

HOP-BY-HOP QOS ROUTING USING STATISTICAL DISTRIBUTION-FREE APPROACH

Chan Siew Yin, Ling Teck Chaw and Mashkuri Yaacob

Faculty of Computer Science & Information Technology

University of Malaya

50603 Kuala Lumpur

Malaysia

email: sychan@perdana.um.edu.my

tchaw@um.edu.my

mashkuri}@um.edu.my

ABSTRACT

The advent of Quality-of-Service (QoS) routing has brought a wide range of applications to network users. While precise network state information is critical to QoS provision, maintaining such accuracy is almost impossible. Towards this end, we propose a hop-by-hop QoS mechanism to operate in networks with inaccurate information. The proposed mechanism, namely DF-PI, adopts the distribution-free (nonparametric) approach to construct two-sided prediction intervals. The prediction interval helps infer the future available bandwidth, and is used to generate the proposed QoS metric – statistical available bandwidth δ . “Widest”-shortest paths are calculated, by which δ replaces the instantaneous available bandwidth in the traditional widest-shortest routing algorithm (WSR). Relative to WSR, simulation results show that DF-PI achieves satisfactory performance in terms of packet loss, commit ratio, link utilization and average end-to-end delay, together with less update message overhead.

Keywords: QoS Routing, Prediction Intervals, Distribution-Free.

1.0 INTRODUCTION

In conjunction with satisfying the elevating QoS demands, the goal of QoS routing is twofold: (1) find network routes which can support the QoS traffic, and (2) increase the overall resource utilization by selecting alternative paths which circumvent “hot spots”. To date, the traditional Internet architecture providing only best-effort services fails to achieve this goal. While the cost of QoS routing is reasonable compared to the excessive benefits gained [1], QoS routing is difficult by nature due to the following reasons [2]:

1. *Diverse QoS requirements and constraints:* Often, multiple constraints or metrics are imposed in order that the stringent requirements for supporting QoS traffic could be guaranteed. Unavoidably, even some compositions with only two additive metrics (e.g. delay and cost) are proved to be NP-complete [3]. To cope with the NP-completeness, researchers have to resort to heuristic and approximation algorithms.
2. *Coexistence with best-effort traffic:* Both best-effort and QoS routing have different roles to play in most real-world networks. Best-effort routing optimizes the throughput and responsiveness of the network whilst QoS routing maximizes the admission of QoS connections. In spite of the counter-productive effects, best-effort and QoS traffic is to coexist so as to provide diverse network applications. How to finely tune the limited share to both types of traffic such that the best-effort traffic is not suffered remains to be a formidable task.
3. *Outdated network state information:* The information imprecision is attributed to network dynamics, propagation delay, computational cost, processing overhead, and aggregated state information contribute [4] [5]. Hence, an inaccurate piece of information served as the input of QoS path calculation will yield inaccurate decisions, leading to severe problems of underutilizing and over-utilizing the network resources.

[1] observes that the major QoS cost stems from the processing of link-state updates. To reduce the frequency of path computation and the amount of link-state update, [6] proposes considerable latitude in tuning the policy parameters. Link-state updates are advertised either periodically or when the changes in link utilization exceed a trigger-based threshold. Unlike the former policy which easily ignores major fluctuations, the latter one enables progressive notification of network congestion. Despite large periodic intervals and coarse triggers could significantly minimize the update overhead, the accuracy of network state information is sacrificed. Attaining the

trade-off between overheads and precision is undoubtedly challenging particularly when a large-scale network is involved.

Because of the complexity of QoS routing and the inevitability of information imprecision, designing QoS mechanisms which operate in such an environment is of paramount importance. Substantial amount of work has been carried out. [4], [5], [7] introduce and deploy the notion of link *safety* of how likely the link could satisfy the QoS requirement. The path safety is then defined as the probability production of all links of the path. The probability density function of the QoS parameter is assumed to be known in advance so as the link safety can be determined accordingly. Provided that this likelihood is not sufficiently high, [8], [9] adopt the idea of relative frequency distribution in the respective probability based routing schemes. Empirical probability distribution is calculated and is used for constructing the link safety. [10] maximizes the likelihood of searching a feasible path by sending multiple probes to perform multi-path parallel routing. Through balancing between flooding and single-path routing, a near-optimal performance is achieved. The enhanced performance, though, comes at the cost of higher overhead of processing the probes and determining eligible outgoing links.

In contrast to deterministic routing that always chooses the (seemingly) best path, randomized routing selects a path randomly from a set of candidate paths according to certain probability. [5] associates the probability with the safety difference between two consecutive candidate paths ordered with decreasing safety, whereas [11] associates the probability weighted with the difference between both the delay-and-safety requirement, and the delay-and-safety of the path. However, [11] did not detail how users should set the global safety requirement. [12] attempts to solve the so-called “magnet-phenomenon” problem by adaptively mapping the link utilization on the link cost. The mapping is performed such that a small change of link state in a highly utilized link would not produce a large change in the corresponding link cost, thus reducing unnecessary update traffic. Interestingly, each node is allowed to selectively use different (convex or concave) function at different time unit. One of the limitations of employing the two functions is that the rather slow adaptation of the link cost function may let major state changes go unnoticed. Also, when the frequency of link cost update could not cope with the environment of precise information, higher blocking rate occurs.

All the aforementioned mechanisms require the information of user requirements, e.g. bandwidth requirement, in order to perform on-demand routing. Inspired by [13] which concentrates on QoS extensions to Border Gateway Protocol (BGP), we explore hop-by-hop routing in an intra-domain basis. While [13] essentially employs confidence intervals to *describe* the path property, we use prediction intervals instead to *predict* the future link state in a longer term. Briefly, our proposed DF-PI mechanism incorporates some statistical properties into the link-state QoS routing approach. Without assuming any probability distribution on the residual bandwidth, two-sided distribution-free prediction intervals are computed. Based on the prediction interval, the statistical available bandwidth δ is defined to convey the current residual bandwidth and help infer the bandwidth available in some future time point. The hop-by-hop forwarding approach is the primary focus of this paper. Alternatively, the proposed mechanism can be applied to explicit QoS routing.

The rest of the paper is structured as follows. After presenting the proposed DF-PI in Section 2, Section 3 discusses the performance evaluation. The conclusion and future work is provided in Section 4.

2.0 PROPOSED DF-PI MECHANISM

A prediction interval (denoted hereafter as PI) is an interval that will, with a specified degree of confidence or prediction level $(1-\alpha)$ contain the next randomly selected observation(s) from a population [14]. A $100(1-\alpha)\%$ PI may be construed as in the long run, one would be correct $100(1-\alpha)\%$ of time in claiming that the future value(s) will be contained within the PI.

Given a sample $b(T)$ of size n in which the elements or observations $b(t_1), b(t_2), b(t_3), \dots, b(t_n)$ represent the available bandwidth of a particular link at time $t_1, t_2, t_3, \dots, t_n$ respectively. The values of the available bandwidth are taken by monitoring directly connected outgoing links. An element is sampled on every sampling interval T_s . $b(T)$ are notably time-series (nonrandom) data. Nevertheless, a more comprehensive model which takes into account the corresponding trend and seasonality is not considered in this work. Instead, for simplicity, the construction of PIs is based on the assumption that the values of available bandwidth are random, and independent and identically distributed (iid). As a result, the calculated PIs generally provide only a lower bound on the total uncertainty. This lower bound uncertainty consists of the following two sources [14]:

1. Since the sample size is limited and the random sample is assumed to be a fair representation of the population, the uncertainty exists in estimating the population parameters (e.g. population mean and standard deviation).
2. Random variation exists in the future sample

For random samples, the statistical method for constructing PIs depends on the type of population distribution. Despite this importance, to correctly identify the true distribution of the available bandwidth for any link before constructing a PI is very complex. Many factors can affect the distribution characteristics. Topology, traffic distribution, arrival rates, link capacity, and routing all contribute to determining the bandwidth distribution [5] [15]. Provided that recognizing the underlying distribution is notoriously difficult, this work applies the distribution-free approach without making any such assumption.

[14] provides the general method to construct a distribution-free PI:

1. Let *order statistics* of the sample be $x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(n)}$ whereby x_1, x_2, \dots, x_n are independent and each with any continuous distribution function $F(x)$
2. Specify the desired prediction level for the interval
3. Determine (from tabulations or calculations) the order statistics that provide the PI with at least the desired prediction level. If no such order statistics exist, use the extreme order statistics to obtain the interval endpoints and determine the associate prediction level
4. Use the selected order statistics as the endpoints of the distribution-free PI

To illustrate, let's construct a two-sided distribution-free 90% PI to contain all voltage measurements of five future electronic circuits from a previous sample of 100 electronic circuits. Part of the order statistics is as follows:

$$5.51, 5.67, 5.69, \dots, 45.32, 46.44, 49.05$$

By consulting the tabulations from [16], with given $n = 100$, $N = 5$, $K = 0.90$, the largest r to contain all five future observations is found through the body of the table. In this case, $r = 2$. Next, to choose l and u symmetrically, r is divided into $r_1 = 1$ and $r_2 = 1$ such that $l = r_1 = 1$ and $u = n - r_2 + 1 = 100$. Thus, the desired 90% two-sided PI to contain all five future observations is the range formed by

$$[x_{(1)}, x_{(100)}] = [5.51, 49.05]. \quad (1)$$

Since this interval uses the two extreme observations (i.e. the smallest and largest order statistics) of the initial sample as the endpoints of the PI, the actual prediction level $1 - \alpha$ is computed as

$$1 - \alpha = \frac{n(n-1)}{(n+N)(n+N-1)}. \quad (2)$$

That is, the actual prediction interval or confidence is $0.9066 = \frac{100(100-1)}{(100+5)(100+5-1)}$ rather than pre-defined $K = 0.90$.

Because [16] does not provide the entries for all possible combinations of parameters, this work adopts the extreme order statistics as the limits. Hence, to construct a distribution-free two-sided $100(1-\alpha)\%$ PI $[b_l, b_u]$ to contain all N future observations from sample $b(T)$, order statistics $b_1 \leq b_2 \leq \dots \leq b_n$ are built by sorting elements in $b(T)$ via the heapsort operation. The operation takes $O(n \log n)$ time and $\Theta(1)$ space [17]. Precisely, $[b_l, b_u] = [b_1, b_n]$, where b_1 and b_n are simply the minimum and maximum values in $b(T)$.

To make the PIs of two distinct links comparable, the QoS metric *statistical available bandwidth* δ is proposed. Besides calculating $[b_l, b_u]$, the current available bandwidth, b_n , is chosen as the stepping stone towards forecasting the unknown future available bandwidth. Due to the fact that experiencing burst traffic within a network is very common, b_n takes the average of the last five elements of sample $b(T)$ to avoid the impact of sudden increase/decrease of the traffic amount. Following this idea, the QoS metric δ is proposed as follows.

$$\delta = \frac{w_1 b_n + w_2 b_\ell + w_3 b_u}{w_1 + w_2 + w_3} \quad (3)$$

where $w_1 > w_2 > w_3$. In this strategy, b_n and b_ℓ have more importance than b_u so as δ expresses some view of the “least” future available bandwidth that could be provided. The values taken by w_1 , w_2 and w_3 could vary according to desirability. An aggressive approach is by putting the highest weight on b_u such that δ signalizes larger future available bandwidth. While the latter approach is worth exploring, this work begins with the former consecutive method.

Now, let $G = (V, E)$ be the directed graph where V is the set of nodes and E is the set edges in G . Assume that each node has the information of complete graph connectivity and the quantities δ_ℓ for every link $\ell \in E$. δ_ℓ is the calculated statistical available bandwidth (see Equation 1) in relation to an estimated indication of the future available bandwidth on link ℓ . δ_ℓ acts as a concave metric, that is, the statistical available bandwidth of a path p , δ_p , is calculated as $\min(\delta_\ell, \forall \ell \in p)$.

No new routing method is invented in this proposal. Taking place of the instantaneous available bandwidth, the metric δ_ℓ is utilized in the traditional widest-shortest routing (WSR). In other words, among the shortest paths to a destination, the one with the highest δ_p is selected. Based on the nature of δ_ℓ , the hop-by-hop QoS routing strategy is adopted, and “widest”-shortest paths to all other nodes are pre-computed independently at each source.

Advertising QoS metrics is necessary to make QoS routing decisions. To advertise δ_ℓ , the triggered-based update policy together with hold-down timers is employed. Under the triggered-based policy, when the percentage change of δ_ℓ value exceeds the predefined threshold, updates are flooded to the rest of the network.

3.0 SIMULATION RESULTS

Simulations are conducted to compare the proposed DF-PI mechanism with the traditional WSR. The simulation environment is detailed, followed by the performance metrics used for evaluation. Simulation results demonstrate that DF-PI is less sensitive than WSR, thus generating much lower protocol overhead.

3.1 Simulation environment

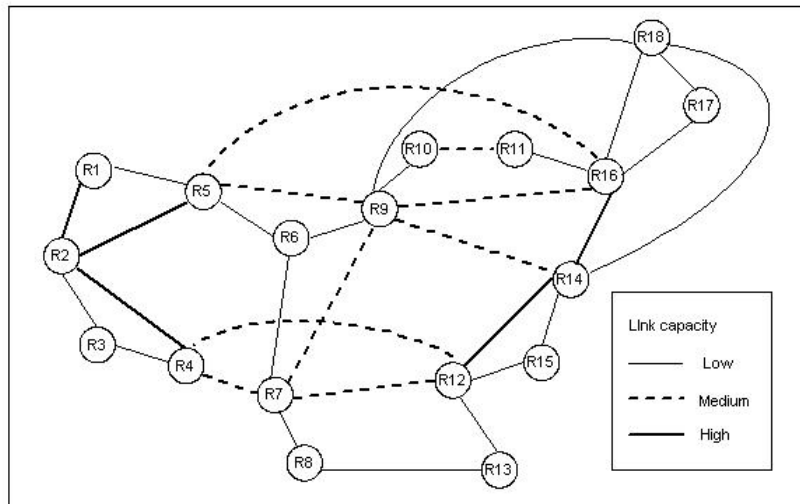


Fig. 1: Simulation topology

Fig. 1 depicts the MCI Internet backbone used as the simulation topology. The topology consists of 18 routers, 32 links and 216 traffic sources. The link capacity values are scaled down to reduce the simulation volume and are

categorized into three classes: class *low* has capacity 4 Mbps, class *medium* 5 Mbps and class *high* 7 Mbps. Every router is QoS-aware and connects to a customer site representing an aggregate of traffic. Each customer site contains a total of 2 constant-bit-rate (CBR) and 10 variable-bit-rate (VBR) applications. Traffic is sent to a destination site which is randomly selected. All sources use the ON/OFF model with Poisson inter-arrival time.

Six simulation sessions for a routing scheme with increasing network load are set up. Each session is run for 200 seconds. The first session begins with the normalized load 1.0. The load (in terms of transmission rate and amount of traffic) is then incremented by 20% of load 1.0 for every next session. For clarity, the six simulation sessions are grouped according to the normalized network load. Load 1.0 and 1.2 are categorized as *light load*, 1.4 and 1.6 as *medium load*, and 1.8 and 2.0 as *high load*. Table 1 shows the weight parameters for determining δ .

Table 1: System parameters for determining δ

w_1	w_2	w_3
10	5	1

Six metrics are defined to evaluate the performance of the routing protocols:

1. *Packet loss ratio*: The ratio between packets dropped and packets received
2. *Packet commit ratio*: The ratio of total delivered packets which have reached the destinations over total sent packets. At the time of measurement, some packets may be still in the transmission. Thus, only an approximation is given here for interpreting per-source packet loss ratio. Also, this parameter provides the measurement of the network throughput. Higher packet commit ratio signifies more packets are successfully received, hence is desirable
3. *Average link utilization*: Defines the link utilization on average for the whole network. A higher value of the parameter implies a higher level of network resource utilization and throughput
4. *Peak link utilization*: Among the long-term link utilization of all routers, the highest value is taken as the peak link utilization for the network. Lower peak link utilization is usually preferable as the load is deemed to be more evenly distributed within the network
5. *End-to-end delay*: The duration between the packet arrival time and sending time.
6. *Update message overhead*: The total number of protocol update messages exchanged in the network per time unit. Fewer updates signify lower overhead because less processing cost is incurred

3.2 Performance study

Fig. 2 shows the packet loss ratio for DF-PI and WSR with different normalized network load. As can be seen in the figure, WSR slightly outperforms DF-PI when the normalized load is light and moderate. The performance difference may be considered trivial, albeit DF-PI is inferior to WSR. At these stages, burst traffic greatly contributes to the network uncertainty. Therefore, high data variability exists in the sampled available bandwidths, leading to large width of distribution-free PIs. Subsequently, the generated QoS metric δ occasionally fails to correctly reflect both the current and future states.

When the load becomes heavy, dark horse DF-PI manages to defeat WSR. The noticeable difference begins at the crossover point (load = 1.6), in which DF-PI reduces the loss ratio by almost 10% of than in WSR when the load is 1.8. Intuitively, low variability has accounted for the improvement. Less data dispersion in heavy loads enables DF-PI to capture the history trend, which assists in inferring future available bandwidths more accurately. As a consequence, DF-PI is capable of “foreseeing” future network state and thus guiding more traffic to avoid congestion links.

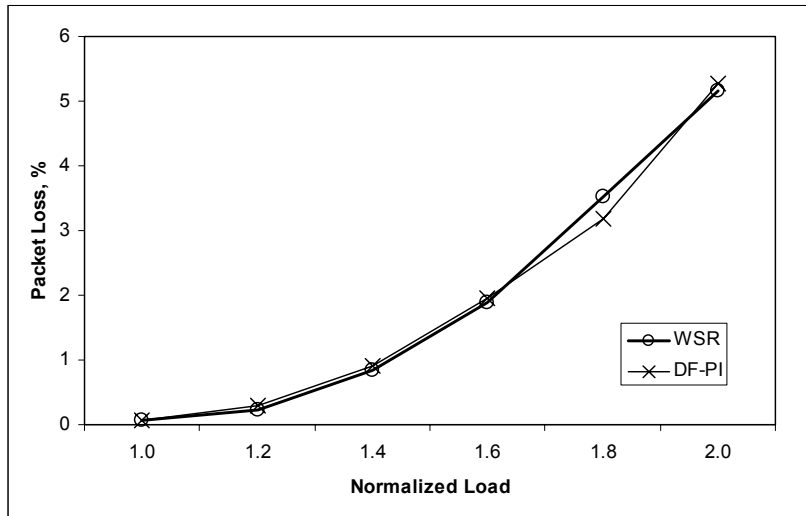


Fig. 2: Packet loss ratio

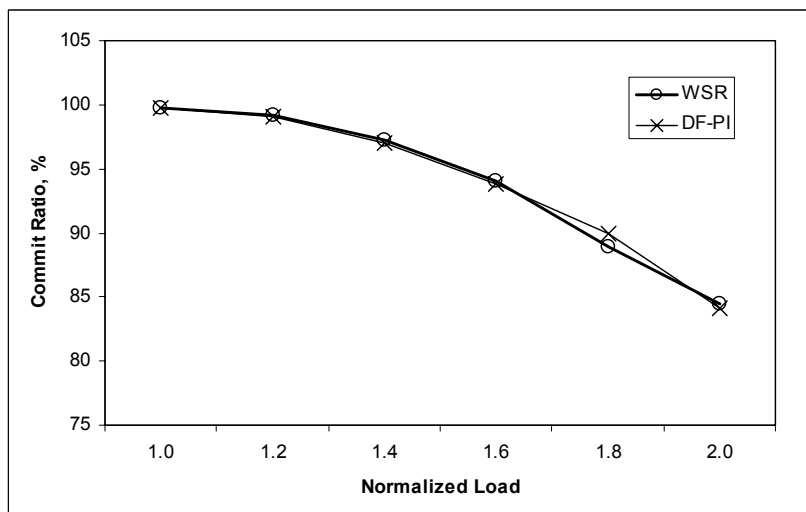


Fig. 3: Packet commit ratio

Fig. 3 illustrates the impact of increasing normalized network load on the packet commit ratio. Overall, the performance of both the mechanisms degrades with rising load. The circumstances are justifiable since more packets are still in the transmission (and may be dropped later). In fact, the results of packet commit ratio are consistent with that of packet loss ratio (previously discussed). When the load is 1.0, WSR and DF-PI achieve nearly 0% of loss ratio and 100% of commit ratio. When the load is 1.8, however, DF-PI has higher commit ratio compared with WSR. The counterpart situation is observed in Fig. 2 by which DF-PI experiences lower packet loss than WSR.

The performance of link utilization is demonstrated in Fig. 4 and Fig. 5 respectively. Fig. 4 shows a constantly growing in average link utilization against the network load. Additionally, WSR and DF-PI achieve very similar performance in every situation. Although the accomplishment of DF-PI may be infinitesimal, the picture of peak link utilization should be brought in for a fair comparison.

The peak link utilization of WSR and DF-PI is depicted in Fig. 5. The overall rise is compatible with the increasing average link utilization (see Fig. 4). Fig. 5 shows clearly that DF-PI has much lower peak link utilization as compared to WSR. The most significant reduction can be seen when the load is 1.0. At this point, DF-PI manages to

minimize nearly 20% of the peak link utilization of WSR. Associating the peak with the average measurement, DF-PI is more successful at balancing load and utilizing network resources.

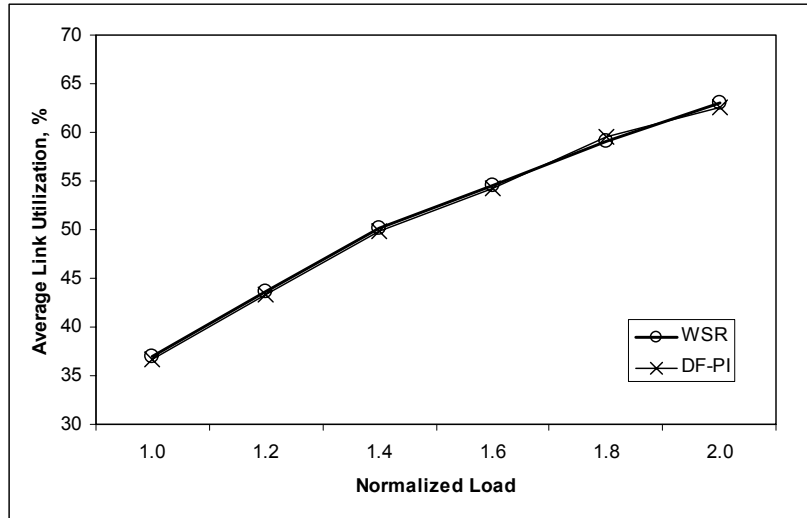


Fig. 4: Average link utilization

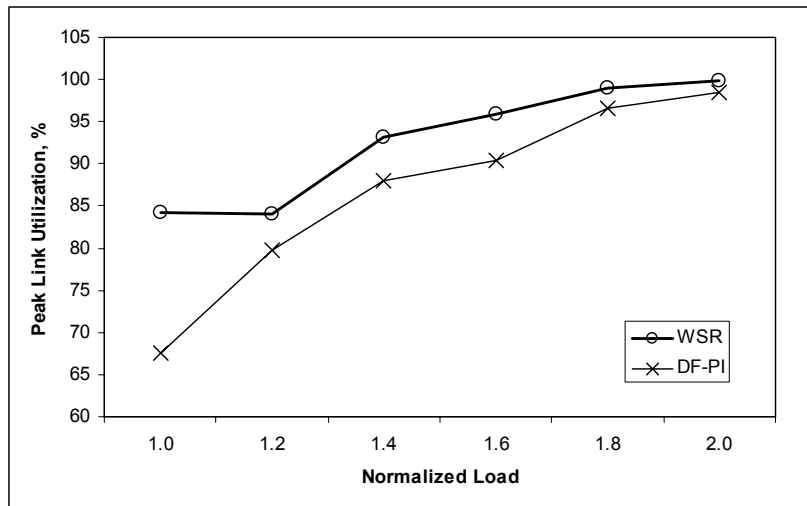


Fig. 5: Peak link utilization

Fig. 6 shows the average end-to-end delay of the routing mechanisms. For both the mechanisms, there is a growth in the average end-to-end delay with the increment of network load. DF-PI has slightly lower average end-to-end delay, and the difference is more apparent when the network load becomes moderate and high. Particularly, DF-PI reduces the average end-to-end ranging from 1.5% to 6.7% when the load is 1.4 and 1.6 respectively.

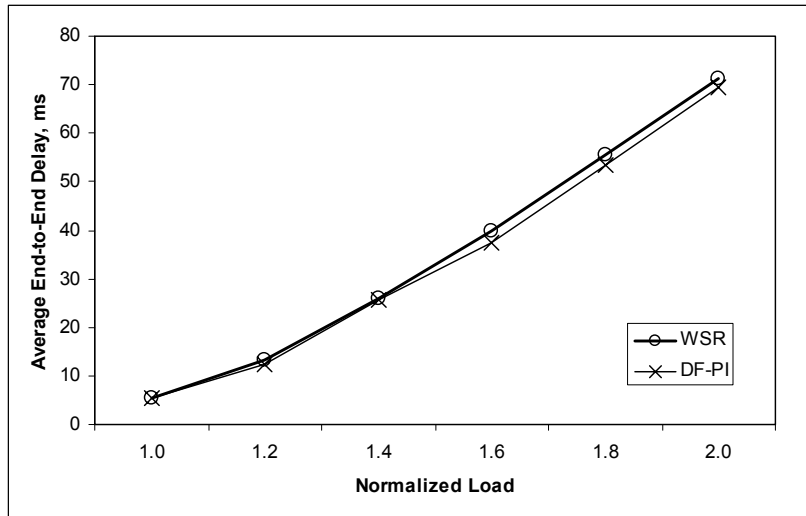


Fig. 6: Average end-to-end delay

Fig. 7 addresses the issue of update message overhead. At first glance, DF-PI has substantially low update message overhead with the increasing network load. In comparison to WSR, DF-PI manages to decrease the amount of overhead by at least 41% (load = 1.8) and at most 63% (load = 1.0). With exactly the same configuration of the triggered-based update policy (i.e. predefined threshold and hold-down timers), DF-PI succeeds to excel WSR in every scenario.

Notice that WSR experiences an irregular fluctuation of update overhead. This phenomenon could be attributed to the simulation topology which embraces mainly VBR applications modeling the web traffic. On occasion, the higher burst arrival rate and shorter mean interval between bursts generate more changes on the link utilization. Hence, the update policy of WSR which depends entirely on these immediate changes may have advertised more updates. Likewise, when the traffic burst is less, few update messages are broadcast, leading to lower update overhead.

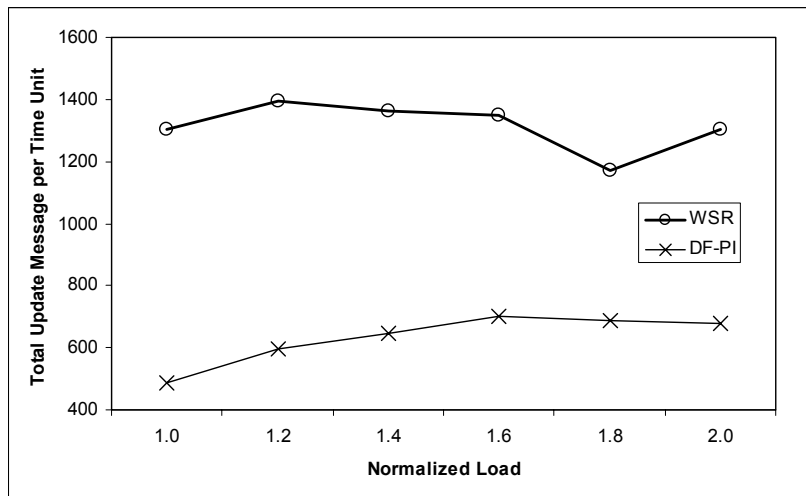


Fig. 7: Update message overhead

DF-PI, on the other, is not affected by the inherent nature of the VBR applications. Initially, the update overhead of DF-PI increases gradually as the load level moves from light to moderate. This increment is due to the consequence of advertising δ , the proposed QoS metric. As previously discussed, the light/moderate load causes high variability in the sampled history. The calculated δ may thus produce gradual rise in the magnitude of change, resulting in slight increments of the update amount. When the load becomes high, the update overhead of DF-PI remains to be

quite stable. Again, the low data variability in the sample probably generates small change magnitude in δ . Therefore, this metric which interprets the history of link usage depreciates the burst impact, causing a steady trend even if the network becomes increasingly congested.

4.0 CONCLUSION AND FUTURE WORK

The simulation results conclude that DF-PI is able to reduce the protocol overhead without curtailing the routing performance in terms of packet loss, commit ratio, link utilization and average end-to-end delay. In fact, DF-PI is shown to be marginally superior to WSR under certain circumstances. The ability to offset the impact of burst traffic makes DF-PI attractive to controlling the update frequency. To summarize, we affirm that DF-PI works well in the presence of information imprecision at reasonable computational cost.

The proposed mechanism chooses the two extreme elements to construct a distribution-free PI. Although such method is not complicated to use, the expected prediction level may fail to be achieved. In a situation where the sample is sufficiently large, the limits could be constructed by using other than extreme elements. As an alternative, for tabulations not provided in [16], the limits can be built by choosing elements (nearly) symmetrically from the extremes of the sample such that the probability of cumulative inverse hypergeometric is close to the desired prediction level. This method increases the flexibility in which the prediction level and the number of future elements are capable of determining the limits of prediction intervals. Further attempts may involve evaluating the method so as the corresponding iterative computation based on trial and error is worth implemented in large-scale networks.

5.0 REFERENCES

- [1] G. Apostolopoulos, R. Guerin, S. Kamat, A. Orda & S. K.Tripathi, "Intradomain QoS Routing in IP Networks: A Feasibility and Cost/Benefit Analysis", *IEEE Network*, Vol. 13 No. 5, 1991, pp. 42-54.
- [2] S. Chen, & K. Nahrstedt, "An Overview of Quality of Service for Next-Generation High-Speed Networks: Problems and Solutions", *IEEE Network Magazine, Special Issue on Transmission and Distribution of Digital Video*, Vol. 12 No. 6, 1998, pp. 64-79.
- [3] M. Garey, & D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: W. H. Freeman and Co, 1979.
- [4] R. Guerin, & A. Orda, "QoS-based Routing in Networks with Inaccurate Information: Theory and Algorithms", in *Proceedings IEEE, Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, 7-11 April 1997, pp. 75-83.
- [5] G. Apostolopoulos, R. Guerin, S. Kamat, & S. Tripathi, "Improving QoS Routing Performance Under Inaccurate Link State Information", in *Proceedings of the 16th International Teletrac Congress (ITC'16)*, United Kingdom, 7-11 June 1999.
- [6] G. Apostolopoulos, S. Kamat, D. Williams, R. Guerin, A. Orda, & T. Przygienda, "QoS Routing Mechanisms and OSPF Extensions", *RFC 2676*, 1999.
- [7] D. H. Lorenz, & A. Orda, "QoS Routing in Networks with Uncertain Parameters", *IEEE/ACM Trans. on Networking*, Vol. 6 No. 6, 1998, pp. 768-778.
- [8] D. Ghosh, & R. Acharya, "A Probabilistic scheme for Hierarchical Routing", in *Proceedings of Ninth IEEE International Conference on Networks*, IEEE, 10 -12 October 2001, pp. 416-421.
- [9] X. Yuan, & G. Yang, "Empirical Probability Based QoS Routing", in *Proceedings of IEEE International Conference on Communications*, 11-15 May 2001, pp. 1713-7.
- [10] S. Chen, & K. Nahrstedt, *Distributed QoS Routing with Imprecise State Information*. Technical Report. Department of Computer Science, University of Illinois at Urbana-Champaign, 1998.
- [11] J. Wang, W. Wang, J. Chen, & S. Chen, "A Randomized QoS Routing Algorithm on Networks with

- Inaccurate Link-State Information”, in *Proceedings of International Conference Communication Technology*, 21-25 August 2000, pp. 1617-1622.
- [12] K. Kowalik, & M. Collier, “ALCFRA – A Robust Routing Algorithm Which Can Tolerate Imprecise Network State Information”, in *Proceedings of 15th ITC Specialist Seminar*, Wurzburg, Germany, 22-24 July 2002.
- [13] L. Xiao, K. S. Lui, J. Wang, & K. Nahrstedt, “Advertising Interdomain QoS Routing Information”, *IEEE Journal on Selected Areas in Communications*. Vol. 22 No. 10, 2004, pp. 1949-1964.
- [14] G. J. Hahn, & W. Q. Meeker, *Statistical Intervals: A Guide for Practitioners*. New York: John Wiley & Sons, 1991.
- [15] Z. Wang, *Emerging Technology: IP Routing Gets Smart*, <http://www.networkingmagazine.com/shared/article/showArticle.jhtml?articleId=8703239>, 2001.
- [16] L. Danziger, & S. A Davis, “Tables of Distribution-free Tolerance Limits”, *Annals of Mathematical Statistics*, Vol. 35 No. 3, 1964, pp. 1361-7.
- [17] M. H. Alsuwailay, *Algorithms: Design Techniques and Analysis*. Singapore: World Scientific, 1999.

6.0 BIOGRAPHY

Chan Siew Yin received the B.IT. degree and M. Comp. Sc from University of Malaya, Malaysia, in 2003 and 2005, respectively. Currently, she is a research assistant in the Faculty of Computer Science & Information Technology at University of Malaya. Her research interests include QoS routing and Grid computing.

Ling Teck Chaw obtained his M. Comp. Sc. and PhD in 1996 and 2005 from University of Malaya. He is a lecturer at the Faculty of Computer Science & Information Technology, University of Malaya, Malaysia. His research areas include Core network research, Inter-domain QoS, VoIP, Grid computing and Network security.

Mashkuri Yaacob is a Senior Professor of Computer Science in the Faculty of Computer Science and Information Technology, University of Malaya. He joined the University of Malaya in 1976 and obtained his PhD in 1980 from UMIST. He has published over 190 research papers and presented papers at both local and international conferences. He is a fellow of the IEE and the Academy of Science of Malaysia. Currently, he is the chairman of IEE Malaysia branch. His research interests are in computer architecture, computer networks and VLSI design.