Proactive Multi-path Routing[1]

Jing Shen[*], Jiaoying Shi[*], Jon Crowcroft[+]
[*]State Key Lab of CAD&CG, ZheJiang University, P.R.China
{jshen,jyshi}@cad.zju.edu.cn
[+]Computer Lab, University of Cambridge, UK
Jon.Crowcroft@cl.cam.ac.uk

Abstract. Internet service provider faces a daunting challenge in provisioning network efficiently. We introduce a proactive multipath routing scheme that tries to route traffic according to its built-in properties. Based on mathematical analysis, our approach disperses incoming traffic flows onto multiple paths according to path qualities. Long-lived flows are detected and migrated to the shortest path if their QoS could be guaranteed there. Suggesting non-disjoint path set, four types of dispersion policies are analyzed, and flow classification policy which relates flow trigger with link state update period is investigated. Simulation experiments show that our approach outperforms traditional single path routing significantly.

1. Introduction

Provisioning ISP's network efficiently is a challenge for current technology. The difficulties root in two aspects: the rapid growth of network and the exponential growth of network usage. Although people have been working with network routing for years and overprovision are widely used by ISPs, there are times that a part of network resource is insufficient to deal with its load while adjacent part is idled.

To make full use of network resource, traffic dispersion has been an active research area in both circuit switching and packet switching network [2][3][4][5][6][7][8]. It is shown that network performance could be improved by spreading traffic equally onto multiple paths. In other related work, alternative routing has been investigated extensively[9][10][11][12][13], which proposes to improve QoS by rerouting traffic over a precomputed alternative path if the shortest path is congested.

Based on the research mentioned above, this paper addresses one of key issues in traffic transmission, i.e., how could traffic be routed proactively over network according to their built-in properties. We focus on networks with non-uniformed traffic pattern and those traffic exhibits correlation with a high variance over long time period. We propose routing scheme that routes traffic according to its built-in properties while making full use of network performance. The idea is to disperse incoming traffic flows proactively onto multiple paths and to optimize routing

[1] This work was supported by China National Science Foundation for Innovative Colony under Contract No. 60021201.

paths for long-lived flows according to QoS and load balancing requirements.

Section 2 outlines the proactive multipath routing scheme on the basis of theoretical analysis. Section 3 follows with policies for path establishment, traffic dispersion and rerouting of long-lived flows. The benefit of our proactive approach is illustrated in section 4 through detailed simulation. Section 5 concludes the paper with summary of our work.

2. Routing with traffic characteristics

In this section, we argue that routing mechanism should be built with careful consideration on traffic characteristics, we shows that proactive multipath routing can improve network performance effectively.

2.1 The challenge for Internet traffic dispersion

People have been working with network routing for years. Most effort was expanded on path finding and on distributed routing algorithm. If we take routing system development as practice of complex system design, such research will not lead to the optimal solution for Internet traffic transmission, because properties of possible load is overlooked. Given an existing network infrastructure, routing mechanism should be designed with careful consideration on traffic characteristics, network connectivity and administration requirements.

It has been found that, internet traffic exhibits long-term correlations with high variance in packet arrivals, and connection size has a so-called "heavy-tail" distribution: most flows are short and small but the number which are very long tend to contribute the majority of traffic[17]. Defining burst as flows whose lifetime is shorter than the time scale of link state advertisements, Xun[8] invoked a theoretical analysis to find best way of bursty traffic transmission. In this model, a pair of ingress and egress nodes are interconnected via a set of n links with the same capacity c, bursty flows arrive at source node with λ_{sd} and average duration time is μ^{-1}. It is demonstrated that, minimum of network congestion could be reached by spreading traffic equally onto k paths according to one of the following equations:

$$\alpha^{*} = \min\left\{\sqrt{\frac{\lambda}{\mu c}\frac{(1-e^{-\mu t})}{e^{-\mu t}}},1\right\} \quad (1)$$

$$y^{*} = \min\left\{\sqrt{\frac{\lambda c}{\mu}\frac{(1-e^{-\mu t})}{e^{-\mu t}}},c\right\} \quad (2)$$

Where, $\lambda=\lambda_{sd}/k$, $k = \alpha^{*} \bullet n$ or k indicates a set of links whose load are below the threshold y^{*}. Although Xun's model is not to the real situation in Internet, it gives a useful starting point. That is, network node should route its connection requests actively over a large set of paths if these requests are mostly short and arrive frequently, or direct its connection requests to the least loaded path if these requests are mostly long and arriving infrequently. To Internet traffic, this means: the large amount of short flows should be spread onto multiple paths, while those long-lived flows should be transmitted along the least burdened path. As most path computing algorithms relate path length with path load, the least burdened path usually means the "shortest" path.

In fact, the multiple paths for traffic dispersion need not to be totally disjoint. There are three reasons: 1) the goal for multipath routing is to improve network performance by making full use of network resource, when a new loopless path is calculated, it is ensured that new resource will be introduced even though it is not totally disjoint from existing paths; 2) if traffic is dispersed onto several disjoint paths between source and destination, it cannot be guaranteed that there is no overlapping between paths for different source destination pairs; 3) if we constraint multipath routing to disjoint paths, nodes with only one outgoing link will always fail in finding multiple paths even when there is idle resource.

Some other problems exist. Firstly, how could the path set keep up with time varying network state? Since link state propagates by relaying between network nodes, staleness in link state information is unavoidable, but the frequency of link state advertisement do have effect on path computation. To reflect network dynamics, path computing needs to be scheduled periodically or triggered by some conditions. On the other hand, the path set should not be modified frequently to avoid frequent route alternation and unreasonable signaling overhead.

Secondly, how should traffic be dispersed onto multiple paths with different quality? Since it's unavoidable that overlapping happens between logical paths, load on some links may accumulate in hot spots. On the other hand, great difference between traffic flows may derive to sharp variation in link state. To optimize network performance, the path assignment policy should be designed with full consideration of path quality, service quality and load balancing requirements.

2.2 Dispersing traffic proactively

To address these requirements, we propose a proactive mutlipath routing scheme that tries to optimize routes for long-lived flows on the basis of dispersity routing. Our approach improves network performance in three critical ways:

- **More resource for traffic transmission**: The amount of resource for traffic transmission is increased by establishing multiple paths between source and destination, dispersity routing restraints the transient overload caused by bursty flows. Thus, the requirement for router's buffer size may be reduced as well.
- **Flow distribution is optimized**: Migration of long-lived flows focuses the "quasi-static" part of network burden onto the shortest path, while bursty traffic is spread onto a large set of transmission links. This way, load distribution and resource utilization are optimized.
- **Fewer link state update message**: Load division lowers the additive effects of bursty flows, and the relative targeting of long-lived flows restrains the link state variation along the shortest path. The result is, links are kept in relatively stable state and link state advertisements can be scheduled with long period.

To exploit these potential benefits, two key design decisions were adopted:

- **Spreading traffic flows according to path quality**
  Considering both path overlapping and self-similarity in internet traffic, it cannot be expected that each path provides service with the same quality or

that the resource consumed by traffic flows coincides with logical paths. To achieve a balance between QoS and traffic engineering, flow level traffic is spread across a set of paths according to their quality which is evaluated periodically.

- **Migrating long-lived flows only if their quality-of-service could be guaranteed**

  There are two possible methods to improve transmission quality of long-lived flows: 1.) reserving resource for long-lived flows; 2.) optimizing routes for long-lived flows on the basis of traffic dispersion. Although trunk reservation for long-lived flows could improve successful rate of flow migration, it is not adopted by our scheme. Three reasons contribute to this decision: 1.) if a part of network resource is reserved (hard or soft) for either long-lived flows or short flows, dispersity routing should be done according to the state of resource allocated to bursty traffic, it's difficult to predict how much resource should be reserved and the overhead of state maintenance will be high; 2.) if trunk reservation is proposed along the shortest path, it is difficult to predict whether a link carries a shortest-path; 3.) by choosing path quality metric and dispersing policy carefully, most of long-lived flows could be assigned to the shortest path from the start. Considering the requirement of QoS provisioning,long-lived flows are migrated only if their bandwidth requirement could be guaranteed.

Another critical problem is the number of paths. Xun[8] suggested a dynamic path set whose length is limited by $(1+\beta^{*})\bullet l_{shortest}$, where β^{*} is a parameter varying with flow arrival rate, flow duration time and link update period. Great variation in Internet traffic makes it very difficult to implement this scheme, while random selection may affect the effectiveness of our scheme. Considering the fact that most of current networks have a diameter less than 10 and node degree is usually below 4, two or three recommended value may be reasonable.

3. Traffic dispersion and rerouting of long-lived flows

This section describes details of proactive multipath routing, including k paths maintenance, traffic dispersion policy and rerouting of long-lived flows.

3.1 Computing and establishing k paths

According to discussion in section two, routing paths should be computed on the state of link capacity. There exist many types of link metrics, including bandwidth, delay, loss rate etc. In our approach, we choose the bandwidth related metric because it contains the fundamental information of throughput. In addition, it has been shown[1] that some other quality-of-service requirements can be converted to bandwidth requirement, and extensions to current routing protocols have been proposed to carry bandwidth information in LSAs. Getting link state information, routing paths could be computed easily by using an extended Dijkstra algorithm or a labeling algorithm at ingress node[16].

The k paths could be established by creating label switching paths in MPLS networks, which populates the forwarding tables in routers along the path. In this scenario, edge routers invoke explicit signaling along the path computed by using CR-LDP or RSVP-TE protocol. If network consists of ATM switches, the k paths can be established by connection signaling similar to MPOA. The path set is maintained in switch routing tables until refresh or recomputing happens.

In order to keep up with time varying network status, it's valuable to recompute the k paths periodically, but frequent computing and signaling may introduce unreasonable overhead and lead to frequent route alternation. To balance these factors, path computing should be scheduled with periods which are at least several times of long-lived flows' maximum lifetime, e.g. one or two days.

3.2 Path evaluation and traffic dispersion

There may be great difference between the k paths for traffic dispersion. Evaluation result of the same path may also differ greatly if different metric is applied. As our approach focuses on improving network performance, only those metrics emphasizing TE&QoS requirements are considered.

With the first type of metric, each path is considered to have the same quality. This may be reasonable if the load on each path equals and their end-to-end transmission capacity is the same. Spreading traffic with this metric, each path should be assigned a same number of flows. Particularly, the outgoing path for an incoming flow could be chosen from the path set by random selection, hash based selection or round robin. We do not recommend hash selection based on header information, because it may introduce load accumulation on a particular path if most flows are between the same subnets. The most outstanding advantage of this metric is, no computing load is introduced. We denote this dispersion policy "random".

With the second type of metric, we focus on end-to-end transmission capacity. As throughput is determined by bottleneck bandwidth, it is chosen as path quality metric. That is, the wider the bottleneck is, the more traffic should be transferred along the path. In order to reflect difference between k paths, weight is assigned to each path by dividing its quality value with the minimum of those sharing the same source and destination. Ingress nodes use weighted round robin as path assignment algorithm. We denote this type of method "wrr-bw".

Using the third and the forth metric, we focus on path quality in end-to-end operational regime. Considering traffic routing as the multiplexing of packets over some part of network resource, hop-count and bottleneck bandwidth indicate the amount of resource consumed when packet stream is transferred from source to destination. In this way, path quality on resource occupation could be identified by "bottleneck_bandwidth/hop_count", which is the third metric we use. On the other hand, as we use bandwidth related link metric in route computation, the path cost reflects the end-to-end transmission delay, then the metric on QoS provisioning could be defined as "bottleneck_bandwidth/path_cost". As with metric two, weighted round robin is used as scheduling algorithm for metric three and four. We denote dispersion policy using metric three and four "wrr-bh" and "wrr-bp" respectfully.

To adapt to time varying network state, path quality evaluation has to be scheduled periodically or triggered by some threshold conditions. As with link state

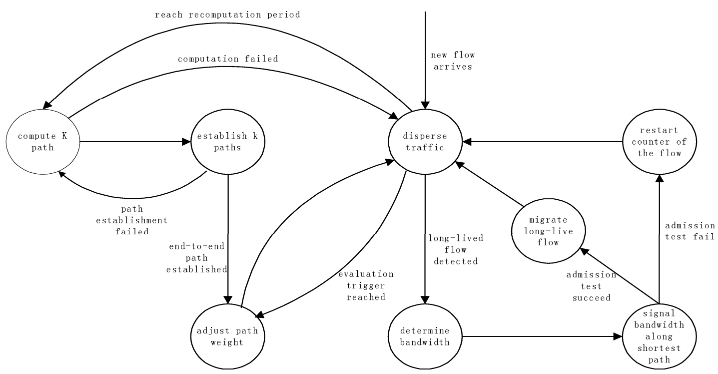
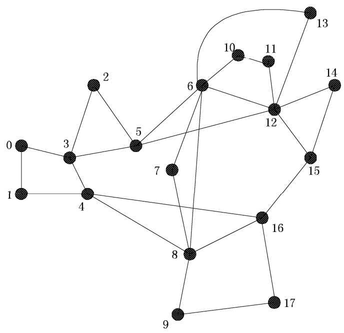


Figure 1: **State Machine of Proactive Multipath Routing**

advertisement, the frequency of path weight adjustment is a tradeoff between performance and overhead. As all information for quality evaluation is learnt from link state advertisement, the evaluation process should be invoked when the number of links announcing new state reaches some threshold value. Based on research on link state advertisement policy, threshold between 0.3 and 0.5 is recommended.

3.3 Detecting long-lived flows and rerouting

Traffic flow can be defined by fields in the IP and TCP/UDP headers. Depending on the exact definition, flow could be of the same TCP connection, UDP stream or the same host end-points or subnets. As most of current routing equipments have been implemented with hardware monitoring and classifying incoming traffic, it possible to identify long-lived flows from a large amount of incoming traffic. By default, routers forward arriving flows according to the traffic dispersion policy discussed in the previous section. Once the accumulated size or duration time of a flow exceeds some threshold (in terms of bytes, packet number or seconds) a long-lived flow is identified. As has been found[17] that single feature classification could not achieve consistency simultaneously in the fraction of total load due to elephants and the number of elephants, two or more feature classification method is preferred by our approach, e.g. long-lived flows are those who have been active for more than a special time period and transfer more than *N* bytes of payload. Once a long-lived flow is identified on a non-shortest path, the router will try to determine it's bandwidth requirement, and signal that value along the shortest path to find out whether there is enough room for migration. If the admission test succeeds, long-lived flow will be migrated to the shortest path by modifying label-binding information, otherwise it will be left on the original path. If it is desired, a counter for the long-lived flow, which is not migrated, could be started over to give another chance of migration. It must be emphasized that the signaling procedure does not reserve any resource for long-lived flows. The path quality does not need not to be reevaluated immediately after flow migration, because "long-lived flow" does not necessarily equate to "high bandwidth flow". Figure 1 gives the



Ingress	Egress	Hop distance	Arrival rate
1	12	4	10.6
0	14	4	10.6
2	17	4	10.6
9	11	4	10.6
6	17	3	10.6
4	13	3	10.6
10	8	2	10.6

Figure 2: **MCI Topology and Non-uniform Traffic Matrix**

state machine for our approach.

4. Performance Evaluation

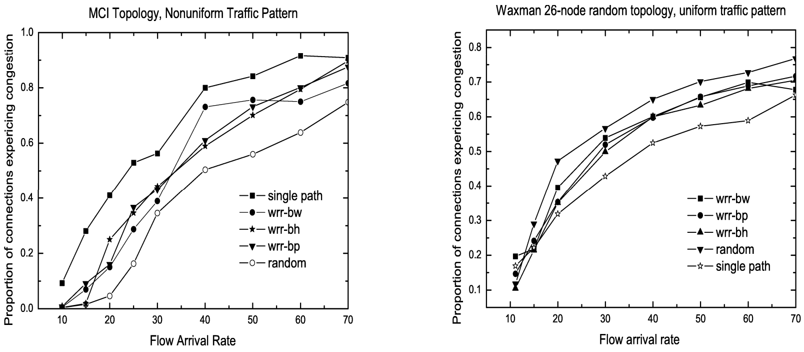
This section evaluates the performance of our approach to proactive multipath rout-ing based on detailed simulation experiments. After a brief description of simulation methodology, we compare our scheme to traditional single path routing under a wide range of load levels and network topology. Experiment results show that, our approach outperforms traditional single path routing in various situations.

4.1 Simulation Setup

We use an event driven simulator in our experiments. The simulator operates at flow level to evaluate the essentials of multiple routing policies and network configuration. Figure 2 shows the topology used in our experiments. For simplicity all links are assumed to be bi-directional and of equal capacity in each direction. During the simulation, flows arrive to the network according to a Poisson process, and the flow holding time is Pareto distributed. The ingress and the egress node of the flows are selected according to Figure 2, which are set up to model a typical WAN traffic pattern. Other parameters are set as follows: link capacity is set to 155 units, and bandwidth request is uniformly distributed between 1.14 units and 2.46 units. Traffic load is increased by scaling the arrival rate as shown in Figure 3. Unless explicitly stated, we use dynamic link metric and the routers exchange link states periodically with a period of 10 seconds.

4.2 Network throughput

From figure 3(a), it's evident that network performance is improved by our scheme when traffic is not uniformly distributed. Specifically, when traffic is not uniformly distributed the relative congestion status is decreased by a factor varying from 90% (arrival rate 10) to 3%(arrival rate 70). This means our approach can improve net work performance effectively, especially when the idled resource is plentiful or load on transmission path is not so heavy. This conclusion is confirmed by Figure 3(b) which plots the result of a 26-node waxman topology with uniformed



(a) Non-uniform traffic pattern

(b) Uniform traffic pattern

Figure 3: **Throughput of Proactive Multipath Routing**

traffic pattern. It could be found that, in a network with uniformed traffic pattern our approach outperforms single path routing only when the network is lightly burdened.

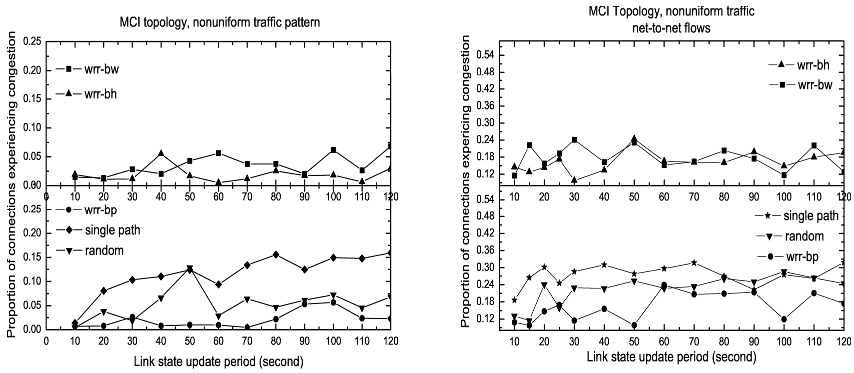
These results reflect the essentials of our scheme: absorbing traffic by multiplexing them onto more network resource. When there is a plenty of idle resource, our method can improve network performance effectively; but if there is little idled resource, traffic dispersion may deteriorate network performance.

4.3 Stale link-state information

Figure 4(a) plots network performance as a function of link-state update period for dynamic single path routing (SPT) and our approach. It may be found that, although both methods succeed in path finding performance of single path routing is much more sensitive to inaccuracies in link-state information. Measuring either with absolute method or with relative method, performance of our approach is more stable than that of single path routing. In particular, when link-state update period increases from 10 seconds to 120 seconds proportion of flows experiencing congestion rises from 0.014 to 0.159 in single path routing network, while the value with “wrr-bp” is 0.008 to 0.032. Among the four quality metrics, “wrr-bp” and “wrr-bh” are more resistant to stale link state information than “wrr-bw” and “random”.

Such a result is directly related to the policy behind each method. In single path routing, all traffic follows the unique path computed from link state information, the staleness of link state information affects network performance directly. With our approach, traffic is spread onto a set of paths, the increase of usable resource weakens the importance of computing accuracy. When a “random” policy is used, the great variation in Internet traffic makes it nearly impossible to reach an optimal flow distribution, so accuracy of path computing plays an important role in network performance. The situation improves with “wrr-bp” and “wrr-bh”, whose robustness is tightened by dispersing traffic on basis of path quality evaluation, while properties of “wrr-bw” lies between “random” and “wrr-bp”(“wrr-bh”).

We also consider the effect of link-state staleness when traffic is aggregated. Figure 4(b) plots the situation when average duration time is improved by 1.8



(a) Link-state update period (b) Increased traffic aggregation

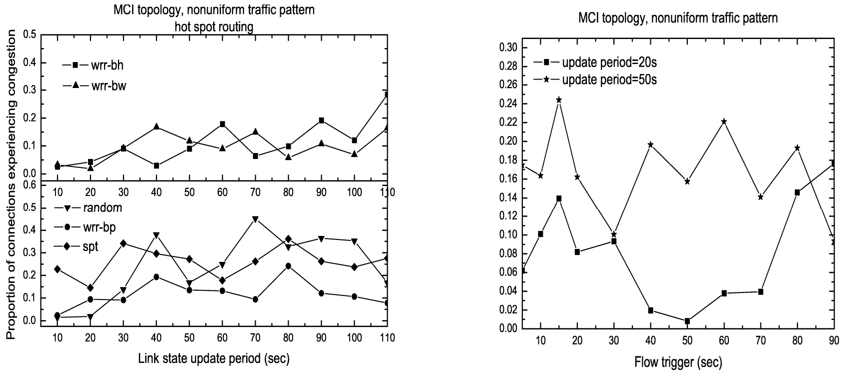
Figure 4: **Effect of Link-state Staleness**

times.It is found, as aggregation increases advantage of proactive multipath routing over single path routing dismisses somewhat but still significant.

To further evaluate the efficiency of our approach, we vary the flow arrivals to the network so that certain “hot-spot” present. Specifically, keeping total ejection rate the same, node 1 sends a part of its traffic to node 17 at a rate of 3.6 flows per second, and so does node 0 to node 13. Figure 5(a) shows that the improvement of network performance achieved by our approach over dynamic single path routing is still significant (20% to 90% of congestion removed). We note that, under this situation there is crossover between “random” and single path routing. This reflects the fact that random selection of outgoing path may worsen the performance under some situation. To be reliable for load balancing, the “random” policy should be avoided in network with hot spots. Unlike the previous experiments, “wrr-bw” outperforms others in hot-spotted network. The reason is, when hot spot appears some nodes receive more traffic than others, traffic to these destinations may compete for the same part of transmission resource. The result is that the bottleneck bandwidth becomes the key factor determining congestion status.

4.4 Doing with long-lived flows

To make it simple, we focus on single feature classification in simulation experiments. Shaikh[9] showed that, by relating flow-trigger with time scale of link state advertisement network performance of load-sensitive routing could be improved. We investigate this policy with our approach by experimenting with “wrr-bp”. Experiments varying the flow trigger illustrate that computing load decreases with increase of flow trigger. This stems directly from the fact that only a small number of flows will be identified and rerouted under large trigger time. Link state update period has little effect on flow detection, while the number of migrated flow decreases if link state update period is lengthened. In particular, with flow trigger of 20s, the percentage of flows migrated decreases from 3.4% to 0.03% when link state update period changes from 20 seconds to 50 seconds, while proportion of flows detected just decreases from 3.95% to 3.58%. The reason is that, when the staleness of link



(a) Effect of Hot-spot Traffi**c** (b) Choice of flow trigger

Figure 5: Trigger in Proactive Multipath Routing

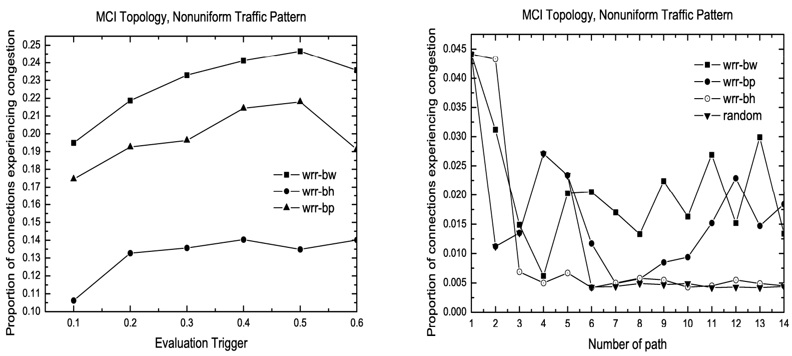
state information improves the warp of path computing increases; when a new long-lived flow is detected, signaling of its bandwidth requirement may be carried along a busy path. Figure 5(b) plots network congestion as a function of flow trigger time. Similar to Shaikh results, congestion curve shows a cup-like shape, while the reason is different: when long-lived flows are triggered at a small threshold,. the bload of admission test improves, but performance of the shortest path decreases for more traffic and more bursty flows; if flow trigger is high a few of flows will be identified as long-lived, some of long-lived flows will be transferred along non-shortest paths, this improves the possibility of network congestion and leads to higher link state variation level. According to definition of bursty flow, we suggest relating flow trigger with link-state update period, and we emphasize that the optimal value of flow trigger should be decided to the state of implementation environment.

Although above results are gained from single feature classification, we argue that it reflects some of the essentials of two or more feature classification. As two or more features are adopted in flow classification to balance requirements between network performance, computing overhead and routing stability, one feature may be chosen as the fundamental feature for multi-criterion classification. According to discussion in previous sections, active time could be the basic metric in flow classification, and performance characteristics under different trigger time forms the fundamental part of performance characteristics in multi-criterion classification.

4.5 Tuning for performance

In order to find way of performance optimization, we experiment with some of key parameters.

As the “random” policy takes any path with no preference, figure 6(a) plots relationship between path quality evaluation trigger and network performance of the other three polices. It is illustrated that network congestion decreases with increase of evaluation frequency. All three policies show good performance with



(a) Choice of path quality evaluation trigger (b) Effect of the number of path

Figure 6: **Tuning for Performance**

trigger 0.1(the number of links announcing new state reaches 10% of the number of links), but we do observe that the computation overhead increases very quickly when evaluation trigger decreases. Based on our simulation results, we recommend 0.3 for evaluation trigger.

Figure 6(b) plots how path number affects the performance of our approach with non-uniformed traffic pattern. It's evident that network performance varies with both number of path and path quality metric: using "random" and "wrr-bh", congestion decreases to a stable level after path number is larger than 6, while congestion curve of "wrr-bw" or "wrr-bp" has a cup-like shape. The reason is that, if traffic is not uniformly distributed there may be a lot of resource idled. Starting from the shortest path, more and more resource is introduced into traffic transmission and degree of overlapping improves. With "random" policy, bursty traffic will be spread onto k paths and most of long-lived flows will focus onto the shortest path gradually; once the amount of transmission capacity exceeds what is needed, network congestion will be lowered quickly to its minimum value which is caused by bursty flows and could not be removed by additional path. "wrr-bh" achieves this result more quickly by spreading traffic according to resource status on each path. By contrast, "wrr-bw" and "wrr-bp" spread traffic according to end-to-end path quality; when a new path does not share many links with existing paths, network performance will be improved; but, when the number of path exceeds some threshold overlapping will make some of the paths share the same bottleneck link, means constructive interference between flows will be prick up and network performance degrades. Referring to our experiments, if adaptive ability is required discrete value 1, 4 and 7 can be the candidates for the number of path.

5. Conclusion

Focusing on routing Internet traffic according to its built-in properties, this paper proposed a multipath routing scheme which routes incoming traffic proactively over network. In this scheme, network traffic is dispersed at flow level onto multiple paths connecting source and destination. The routing paths are established as LSPs that are not required to be totally disjoint. Path quality is used as reference in traffic dispersion and weighted round robin is used as path selection algorithm. Four types of path quality evaluation methods are proposed and experiments show that "wrr-bp" may be the best policy fitting requirements of TE&QoS. To optimize network performance, long-lived flows on non-shortest path are detected and migrated to the shortest path if their QoS could be guaranteed there. Two or more feature flow classification method is preferred while policy relating flow trigger with link state update period is investigated. Simulation results demonstrate that, our approach can improve network performance significantly under various network environments, and its performance can be optimized by tuning of key parameters.

Reference

1. R. Guerin, A. Orda and D. Williams, "QoS Routing Mechanisms and OSPF extensions (QOSPF)", Proceedings of 2nd Global Internet Miniconference (joint with Globecom'97), November 1997
2. E. Gustafsson and G.Karlsson, "A literature survey on traffic dispersion", IEEE Network, pp28-36, 1997
3. N.F. Maxemchuk, "Dispersity routing", Proceeding of ICC'75, pp.41.10-41.13, 1975
4. N.F. Maxemchuk, "Dispersity routing on ATM networks", Proceeding of IEEE Infocom, Vol.1 pp347-357, 1993
5. C. Villamizar, "OSPF Optimized Multipath (OSPF-OMP)", Internet Draft, 1998
6. C. Villamizar, "MPLS Optimized Multipath (MPLS-OMP)", Internet Draft, 1998
7. S. Vutukury and J. J. Garcia-Luna-Aceves, "A simple approximation to minimum-delay routing", Proceeding of ACM Sigcomm'99, pp227-238, 1999
8. X. Sun and G. Veciana, "Dynamic multi-path routing: asymptotic approximation and simulations", Proceeding of Sigmetrics, 2001
9. A. Shaikh, Jennifer Rexford and Kang G. Shin, "Load-senstive routing of long-lived IP flows", Proceeding of ACM Sigcomm'99, 1999
10. D.Sidhu, S.Abdul and R.Haiy, "Congestion control in high speed networks via alternative path routing",High Speed Networks, Vol.1 ,No.2, 1992
11. S.Bahk and M.E.Zaiki, "A dynamic multi-path routing algorithm for ATM networks", High Speed Networks, Vol.1, No.3, 1992
12. S.A.Lippman, "Applying a new device in the optimization of exponential queuing systems", Operations Research, Vol.23, No.4, 1975
13. S.Sibal and A. Desimone, "Controlling alternate routing in general-mesh packet flow networks", Proceeding of Sigcomm'94, 1994
14. V. Nguyen, "On the optimality of trunk reservation in overflow processes", Probability in the Engineering and Informational Sciences, Vol.5, 1991
15. S. Nelakuditi and Zhi-Li Zhang, "On selection of paths for multipath routing", In *Proc. IWQoS'01*, June 2001
16. E. Qusiros etc., "The K shortest paths problem", 35th IEEE Symp. Foundations of Comp. Sci., Santa Fe, 1994
17. M.Crovella and A.Bestavros, "Self-similarity in world wide web traffic: evidenceand possible causes", IEEE/ACM Trans. On Networking, Vol.5, Dec. 1997
18. K.Papagiannaki, N.Taft etc., "On the feasibility of identifying elephants in internet backbone traffic", Sprint ATL Technical Report TR01-ATL-110918, 2001