

An Efficient QoS Routing Algorithm Based on Nonlinear Path Distance for Min-sharing Interference and Load Balancing

Xiaolong Yang¹, Shuheng Zhou¹, Min Zhang^{1,2}, Xuefei Tang², Jinde Liu²

¹ Chongqing Univ. of Post and Telecom., Chongqing, 400065, China

² Univ. of Electronic Science and Technology, Chengdu 610054, China

E-mai : (yangxl, zhangmin)@cqupt.edu.cn

Abstract

As known, it is difficult for the multiple constraints QoS routing to get a polynomial-time solution. Currently, many heuristic algorithms based on the path distance have been proposed to solve this problem. However, most of them are based on linear path distance, which has the greatest shortcoming, i.e., the search of feasible solutions is inefficient. Besides, they don't also consider the link sharing interference and load balancing. To improve the performance, this paper proposes a novel QoS routing based on the nonlinear path distance, which is called MIS-LB. At first, the additive and multiplicative QoS metrics are represented by a uniform metric, i.e. the nonlinear path distance. Then, the link sharing and load balancing are integrated into the uniform metric. Based on the shortest-first criterion, MIS-LB can not only find feasible paths as the normal routing algorithm (e.g., Dijkstra algorithm), but also adjust the link sharing, and balance the network loads between multiple feasible paths. So compared with TAMCRA and H_MCOP, the simulation results show that MIS-LB can further improve network performance.

Keywords: QoS Routing, Nonlinear Path Distance, Link Sharing Interference, Load Balancing.

1. Introduction

In traditional IP networks, packets are mostly delivered in the best-effort routing paradigm. However, The routing paradigm is primarily concerned with connectivity between source node and destination node, and without any assurance of quality of service (QoS). As usual, this routing paradigm characterizes the network with a single metric, such as hop-count or delay, and uses shortest-path algorithms for path computation. For traditional applications like File Transfer Protocol (FTP), the best-effort routing paradigm has been adequate. However, for many popular applications today, such as real time audio or video transmission, it is not enough. These

applications can be just acceptable if and only if their transmission paths satisfy a wide range of strict QoS requirements, such as bandwidth, delay, jitter and loss probability. Under this circumstance, in order to find a path that satisfies multiple QoS requirements, the QoS-aware routing algorithms are required. Besides the search of feasible paths satisfying corresponding QoS requirements from one side, i.e., the end-to end connectivity, QoS-aware routing algorithms have also concerned about the performance optimization from the other side, i.e., the resource utilization of the entire network, and then are expected to be able to find the best path for each arriving packet from the feasible path set. This is the key difference between the best-effort routing and the QoS routing.

As known, the QoS routing under multiple constraints, which can be additive, multiplicative and concave, is a NP-complete problem, and is difficult to get a polynomial-time solution [1]. As usual, it is solved approximately by some heuristic algorithms, and the shortest-path is one of the most practical heuristic criteria of this kind of algorithm. Up to now, many heuristic algorithms are proposed by the literatures (e.g., [3]-[6]). However, most of them are based on the linear path distance, which have wider meaning, and maybe the total cost, or the hop count, etc. And only a few algorithms are based on the nonlinear path distance. Compared with the linear QoS routing, the nonlinear QoS routing can more easily find feasible path from the candidate path set. Hence, the nonlinear routing always outperforms the linear routing.

Based on the property of QoS constraint, the multiplicative constraint can translate into the additive one by mathematic operations. In fact, the QoS routing with multiple constraints can formulate as a mathematic problem, i.e., linear programming, according to the theory of optimization. In the algorithms based on linear path distance, the weight of each type QoS metric contributing to the path distance is set in terms of the importance of each QoS metric. However, these weights are fixed and independent of

the QoS requirements of application; on the other hand, this distance representation makes the hit ratio of feasible path in the search space to be very low. The literature [3] presents an efficient definition of path distance, which exactly includes the information of the individual QoS metrics and the total QoS requirements. Under the definition, the path distance is nonlinear, and the additive and multiplicative QoS metrics can efficiently integrate into the uniform metric. Compared with the linear definition, it can increase the hit ratio of feasible path in the search space.

However, it is possible that in network there maybe exists sharing link among different flows paths. Each path can separately satisfies the QoS constraints of application, but the existence of link sharing would break the satisfaction of previous corresponding paths. On the other hand, it is possible that data of an application can be delivered in several feasible paths. In this situation, how to distribute the traffic load into these paths in order to obtain better performance is an important aspect of QoS routing. Hence, the computation of feasible routing must take the link sharing and load balancing into account. On the foundation of literature [3], the paper extends the definition of nonlinear distance, and also brings the link sharing and load balancing into the nonlinear path distance.

Based on the definition of nonlinear distance and its extension, this paper proposes a novel QoS routing to consider QoS constraints, link sharing interference and load balancing at the same time, called MIS-LB (Min-Sharing Interference and Load Balancing QoS Routing). Under the shortest-path first routing criterion, the path distance acts as the unique factor of QoS routing decision-making. MIS-LB can not only find feasible paths, but also adjust the link sharing according to the nonlinear path distance information, and finally balance the network loads between multiple feasible paths. The simulation results show that MIS-LB can efficiently improve network performance.

2. The definition of nonlinear path distance

2.1 The classification and additivity processing of QoS Constraints

Based on their properties of combination and operation, literature [2] classifies the QoS constraints into three types: additive constraint (e.g. delay, jitter, cost and hop count), multiplicative constraint (e.g. link reliability, and packet loss probability), and concave constraint (e.g. bandwidth). In practice, the QoS

requirements of an application can be flexibly represented by the three types constraints.

For simplicity, this paper firstly discusses two types constraints: additive and multiplicative. From the two types, we select one metric respectively as our discussion example, e.g. delay and reliability. For a QoS-aware application, the delay of its feasible paths must be less than an upper bound T . Because of the additivity of delay, the relationship between the individual link delay and the path delay can represent as follows

$$\sum_i delay_i \leq T \quad (1)$$

where $delay_i$ denotes the individual delay of a link i belonging to the feasible path.

Similarly, the reliability of its feasible paths must be no less than an upper bound P . Because of there multiplicativity of reliability, the relationship between the individual link reliability and the one of the entire path can represent as follows

$$\prod_i prob_i \geq p \quad (2)$$

where $prob_i$ denotes the individual reliability of a link i belonging to the feasible path.

From the representations (1) and (2), it is known that multiplicative constraints can be converted additive ones by some mathematical methods. For the representation (2), we can apply the logarithm operation to get another reliability representation, namely additive one, as shown in the representation (3)

$$\sum_i \log prob_i \geq \log p \quad (3)$$

After this kind of mathematical conversion, the individual multiplicative constraints of each link in the feasible path can become additive constraints.

For the concave constraint, such as the bandwidth, we can express it as follows

$$\min_i (BW_i) \geq BW \quad (4)$$

where BW_i denotes the individual available bandwidth of a link i belonging to the feasible path. When there are some concave constraints in QoS requirements of an application, links that cannot meet this type constraint are firstly pruned, and then the remains are processed based on the two other types constraints [5].

2.2 Nonlinear Path Distance

Assumed that an application has m types QoS constraints; and one of path between source node S and destination node D consists of h hops link, i.e.

$\overbrace{S - a - b - \dots - D}^{h\text{-hop}}$. lc_i , pc_i and L_i ($i=1,2, \dots, m$) denote i -th type metric of link, i -th type metric of path, and i -

i -th type QoS constraint. Hence, the QoS metrics of the link $a-b$ can be denoted by a vector of n dimensions

$$Link(a-b)=[lc_1(a-b),lc_2(a-b),\dots,lc_m(a-b)] \quad (5)$$

Similarly, the QoS metrics of the path $S-D$ can be denoted by a vector of n dimensions

$$Path(S-D)=[pc_1(S-D),pc_2(S-D),\dots,pc_m(S-D)] \quad (6)$$

Expression (5) and (6) hold the following relationship

$$pc_i(S-D) = \underbrace{lc_i(S-a) + lc_i(a-b) + \dots + lc_i(r-D)}_{h \text{ items}} \quad (7)$$

$$i = 1, 2, \dots, m$$

If the path distance is defined in linear form, the QoS routing problem can be formulated by the following mathematic optimization:

$$\begin{cases} \text{Min}[Length(S-D)] = \text{Min}[\sum_i w_i \cdot pc_i(S-D)] \\ \text{subject to: } pc_i(S-D) \leq L_i \quad i = 1, 2, \dots, m \end{cases} \quad (8)$$

where w_i denotes the weight value, which is the contribution proportion of each type QoS metric to the path distance.

As known, the optimization problem is a linear programming [2], which can be simply solved by illustrated method. The method is operating as follows: within the QoS constraint space, to shift a straight equilength line of object function toward the direction to have object minimizing, and to terminate the shift operation when equilength line is beyond the QoS constraint space. From the illustrated method and the literature [3], the feasible solution is better than any other ones outside the QoS constraint space, that is, the distance of path satisfying constraints is less than that of path not satisfying constraints in the maximum likelihood, if the weight and the QoS constraint of each type satisfy the following relationship:

$$w_1 : w_2 : \dots : w_m = \frac{1}{L_1} : \frac{1}{L_2} : \dots : \frac{1}{L_m} \quad (9)$$

Therefore, the above optimization problem with explicit constraint condition can approximately convert a normalization optimization problem with implicit constraint condition, which is rewritten as follows

$$\text{Min}[Length(S-D)] = \text{Min}[\sum_i \frac{pc_i(S-D)}{L_i}] \quad (10)$$

$$i = 1, 2, \dots, m$$

In fact, the expression (10) must subject to a connotative condition,

$$\frac{pc_i(S-D)}{L_i} \leq 1 \quad (11)$$

From the explicit geometrical meaning of the illustrated method, we can draw a conclusion that the search space of feasible solution can be effectively reduced if the equilength line of object function is similar to the boundary line of the QoS constraint space. Based on the conclusion, the definition of a

nonlinear path distance can be easily derived as the follows[3]

$$Length(S-D) = \left[\sum_i \left(\frac{pc_i(S-D)}{L_i} \right)^q \right]^{1/q} \quad (12)$$

$$i = 1, 2, \dots, m$$

where q denotes the nonlinear degree of path distance, and in the expression (12), $q > 1$. When $q=1$, the expression (12) is the definition of linear path distance. The equilength line is more similar to the boundary line if q is larger. When q is infinity, the nonlinear path distance can be ideally expressed as follows

$$Length(S-D) \Big|_{p \rightarrow \infty} \approx \text{Max}_i \left(\frac{pc_i(S-D)}{L_i} \right) \quad (13)$$

$$i = 1, 2, \dots, m$$

Hence, the QoS routing problem translates the following simple expression:

$$\text{Min}(Length(S-D)) = \text{Min} \left\{ \text{Max}_i \left(\frac{pc_i(S-D)}{L_i} \right) \right\} \quad (14)$$

$$i = 1, 2, \dots, m$$

2.3 The Interference for sharing link

When there exist sharing links among several paths, which have great impact on the QoS guarantee of each path. In the worst case, none of feasible paths can meet its QoS requirements due to sharing link. This is because the decision process of feasible path does not take the sharing link interference into consideration. Therefore, the literatures [5] and [6] have proposed the QoS routing algorithms based on minimum interference. The basic idea of them is in that the delivery of QoS-aware application data uses few sharing links as possible as it can, and so effectively reduces the interference to the other paths. Literature [5] defines the interference as the reduction in maxflow value between that source-destination pair due to the routing some traffic to sharing link. In this paper, we will define the sharing interference as the increment of nonlinear path distance.

Assumed that n paths, which have m types QoS constraints, share the link $a-b$. For k -th path, $h^{(k)}$, $lc^{(k)}_i$, and $pc^{(k)}_i$, $L^{(k)}_i$, $Link^{(k)}(a-b)$, and $Path^{(k)}(S-D)$ ($i=1,2,\dots,m$; $k=1,2,\dots,n$) denote the hop count, i -th type link QoS metric, i -th type path QoS constraint, i -th type QoS requirement of application, the link metrics vector, and the path metrics vector.

Based on the definitions in representations (5)-(7), we have the sharing link vector as follows

$$\text{Share_Link}(a-b) = [\text{Share_lc}_1(a-b), \text{Share_lc}_2(a-b), \dots, \text{Share_lc}_m(a-b)] \quad (15)$$

where $\text{Share_lc}_i(a-b) = \sum_k \text{Share_lc}_i^{(k)}(a-b)$
 $i = 1, 2, \dots, m; k = 1, 2, \dots, n$

We have the sharing path vector as follows

$$\text{Path}^{(k)}(S-D) = [pc_i^{(k)}(S-D), pc_2^{(k)}(S-D), \dots, pc_m^{(k)}(S-D)] \quad (16)$$

where

$$pc_i^{(k)}(S-D) = \underbrace{lc_i^{(k)}(S-a) + \text{Share_lc}_i(a-b) + \dots + lc_i^{(k)}(r-D)}_{h^{(k)} \text{ items}} \\ = lc_i^{(k)}(S-a) + \sum_k lc_i^{(k)}(a-b) + \dots + lc_i^{(k)}(r-D)$$

Note that the representation (16), which defines the nonlinear distance of path with a sharing link, is meaningless significant if the inequality $pc_i^{(k)}(S-D) \leq L_i^{(k)}$ does not hold. Under this case, the sharing link becomes a bottleneck link, and obviously the path distance should be defined infinite. This case shows that the n paths cannot share the link $a-b$ at the same time, and we should reduce the number of paths using the sharing link. In other words, we should route some traffic to another link, not to the sharing link.

When there are many flows in network, we can deal with them in the following way. The first step is that to compute possible routing paths for each flow according to the normal mode, and then the second step is that to judge whether there are sharing links among these possible routing paths of different flows. If there exists sharing link, and then these routing paths need to adjust after recalculating the path distance according to the representations (15) and (16).

2.4 Load Balancing based on Nonlinear Path Distance

As usual, the congestion of link will be heavy when traffic load increases. If traffic load exceeds the capacity of link, then the link will collapse. Under this case, the load balancing will be more important. Of course, the load balancing can be implemented by many ways. In this paper, we will propose the novel approach to obtain load balancing based on nonlinear path distance.

Therefore, we need to define a non-descending function with respect to path distance and traffic load, which should own the following form:

$$\Delta l = \begin{cases} 0 & v = 0 \\ f(v) & 0 < v < C \\ \infty & v \geq C \end{cases} \quad (17)$$

where Δl , v and C denote the increment of path distance, the input traffic load, and the maximal link capacity, respectively.

If so, having taken the part of load-related path distance into consideration, traffic load can be adaptively distributed among several feasible paths, and the load balancing can easily obtain since packet can always choose the shortest-path under the shortest-path first routing criterion.

In order that the sharing link and load balancing are expected to be simultaneously consider during the selection of feasible routing path, the definition of the nonlinear path length can be rewritten as follows according to the representations (13), (16) and (17)

$$\text{Length}(S-D) = \max_i \left(\frac{pc_i(S-D)}{L_i} \right) + \Delta l \quad (18) \\ i = 1, 2, \dots, m$$

3. MIS-LB: A QoS Routing Algorithm Based on Nonlinear Path Distance

According to the above definitions of the nonlinear path length with sharing link and load balancing, this section presents the implementation of our QoS routing algorithm. The algorithm works as follows. At first, it translates the QoS constraints into the nonlinear path distance, which contains the information of the link sharing interference and the traffic load distribution. Then under the shortest-path first routing criterion, while searching feasible routing paths from solution space, it adaptively regulates the sharing of link and the distribution of traffic load. Its implementation pseudo code is shown as Fig.1.

4. Numerical Simulation Results

In the section, we evaluate the performance of our proposed QoS routing algorithm, i.e. MIS-LB. The simulations are on the basis of Waxman-based random graph with 50 nodes, where there exists edge between any two nodes according to a probability expressed by [7]. Assumed that the link bandwidth uniformly distributes in the range [0, 155Mb/s], and the flows have three types QoS constraints: delay, packet loss probability (PLP), and bandwidth. Delay and PLP of link, both of which are independent to each other, also

Input:

$G = (V, E)$ //A graph with vertex set V and edge set E //

$(S-D)$ //A source-destination pair: s : source node, d : destination node //

m //The type number of QoS constraints //

Implementation:

1. for each $Link \in E$
2. if $Link$ does not satisfy the concave QoS constraint L_i " then $G'(V, E') = Prune(Link)$ //Prune these links not satisfying the concave QoS constraint //
3. if K feasible paths have been searched, or there exists none of feasible path in $G'(V, E')$, then stop
4. else $Path[k] \leftarrow Tamcra(G'(V, E'), m \text{ types QoS constraints}, S-D)$ // To invoke Tamcra function in the literature [3] to record the links of a feasible routing path //
5. if the intersection between $Path[k]$ and $Path[i]$ ($i < k$) is empty, or there is no traffic load in the path then $k \leftarrow k+1$
6. else to compute the distance with sharing link according to representation (15) and (18),
7. then to return the row 4
8. endif
9. endif

Output: K feasible paths

Fig.1 The pseudo code implementation of MIS-LB

uniformly distribute in $[1, 50\text{ms}]$ and $[10^{-6}, 0.3]$, respectively. The delay and PLP constraint of application are assumed to respectively distribute in $[10, 80\text{ms}]$ and $[10^{-4}, 0.4]$ while bandwidth requirement is variable. For simplicity, the function with respect to path distance and traffic load is defined as follows

$$\Delta l = f(v) = \frac{v}{C - v} \quad (19)$$

In the section, our performance is evaluated from

two sides, which is implemented by the comparison with the other two algorithms (i.e., TAMCRA [3] and H_MCOP [4]). One is the ratio that can find feasible paths to satisfy the QoS requirements of application flow; the other is the throughput. In order that the comparison is simple, our proposed algorithm is assumed to be able to find 5 shortest feasible paths at the ideal situation, and short for 5-MIS-LB in Fig.2 and Fig.3. Similarly, the tunable parameter of TAMCRA can also take 5 according to [3] (short for 5-TAMCRA) while the path index of H_MCOP is just

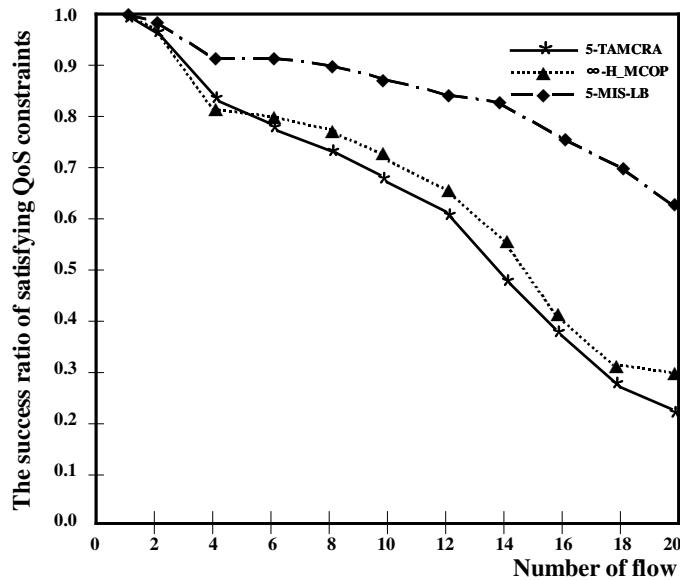


Fig.2. The performance comparison of QoS support of three algorithms (the bandwidth requirement of each flow is 20Mbps)

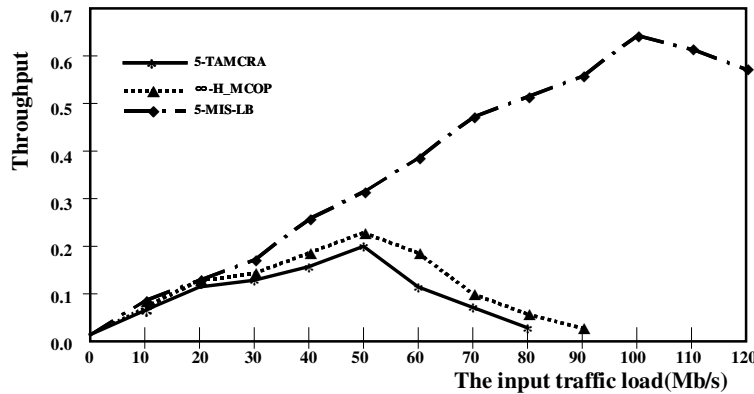


Fig.3. The throughput comparison of three algorithms (3 flows)

infinite according to [4] (short for ∞ -H_MCOP).

Usually, our proposed algorithm MIS-LB can support multi-path. However, the multi-path support capacities of TAMCRA and H_MCOP are indeed weaker, and the traffic is carried through the unique best routing path. Hence, some parts of network is possibly overloading in TAMCRA and H_MCOP routing algorithms. As illustrated in Fig.3, MIS-LB obviously outperforms the other two algorithms in throughput. This phenomenon can be explained by representations (17)-(19).

The simulation results shown in Fig.2 and Fig.3 are obtained under a Waxman random graph with 50 nodes. In addition, we also implement several simulations experiments in the graphs with 20, 40, and 100 nodes. These results almost follow the mentioned variation trend.

5. Conclusion

On the one hand, it is possible that data of a flow can be delivered in several paths. In this situation, how to distribute the traffic load into the several paths in order to obtain better performance is an important aspect of QoS routing. On the other hand, it is also possible that there maybe exists sharing link among different flows paths. Each path can separately satisfies the QoS constraints of flow, but the existence of link sharing would break the satisfaction of previous corresponding paths. Hence, the computation of feasible routing must take the link sharing and load balancing into account. This paper represents the additive and multiplicative QoS metrics in the uniform metric, i.e. the nonlinear path distance. Moreover, the paper also brings the link sharing and load balancing into the nonlinear path distance. Under the shortest-path first routing criterion, the path distance acts as the unique factor of QoS routing decision-making. So, it

can effectively simplify the process of QoS routing, but its performance does not degrade. Contrarily, compared with the other algorithms (i.e. 5- TAMCRA, and ∞ -H_MCOP), our proposed algorithm based the uniform metric, i.e. MIS-LB, can adaptively regulates the sharing of link and the distribution of traffic load. The simulation results show that MIS-LB outperforms the other two ones in throughput and the satisfying of the QoS constraints of flow.

REFERENCES

- [1] Z. Wang, J. Crowcroft, "Quality-of-service routing for supporting multimedia applications", IEEE JSAC, 14(7), p.1288-1234, Sept.1996.
- [2] Yingding Fu, Xiaoyu Cheng, and Yinghui Tang, "Optimization theory and method", Press of UESTC, Sept. 1996.
- [3] H.De Neve, P.Van Mieghem. "TAMCRA a tunable accuracy multiple constraints routing algorithm", Computer Communications, 23(7), p.667-679, March 2000
- [4] T.Korkmaz, M.Krunz, "Multi-Constrained Optimal Path Selection", Proc. of IEEE Infocom'01, vol.2, pp.834-843, April, 2001
- [5] Murali Kodialam,T.V.Lakshman, "Minimum Interference Routing with Applications to MPLS Traffic Engineering", Proc.of IEEE Infocom'00, vol.2, pp.884-893, March 2000.
- [6] S. Nelakuditi, Z.L. Zhang, R.P.Tsang, "Adaptive Proportional Routing: A Localized QoS Routing Approach", IEEE/ACM ToN, 10(6), pp.790-804, Dec. 2002.
- [7] K.I. Calvert, M.B. Doar, E.W. Zegura, "Modelling Internet Topology", IEEE Communications Magazine, 35(6), pp.160-163, June 1997