]. Multi-criteria shortest path problem is not well defined. Ther e-fore, it is important to select most reliable path among many path and to optimize it. If the given metric is a d y-namic metric that changes according to network status, accurate value is essential to select the path. However, network size and topology aggregation and tradeoff with network traffic and routing update cause inaccurate routing information. [10] deals with the impact of inaccuracy on the ability of path-selection. Even though, given the QoS request of a flow or an aggregation of flows, we have t select a path that will provide best ser vice to the connection.

When network load is light, best-effort service and integrated service differ little. Performance degrades when traffic load is heavy. Network congestion can be caused by lack of network resources or uneven distrib ution of traffic Traffic engineering is a process of arranging how traffic flows through the network so that congestion caused by uneven network utilization can be avoided. Changing path to lower cost [7], [20] or sending traffic along multiple path [13], setting cost acc ording to ba dwidth availability [22] is a way to realize traffic engine ring. In this paper we focus on path selecting algorithm subject to multiple metrics. This algorithm also has tra ffic distributing ability that enables traffic engineering.

The rest of the paper is structured as follows. In Section II, we present a widest-least cost multi-metric routing algorithm. In Section III, performance of our algorithm will be compared with other routing algorithms. In Section IV we summarize our findings.

2. Widest-Least Cost Routing Algorithm

In this section, we address the limitation of computing paths with more than one metric and present our algorithm to improve this situation.

When routing in networks, routing metrics are the repr e-sentation of network. Therefore, while computing path between nodes, use of a metric will decide the resultin path. Path computation algorithms for a single metric, such as delay and ho -count, are well known and have been widely used in current network. Using metric more than one can certainly model network more accurately, and will provide a path that supports various QoS requirement in connections. Several path selection algorithms have been proposed. They include the widest-shortest path[6], the shortest-widest path[5], and bandwidth-delay-constraint path[5].

Even though widest-shortest path and shortest-widest path routing computes path with more than one metric, resulting path depends mostly on first metric. The reason is that, during path computation, second metric is used only when first metric could not determine the best path. What we use for first metric will decide most of the performance of routing algorithm and network efficiency. Therefore, shortest-widest algorithm focuses on load balancing while widest-shortest algorithm focuses on shortest path. Ho w-ever, shortest-widest algorithm's tendency to focus on wide bandwidth link may select long path even though there is a short path with sufficient bandwidth. In previous works, widest-shortest path has shown better performance when network load is heavy[10] and performed bette when using in routing server[20] and topology aggre a-tion[24].

The problem for widest-shortest path is that if it selects only one path according to the first metric, no other path can be selected. Then, this selected path is the same path calculated with one routing metric which is dijkstra alg o-rithm. The path will not be able to redirect path when net-work cannot accept another path according to first metric, and the path will be fixed what ver the network state may be. Bandwidth-delay-constraint path performs well on call acceptance because the algorithm eliminates links that does not meet the call requirement, but it also has weak-ness in load balancing.

Key idea for our algorithm is to use delay and band-width metric in network to create cost metric which is c l-culated from both of these metrics. The cost metric i based on delay metric and influenced by bandwidth metric. This cost metric is not always used. We will use this metric when network load increases over predefined threshold rate where traffic should be moved to other links. The value of cost is same as delay when network load is under threshold rate, but when network load is over threshol rate, we use following equation to calculate the value of bandwidth dependant cost metric *cost*(*l*) of link *l*.

$$cost(l) = d_l \times \frac{T}{\sigma \times \left(1.01 - \frac{b_l}{c_l}\right)}$$

where d_l is delay of the link l, T is the threshold band-width rate, σ is the bias value, b_l is current consumed bandwidth of link l, c_l is capacity bandwidth of link l. Bias value σ is used to control the total cost value. The equation has the form of threshold rate over bandwidth available rate. If σ value is small, it will increase the cost value. It i used as a control factor of the amount of changes in cost value. Therefore, the value of σ will change the perform-ance of the algorithm. However, bias value cannot increase too much as to form the denominator of the equation to be greater the threshold value. This would decrease cost to be smaller than delay and would bring confusion in path cal-culation.

We now present widest-least cost algorithm. It is the path finding algorithm from a source to all destinations in the network. Consider a directed graph $G = (N,A)$ with nodes numbered 1, …, N. Each arc $(i,j) \in A$ has a cost or "length" a_{ij} , current consume bandwidt b_{ij}, and ba d-width capacity c_{ij} associated with it. Given any d irected forward path $(i, j, k, \ldots, l, m)$, the width of the path width(p) is defined as the bottleneck bandwidth of the path, i.e., width(p) = $\min\{b_{ij}, b_{jk}, \ldots, b_{lm}\}$, and the length of the path length(p) is defined as the sum of arc length, i.e., length(p) = $a_{ij} + a_{jk} + \ldots + a_{lm}$. The path is said to be short-est if it has minimum length over all f rward paths with the same origin and destination nodes. For some pat p_i from 1 to i, d_i can be interpreted for all i as the length o path p_i, i.e., d_i = length(p_i). Let V as the candidate list and B_i be the width along the path p. The widest-least cost path algorithm can be produced by adding bandwidth checking when there are multiple equal cost paths. The widest-least cost algorithm is as follows.

1) Initially, $V = \{1\}$ $d_1 = 0$, $d_i = \infty$, $B_i = \infty$, $\forall_i \neq 1$.

2) The method proceeds in iterations and terminated when V is empty.

Iteration of Widest-Least Cost

3) Remove from the candidate list V a node i such that

$$d_i = \min_{j \in V} d_j$$

4) For each outgoing arc $(i, j) \in A$,

a) if $b_{ij}/c_{ij} > T$,

$$a_{ij} := result\ \ from\ cost\ equation$$

b) if $d_j > d_i + a_{ij}$, set

$d_j := d_j + a_{ij}$

$B_j = \min\{B_i, a_{ij}\}$

c) else if $d_j = d_i + a_{ij}$,

```
    if $B_j < \min\{B_i, a_{ij}\}$
        $d_j := d_j + a_{ij}$
            $B_j = \min\{B_i, a_{ij}\}$
5) add $j$ to $V$ if it does not already belong to $V$.
        goto 2)
```

The algorithm checks every node and link by putting nodes in candidate list *V* and terminates when all the node has been checked and removed from the list. Our algorithm works like widest-shortest path algorithm when network load is under threshold rate. When network load su r-passes threshold rate, our widest-least cost algorithm starts to distribute traffic to other links. Therefore, we can view widest-shortest path algorithm as a case of widest-least cost algorithm.

3. Simulation and Performance

In this section, we present extensive simulation result to evaluate the proposed routing algorithms and compare the performance with other existing routing algorithms. The purpose of our experiments is to determine that our al go-rithm has better performance over other algorithms o network efficienc and load balancing which i ncreases call acceptance in network.

3.1 Simulation Model and Performance Measures.

In our simulation model, we have used two topologies. Fig 1 is conceptual model topology used to show the basic operation and characteristics of our algorithm. We have used Fig. 2 for performance evaluation. Fig. 2 is designed based on our national high-speed backbone network. Bandwidth for the links in Fig 2 are 155Mbps except for the link shown as bold. This link has 622Mbps for its bandwidth. Link lengths of Fig. 2 are chosen randomly, but it relates to the length in Fig. 2. Each node in Fig. 2 i composed with one backbone router connected to the backbone and two edge routers connected to backbone router and with each other. Two edge routers are connecte to two hosts that generate and receive calls.

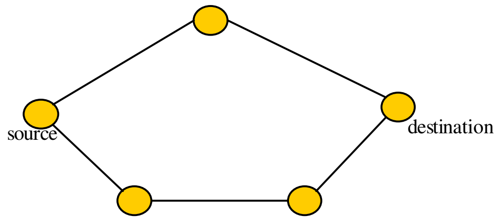

Fig 1. Conceptual topolog

Simulation program selects source and destination hosts randomly, generates call connection. We assume connection calls arrive in the network following poisson distribu-

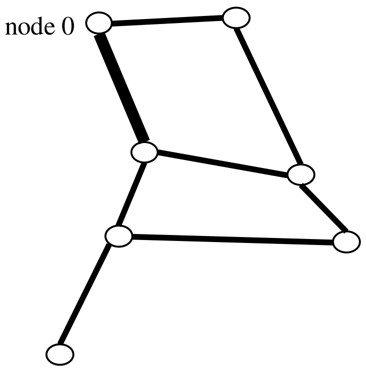

Fig. 2 Network topology of the backbone used in simulations

tion. When each call is successively connected, the duration period of call will be selected from exponential distribution value. After call has been generated, feasible path for each call is searched using routing algorithm which performance is to be checked. If the algorithm finds a path for connection, each call tries to set up path along the returned path. Call will be rejected when one of the lin along the path from source to destin ation does not fulfill the requested bandwidth or end -to-end delay exceeds requested value. Algorithms used for performance comparison are dijkstra algorithm, widest-shortest algorith and proposed widest-least cost algorithm.

For the threshold rate and bias value to be u ed in the proposed algorithm, we have chosen 0.8 for threshold rate and bias value of 3. Fig. 3 shows the resulting values to be multiplied to delay according to the bandwidth usage of the link. Bias value of 3 is chosen to make the value to start at 1 when bandwidth is near threshold and increases slowly at start and grow to infinite when link is full.

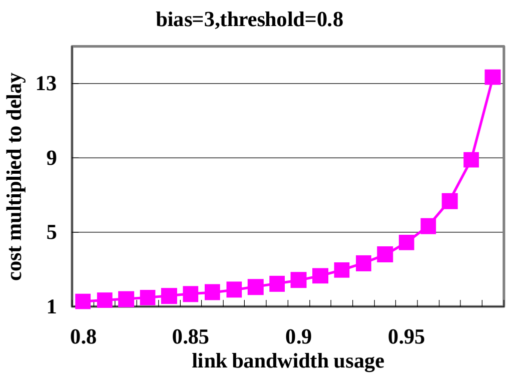

Fig.3 value to be multiplied to delay when link is over threshold

In order to compare the performance of our algorithm, we have used two performance measures: *call rejection rate*(CRR) and *bandwidth efficiency* (BWE). CRR is the number of calls rejected showing paths selected by alg o-rithm satisfies given constraints while ABWE is the average value of BWE that is bandwidth usage among total link capacity. CRR and BWE can be calculated as follows

$$\mathrm{CRR} = \frac{n\_rej}{n\_req}$$

$$\mathrm{BWE} = \frac{b_l}{c_l}$$

where *n_rej* is the number of calls rejected, and *n_req* is the number of calls requested for connection.

Our goal of path selection is to reduce CRR and choose feasible path that achieves efficient utilization of network resources

3.2. Results

We consider two cases from Fig 1. This simple simulation will show the performance of our algorithm. First case is that delay for the upper path and lower path have same value. Results in Table I show that traffic is di stributed equally on the network. Average bandwidth usage of every link is same. No calls have been rejected.

Table I. Simulation result of equal path length for topology in Fig.

	dijkstra	widest-shortest	widest-least cost
CRR	0.0	0.0	0.0
Max ABWE	53%	46%	47%
Min ABWE	52%	46%	47%

Table II. Simulation result of different path length for topology in Fig.1

	dijkstra	widest-shortest	widest-least cost
CRR	0.28	0.24	0.0
Max ABWE	84%	85%	71%
Min ABWE	0%	0%	22%

.

This situation changes when the delay for upper path is shorter than lower path. Let the delay for each link have same value of 1. Then upper path will have end-to-end delay of 2, and lower path will have e -to-end delay of 3. Results in Table II show that for dijkstra and widest-shortest algorithm, all the traffic is located at one side. In this case it is upper path. No traffic is located at other links. Therefore only part of the network is being used. On the other hand, our widest-least ost algorithm uses both upper and lower path for call connection. The results show that we have lower value of call rejection rate

Comparing values of case1 and case2 reveals that while widest-shortest routing algorithm uses two metric for path cal ulation in some cases the resulting path depend only on first metric. Therefore, traffic is crowded on one side. Our routing algorithm performs perfectly on both cases. Besides path selection, it has ability to distribute traffic.

For the topology shown in Fig 2, the simulation wa executed with all sourc hos s and destination hosts in nodes generating randomly. Requested bandwidths for the calls are 512Kbps with 100ms delay. We have changed traffic load of the simulation b varying the number o calls generated in the network per minute. The simulation has been done for ten times for each case with simulation time of 600 minutes each. The result is shown in Fig. 4. It is the call rejection rate from the host in a node. We have observed the performance through one source host. It could be seen that our algorithm has lower rejection rate than other algorithms.

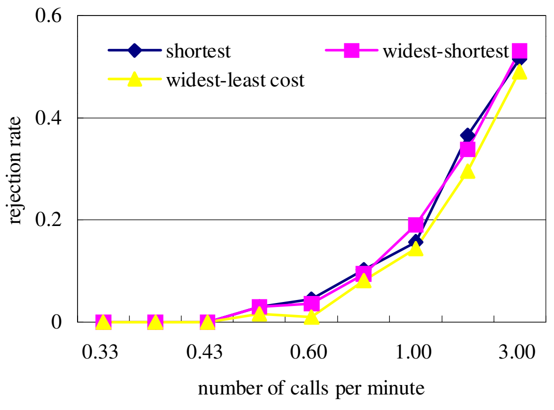

Fig 4. Call Rejection rate

Fig. 5 shows the average delay of the path selected for the a ccepted call in the simulation. Our algorithm has similar value when network load is light and larger value than other algorithms when network traffic increases. Thi is because our algorithm finds alternate path that increases the average delay value. Shortest path algorithm and widest-shortest algorithm shows similar performance. Result of the simulation shows that our algorithm checks more paths than other algorithms. Our algorithm has smallest CRR and we have no link with 0% u sage.

4. Conclusion

The performance of a new routing algorithm called widest-least cost algorithm is proposed and compared with two other routing algorithms. The algorithm is superior to other algorithms in that it can operate as widest -shortest

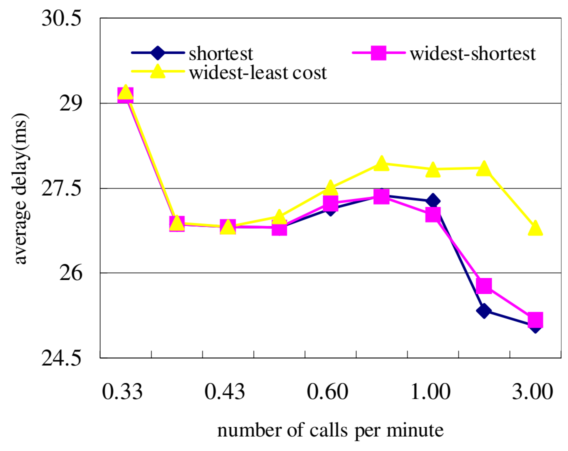

Fig 5. Average path delay of calls

algorithm at light load network and widest-least cost at heavy load networks. The ability to use other network lin according to the link usage enables the power to move traffic to less congested link result in traffic engineering which is useful to distribute load over network. This encourages reduce in call rejection. This algorithm can be used for traffic load balancing and useful in avoiding traffic congestion.

Effective threshold value and bias for the cost function is for further study. These values have important impact on network.

References

[1] D.Bertsekas and R.Gallager, Data Networks, Prentice Hall, 1992.
[2] C.Huitema, Routing in the Internet, Prentice Hall, 1999
[3] W.C.Lee, M.G. Hluchyi, and P. Humblet, "Routing Subject to Quality of Service Constraints in Integrated Communication Networks", *IEEE Networ* , pp.46-55, July/August, 1995.
[4] R.Vogel, R.Herrtwich, W.Kalfa, H.Wittig, and L.Wolf, "QoS-Based Routing of Multimedia Streams in Computer Networks", JSAC, vol 14, no.7, September 1996.
[5] Z.Wang and J.Crowcroft, "Quality-of-Service Routing fo Supporting Multimedia Applications", IEEE Journal on Selected Areas in Communications, Vol. 14, No. 7, Sep., 1996
[6] R.Guerin, A.Orda, and D.Williams. "QoS Routing Mechanisms and OSPF Extensions ", Internet draft, Nov.,1996.
[7] Zhang, Sanchez, Salkewicz, and Crawley, "Quality of Service Extensions to OSPF", Internet draft, Sep., 1997.
[8] H.Salama, D. Reeves, and Y.Viniotis, "A Distributed Algorithm for Delay-Constrained Unicast Routing", IEEE INFOCOM '97.
[9] S.Murthy, and J.Garcia-Luna-Aceves, "Loop-Free Internet Routing Using Hierarchical Routing Trees", IEEE I NFOCOM '97.
[10] Q.Ma, and P.Steenkiste, "On path Selection for Traffic with Bandwidth Guarantees", IEEE International conference on Network Protocols, Oct., 1997
[11] R. Guérin , and A. Orda, " QoS-based Routing in Networks with Inaccurate Information: Theory and Algorithms", INFOCOM'97, pp.75-83, 1997.
[12] D.Bertsekas, Network Optimization: Continuous and Discrete Models, Athena Scientific, 1998
[13] Eric Crawley, Raj Nair, Bala Rajagopalan, and Hal Sandick, "A framework for QoS-based Routing in the Int rnet," IETF RFC 2386, August 1998.
[14] N.Rao and S.Batsell, "QoS Routing Via Multiple Paths Using Bandwidth Reservation", IEEE INF OCOM'98, March 1998.
[15] A. Orda, "Routing with End to End QoS Guarantees i Broadband Networks", *INFOCOM 9* , pp.27-34, 1998.
[16] D. Lorenz, and A. Orda, "QoS Routing in Networks wit Uncertain Parameters", *INFOCOM 9* , pp.3-10, 1998.
[17] X.Xiao, and L. Ni, "Internet Q oS: A Big Picture", *IEEE Network,* pp.8-18, March/April, 1999.
[18] Q. Ma and P. Steenkiste, "Supporting Dynamic Inter-Class Resource Sharing : A Multi-Class QoS Routing Algorithm", *INFOCOM99*, pp.649-660,1999.
[19] A.Orda, "Routing with End -to-End QoS Guarantees in Broadband Networks", IEEE/ACM Transactions on Networ king, Vol.7, No.3, June 1999.
[20] G.Apostolopoulos, R.Guerin, S.Kamat and S.K.Tripathi "Server Based QoS Routing", Globecom'99, pp.762-768, 1999
[21] S.Vutukury and J.J.Garcia-Luna-Aceves, "A Distributed Algorithm for Multipath Computation", Globecom'99, pp.1689-1693, 1999.
[22] T.Korkmaz and M.Krunz, "A Randomized Algorithm for Finding a Path Subject to Multiple QoS Constraints", Globecom'99, pp.1694-1698, 1999.
[23] B.Fortz and M.Thorup, "Internet Traffic Engineering by Optimizing OSPF Weights", INFOCOM2000, 2000.
[24] F.Hao and E.Zegura, "On Scalable QoS Routing: Performance Evaluation of Topology", INFOCOM2000, 2000.