TRAFFIC ENGINEERING WITH REROUTING IN DIFFSERV MPLS NETWORKS

K.K. Phang, T.C. Ling, T.F. Ang
Faculty of Computer Science and Information Technology,
University of Malaya, Malaysia
Email: { kkphang, tchaw, angtf}@um.edu.my;

ABSTRACT

An important aspect of traffic engineering is the re-optimization of network resources. In this process, traffic over a congested path is rerouted to a new path to improve efficiency in network resource utilization. The aggregation nature of DiffServ MPLS networks often gives rise to long-lived flows within the network. A rerouting mechanism that supports the tearing-down of old connections and the setting-up of new connections is essential. In the absence of a rerouting mechanism, long-lived flows will be tied to inefficient paths for an extended period of time. A simple non-shortest path rerouting algorithm, namely SimpleReroute (SRR), is proposed. The simulation results indicate that SRR exhibits a lower packet loss ratio. SRR is able to reduce packet loss up to about 15 orders of magnitude, i.e. reducing the packet loss percentage from as high as 14.9% to 1% when the network is heavily congested.

Keywords: Routing, Rerouting, MPLS, DiffServ, QoS

1.0 Introduction

Quality of service is one of the most actively studied open issues in communication networks [1]. It is impossible to satisfy conflicting requirements such as maximizing network resources utilization while at the same time maintaining the maximum number of connections. One solution is load balancing through traffic engineering whereby traffic over a congested path is routed over an alternative feasible uncongested path. This paper undertakes a detailed study of a dynamic rerouting mechanism to achieve better quality of service in Differentiated Service (DiffServ) Multiprotocol Label Switching (MPLS) networks [2].

The dynamic nature of IP networks implies that the network states are always changing. Routing information is updated periodically based on the current state of the network to take advantage of the current available network resources. More current, up-to-date routing information will be recorded in the routing table so as to support more efficient routing. The problem with long-lived connections is that these long-lived connections are not able to make use of the up-to-date routing information. Packets are transmitted using out-dated routing information created during connection set-up time. Long-lived flows based on obsolete routing information are inefficient. For instance, these connections are congested while abundant bandwidth is available elsewhere. A rerouting mechanism that supports the tearing-down of old connections and the setting-up of new connections is essential. In the absence of rerouting mechanism, long-lived flows will be tied to inefficient paths over an extended period of time. Congestion may build up. Instead of passively waiting for packets to get dropped when the links become congested, a more proactive measure should be taken. A portion of the traffic should be routed away from links with higher link utilization to avoid congestion and provide better load balancing. [3, 4, 5]

In this paper, a non-shortest path rerouting algorithm, namely SimpleReroute (SRR), is proposed. SRR uses Traffic Engineering OSPF with Mix Metrics (TEOSPF-Mix) [6] which is an enhanced version of Dijkstra's algorithm [7, 8] with a path computation mechanism which improves the trade-off between hop-count and bandwidth optimization. SRR reroutes packets in congested path to alternate paths in reaction to congestion feedback information.

The organization of the paper is as follows. The literature survey and related works are presented in Section 2. Section 3 discusses the proposed mechanism, while Section 4 presents the simulation results and the performance analysis. Finally, Section 5 concludes the paper with future work.

2.0 Literature Review

The root of alternate routing algorithms can be traced back to the works of Ash et al. and Gibbens et al. [9, 10] on dynamic alternative routing (DAR). In their proposals, if a direct connection link becomes unavailable, a random node will be selected. The available bandwidth of the link through this node must be above the trunk reservation threshold. The work of Mitra et al. [11] proposed the use of a variation of DAR where a routing decision is made by considering the load on each path.

Patek et al. [12] proposed an alternate routing mechanism, namely Simple Alternate Routing (SAR), which supports DiffServ [13, 14]. This mechanism assigns portions of unmarked packet flows to alternative feasible paths in response to congestion feedback information. SAR allows the setting up of a tunnel between the ingress and egress border nodes. This tunneling mechanism only involves border routers and traffic is treated in an aggregated manner.

Oliveira et al. [15] proposed preemption policies with the aim of minimizing rerouting caused by preemption. The proposed V-PREPT is a heuristic preemption policy that takes into consideration of 1) the number of LSPs to be preempted 2) the priority of LSPs to be preempted, and 3) the amount of bandwidth to be preempted. However, the proposed mechanism is too complex and scales poorly.

Mohamed and Elsayed [16] proposed the Distributed Explicit Partial Rerouting (DEPR) scheme for load balancing in MPLS networks. DEPR uses downstream nodes for rerouting and runs on a pure MPLS network. To avoid over consumption of network resources, DEPR limits the maximum length of any partial rerouting path to not more than 4. Details of the path recomputation mechanism over non-shortest paths are not discussed in the paper.

Li et al. [17] proposed a load balancing algorithm, namely Optimal Dynamic Load Balance (ODLB). ODLB relies on the existence of parallel routes from the same source to the destination. ODLB improved bandwidth usage and reduced rerouting frequency of ongoing traffic. However, the proposed mechanism is too complex and does not scale well.

Zhu et al. [18] proposed an adaptive preemption scheme to minimize rerouting in MPLS networks. During link failure the scheme selects and preempts lower priority LSPs to accommodate the new high-priority LSP setup request. Simulation results showed that the scheme improves network resource utilization while minimizing LSPs' rerouting.

Zhang et al. [19] proposed a MPLS protection switching scheme based on label distribution protocol (LDP), such that when network failures happen, no route re-computation is required. Compared with traditional rerouting strategies in IP networks, the scheme reduces the failure recovery time of rerouting and has the advantage of more efficient network resource utilization. This scheme scales well but may suffer from the problem of obsolete routes.

Hock et al. [20] proposed the Fast Reroute (FRR) scheme. FRR can repair failures by pre-computing shortest backup paths around network elements. FRR uses unique shortest path instead of the usual shortest path in rerouting. However the use of obsolete backup path and shortest path may limit the effectiveness of the proposed scheme.

3.0 The Proposed Simple Rerouting Mechanism

A traffic congestion management scheme, namely SimpleReRouting (SRR), is proposed to reroute packets over alternative non-congested LSP. Using SRR, a portion of the LSP in the long-lived connection (say 25%) can be re-assigned or rerouted to alternate paths in reaction to congestion feedback information.

The SRR mechanism consists of the following components:

- Congestion detection (within a link).
- Alternative paths searching.
- Traffic reallocating.

SRR is a straightforward rerouting algorithm for long-lived flows and requires only minor modifications and enhancements to current MPLS DiffServ networks. SRR uses dynamic and automatic processes. Parameters in SRR such as the update hold down timing and rerouting percentage can be fine-tuned by network administrators.

The proposed SRR scheme is different from DAR. DAR blocks a call when congested whereas SRR reroutes congested traffic away from a bottleneck path. Contrary to the per-flow approach of DAR, SRR reroutes traffic in an aggregated manner. Furthermore, in SRR no tunnelling is involved whereas SAR uses a tunnelling mechanism. SRR takes advantage of the LSP mechanism in the MPLS network.

SRR differs from DEPR where upstream nodes, instead of downstream nodes, are used in the setup of the rerouting path by taking advantage of the LSP protocol. DEPR runs in pure MPLS networks whereas SRR runs on DiffServ MPLS networks to take advantage of the aggregated nature of DiffServ networks.

SRR differs from previously proposed mechanisms and algorithm for fault restoration and spare capacity allocation during link failure. [21, 22, 23]. These studies include more complex mechanisms such as the evaluation of path or partial path restoration protocols and backup route selection algorithms. SRR, on the other hand, can be considered as a congestion management mechanism.

3.1 Simple Re-Routing mechanism: A Case Study

Fig. 1 depicts the need for rerouting of long-lived flows in a network.

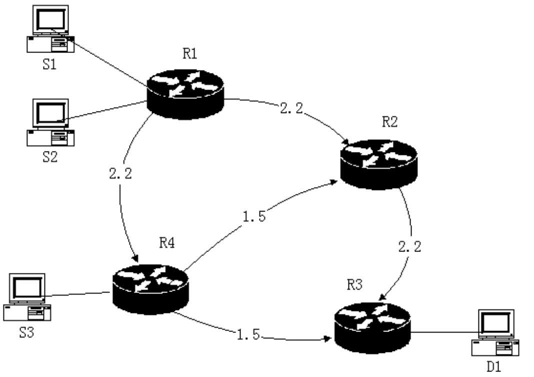

Fig. 1: Simple Topology used to demonstrate a rerouting mechanism

Sources S1, S2 and S3 send data to D1 based on the time interval indicated in Table 1.

Table 1: Traffic characteristic and schedule

Start time	**Source to Destination**	Rate (Mbps)	Duration (s)
1 sec	S1 to D1	1	120
2 sec	S2 to D1	1	5
3 sec	S3 to D1	1.5	120

Case 1: TEOSPF-Mix without rerouting mechanism

Table 2**:** Path selected by the TEOSPF-Mix without rerouting mechanism

Start time	**Source to Destination**	Rate (Mbps)	Duration (s)	Paths
1 sec	S1 to D1	1	120	R1-R2-R3
2 sec	S2 to D1	1	5	R1-R4-R3
3 sec	S3 to D1	1.5	120	R4-R2-R3

From Table 2, at the 1st second, S1 sends data at a rate of 1 Mbps to D1. Two possible paths are available from S1 to D1: path R1-R2-R3 and path R1-R4-R3. Since the former has a higher available bandwidth (2.2), path R1-R2-R3 is selected (Fig. 2).

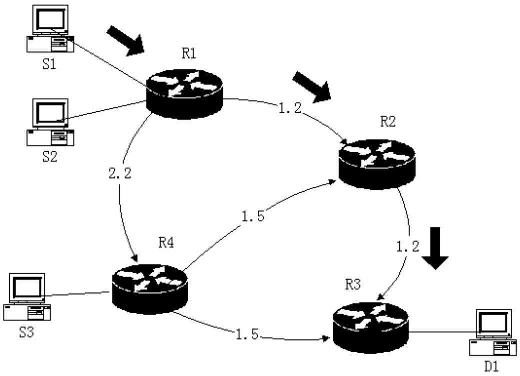

Fig.2**:** Path from S1 to D1: R1-R2-R3

The available bandwidth of path R1-R2-R3 is reduced to 1.2. At the 2nd second, S2 sends data at a rate of 1 Mbps to D1. Three possible paths are available from S2 to D1: path R1-R2-R3, R1-R4-R3 and R1-R4-R2-R3. Since R1-R4-R3 has a lower hop-count and higher available bandwidth (1.5 Mbps), path R1-R4-R3 is selected (Fig. 3). The available bandwidth of path R1-R4-R3 is now 0.5 Mbps.

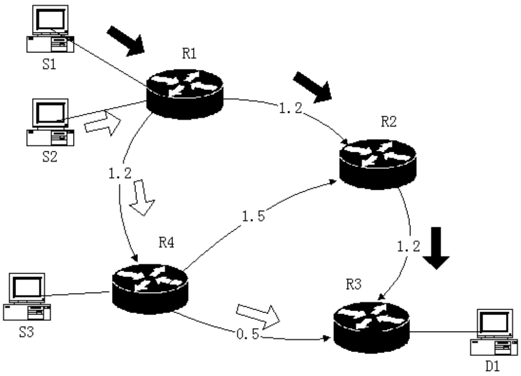

Fig. 3: S2 to D1 using path R1-R4-R3

At the 3rd second, S3 sends data at a rate of 1.5 Mbps to D1. Three possible paths are available from S2 to D1: path R4-R1-R2-R3, R4-R2-R3 and R4-R3. TE-OSPF-Mix will select path R4-R2-R3 (Fig. 4), as the available bandwidth is highest (1.2 Mbps). Since the path is unable to support the full transmission rate of 1.5 Mbps, there will be an average packet drop of 0.3 Mbps for traffic from S3 to D1.

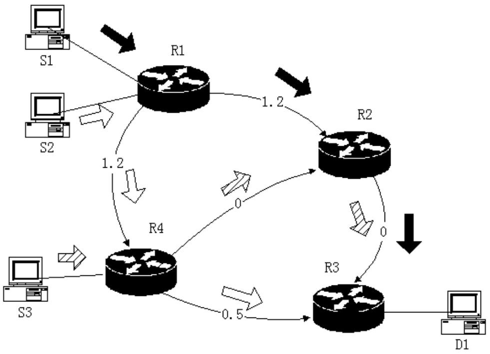

Fig. 4: S3 to D1: path R4-R2-R3

Path R1-R4-R3 is released at the 7th second. This path should be made available for traffic from S3-D1. However, without a proper rerouting mechanism, traffic from S3 will continue to take the congested path R4-R2-R3 until all traffic is sent. This is clearly inefficient.

Case 2: TEOSPF-Mix with rerouting mechanism

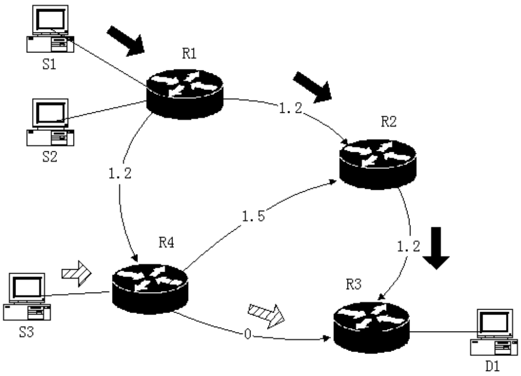

Fig. 5: Traffic from S3 to D1 is rerouted over the path R4-R3

Path R1-R4-R3 is released at the 7th second by traffic from S2. Since the path from S3 to D1, that is, R4-R2-R3, is congested, rerouting is triggered and an alternate path R4-R3 is selected (Fig. 5). Traffic is then rerouted over the new path. The new path is able to accommodate the entire 1.5 Mbps rate. Therefore, no packet loss will be experienced by the traffic from S3 to D1 from the 7th second onwards.

SRR is intended for a network that uses DiffServ MPLS mechanism. SRR is only applicable to the Assured Forwarding (AF) and Default (DE) traffic but not Expedited Forwarding (EF). In the case of EF traffic, the issue of long-lived flow does not arise. EF traffic uses explicit on-demand source routing. More formal definition of SRR is presented in section 3.2 and 3.3.

3.2 Simple Re-Routing Control Mechanisms

The LSR routing table is maintained by an underlying link state routing protocol, such as OSPF or IS-IS. The link state routing protocol will update routes periodically. SRR depends on the MPLS feedback mechanism to provide the congestion level of every directly connected link on each router. The congestion detection algorithm implemented in the SRR is listed as follows:

Congestion detection algorithm

- LSP maintenance
    - o Checking for idle LSP
    - o Checking for congestion (rerouting trigger)
        - Every LSR checks for congestion in every link periodically (at a configurable window period). If the link utilization exceeds a configurable threshold (e.g. 95%), a reroute event is triggered.

The LSRs are required to periodically collect congestion information to facilitate subsequent redirection of traffic flows. Each LSR periodically examines the congestion state of all the LSPs. SRR enables LSR to compute a better alternate path, if it exists, in response to the congestion feedback.

An alternative approach is to use probe packets [12]. Probe packets are transmitted to other LSRs to determine the congestion level on all existing paths. However, probe packets are difficult to implement in the DiffServ MPLS environment because of its aggregated nature. Furthermore, care must be taken to provide greater priority to the probe packet. Often, even the prioritized probe packet may have difficulty in getting transmitted during network congestion. Major modifications to the signalling mechanisms are required in order to incorporate probe packets in the DiffServ MPLS network.

The following shows the LSP rerouting algorithm:

LSP rerouting algorithm

```
constantly checking for congestion in each link

When a reroute event is triggered, check to see whether there
is a different (better) outgoing path for current LSPs

If there is a better path, then
      request MPLS label from the new downstream
Else if not ingress,
      send a reroute request to upstream
Else ignore the reroute event

When a label mapping is assigned by the new downstream,
release the old label and switch to the new path

When receive a reroute request from downstream, perform as if
a reroute event is triggered (the step described above)

If a better alternate path exists (computed using TE-QOSPF-
Mix) then establish the path

Reroute a fraction (configurable, e.g. say 25%) of all
affected LSPs away from the congested path
```

Consider the network in Fig. 6. There are many long-lived flows (LSP1 to LSP6) within the path R1-R2-R3-R4. The link utilization of Link L3 is above a given threshold and is considered congested. The rerouting mechanism in R3 is thus triggered (Fig. 7).

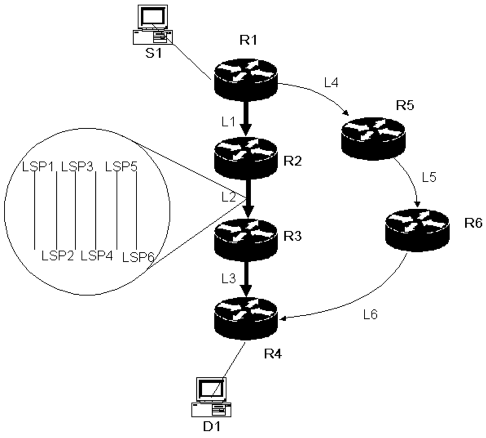

Fig. 6: Congestion in Path R1-R2-R3-R4

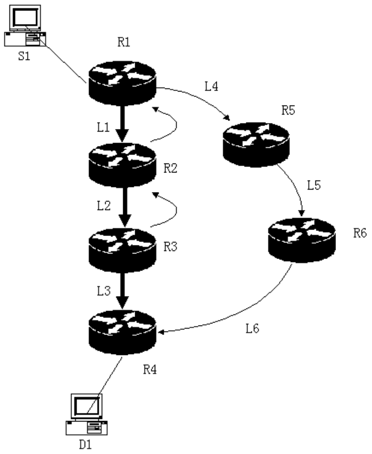

Fig. 7:Triggering of Rerouting Mechanism

If no better path can be found, R3 sends a reroute trigger to the upstream LSR, which is R2. When R2 received a reroute request from R3, R2 will perform a reroute event. If no better path can be found by R2, R2 will then send a reroute trigger to R1. R1 finds a better path to R4 via R5. The new MPLS label request process will start. R1 requests a label from the new downstream R5. R1 then reroutes, say, 50% of the affected LSPs (LSP2, LSP4, LSP5—randomly selected) to the alternate path (R1-R5-R6-R4). The remaining LSP will continue to send data over the old path (R1-R2-R3-R4) as shown in Fig. 8. The original path is thus relieved of congestion.

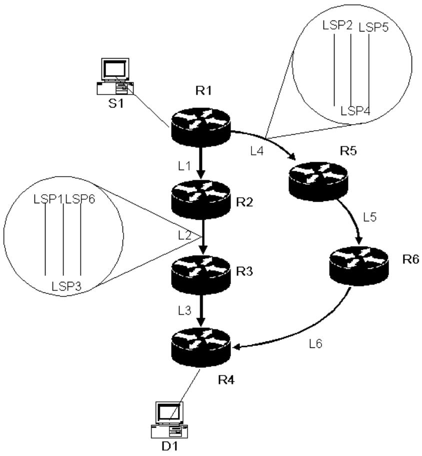

Fig. 8: 50% of the LSPs are rerouted

3.3 Design of the Rerouting Mechanism

There are two main concerns when designing the rerouting mechanism within a DiffServ MPLS network. The first is the problem of simultaneous triggers and the second is the handling of transient packets. Multiple triggers may be activated among the LSRs along the congested path. These simultaneous triggers will create unnecessary loops. Uncontrolled triggers may also create high overhead of control traffic within the network.

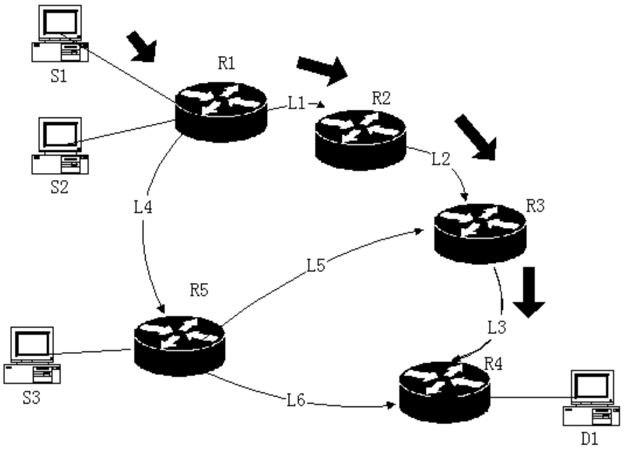

Fig. 9: Topology to illustrate multiple triggers

Consider the network diagram in Fig. 9. Source S1 sends data along LSP1 (R1-R2-R3-R4) to destination D1. The link utilization of both links L2 and L3 are high and above the rerouting threshold.

Scenario 1: (Triggered by congestion in Link L3)
Congestion in Link L3 triggers the rerouting mechanism in R3. R3 is unable to find a new feasible path. R3 then triggers R2 to reroute. Since R2 is unable to find a new feasible path, a new trigger is sent from R2 to the upstream R1 to reroute.

Scenario 2: (Initiated by congestion in Link L2)
Congestion in Link L2 triggers the rerouting mechanism in R2. R2 is unable to find a new feasible path. R2 then triggers R1 to reroute.

If both scenarios 1 and 2 happen simultaneously, both rerouting actions will be executed. Undesired effects such as inconsistency and loop may occur. A hold-down timer is implemented to avoid this problem. In scenario 1, as soon as congestion in link L3 triggers the rerouting mechanism, all routers along LSP1 will enter into the *reroute state*. The hold down timer is started. In this state, rerouting update can only be triggered by the downstream LSRs. No other rerouting can be triggered due to link congestion until the expiry of the hold down timer. In this way, the issue of multiple triggers is resolved.

In the event where the rerouting mechanism finds a feasible alternate path, a portion of the LSPs will be rerouted over the new path. Since traffic is sent from the source to the destination in a continuous manner, there are bound to be packets in transit along the old path (Fig. 10). These packets can be handled using the flush packet or “do-nothing” approaches.

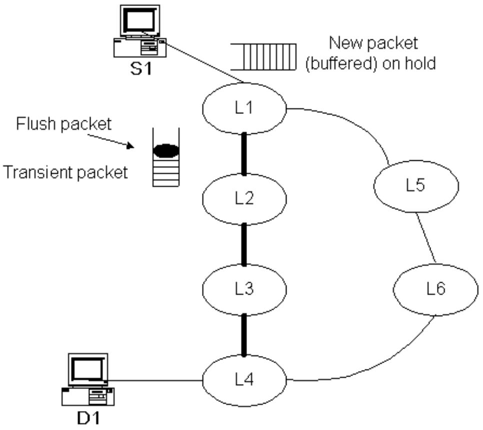

Fig. 10 : Transient Packets

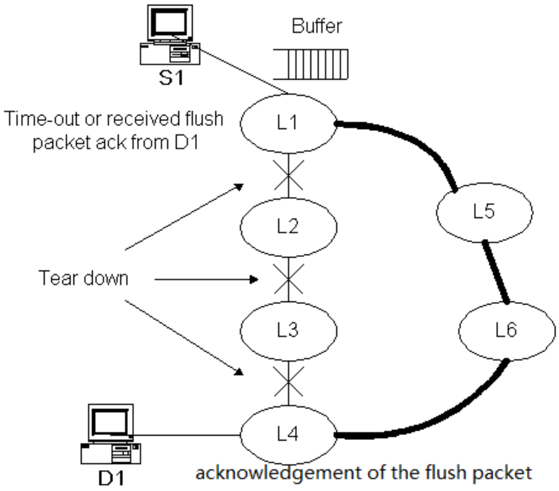

Fig. 11 : Flush packet approach

The flush packet approach uses the following mechanism (Fig. 11):

1) Add a flush packet at the end of the transient packet.
2) Hold all incoming packet at the ingress of the new alternate path. Wait for the flush packets to get to the destination.
3) The incoming packets are released into the new path after the ingress router (of the new path) receives the acknowledgement of the flush packet from the destination.
4) The old path is torn down.

In the “do-nothing” approach, transient packets are ignored. No mechanism is used to explicitly drop these transient packets. The routers along the old path will discard these packets automatically. This is because the route entry for the old path no longer exists after the path had been torn down by SRR.

To a certain degree, the flush packet approach can avoid packet loss, provided that the buffer is large enough. However, this approach has many drawbacks.

- Complex mechanism is required.
- Additional buffer space.
- May take a long time to receive the flush packet acknowledgement; system may experience buffer overflow and packet drop is unavoidable.
- A timer is needed. Time-out interval is difficult to estimate. If the flush packet is not acknowledged after a certain time interval, the system may need to resend the flush packet or issue a time-out.

The drawback of the “do-nothing” approach is that some packets will be dropped during the initial state of rerouting. However, packet drop is justifiable. Without proper rerouting mechanism, as the link becomes more congested, packet drop is inevitable.

The advantages of the “do-nothing” approach are as follows:

- Simple mechanism.
- No additional buffer is required.
- No delay in retransmission of the packets over the new route.

SRR adopts the “do-nothing” approach due to the following reasons:

- Scalability issues: For autonomous system (AS) with large number of flows, the flush packet approach is not scalable. The system has to handle the transient packets of many flows at the same time.
- Simplicity of implementation: No additional buffer is needed and no additional mechanism is required. This is in-line with the design philosophy of the DiffServ MPLS network.
- Congestion: The flush packet approach may have difficulty in sending the high priority flush packets to the destination using the congested (old) path.
- Retransmission: High-level protocols such as TCP can be used to request for the re-transmission of lost packets. AF and DE traffic can tolerate some form of packet loss.

4.0 SIMULATION ENVIRONMENT AND RESULTS

In this section, the proposed TE-QOSPF-Mix algorithm with Simple Re-Routing mechanism is evaluated. The topology used is the MCI topology (Fig. 12) which has been used in many network performance studies [24]. The evaluation metric is the packet loss ratio based on the AF and DE traffic. EF traffic will not be evaluated since the rerouting mechanism is only applicable to the AF and DE traffic. Table 3 is the simulation plan.

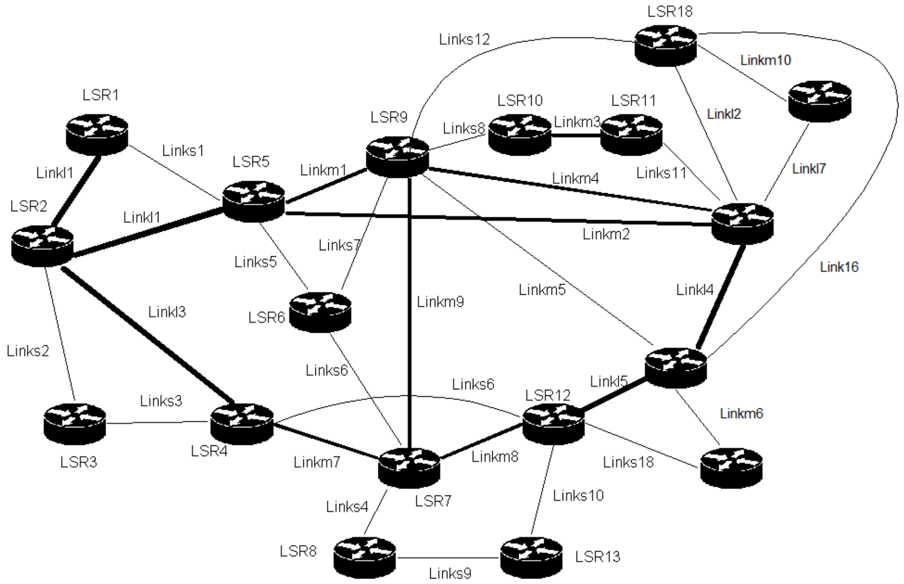

Fig. 12 :The MCI Topology

Table 3 **:**Simple Re-Routing -- Simulation Plan

Topology	**Traffic**	**# of Nodes**	**# of Links**	**# of CBR Source**	**# of VBR Source**	**Evaluation Metric**
MCI	EF, AF, DE	18	32	50	250	Packet loss ratio

The CBR traffic parameters are as follows:

- Bit Rate: 0.2 Mbps
- Amount to be sent: 2 Mbit
- Delay between call: 3 seconds

Traffic is sent continually with a delay between call of 3 seconds. In every call, each source will send exactly 2 Mbit of data to a random destination. Simulation starts with a normalized load of 1. After each run, the load is increased at a rate of 10% by increasing the “number of bits to be sent” of each source by 10%.

The VBR traffic used for this simulation consists of the following parameters.

- Bit Rate: 1 Mbps
- Mean Burst Length: 5000 µsec
- Mean interval between burst: 15000 µsec
- Number of bits to be sent: 2 Mbits
- Delay between call: 3 seconds
- ON-OFF model with Poisson distribution

Table 4 : Simulation Details

Formula	**Details of the Formula**
1	Mix (k := 1, l :=2) without rerouting
2	MixT90R25 (k := 1, l :=2) with 90% threshold and reroute 25%
3	MixT95R25 (k := 1, l :=2) with 95% threshold and reroute 25%
4	MixT95R50 (k := 1, l :=2)with 95% threshold and reroute 50%

Table 5 : Packet Loss Ratio(%)

Normalized Load	**Mix**	**MixT90R25**	**MixT95R25**	**MixT95R50**
1	0.000	0.000	0.000	0.000
2	0.027	0.001	0.000	0.000
3	0.358	0.000	0.000	0.000
4	2.166	0.014	0.007	0.007
5	5.431	0.201	0.091	0.033
6	10.498	0.525	0.300	0.602
7	14.954	2.001	1.001	2.211

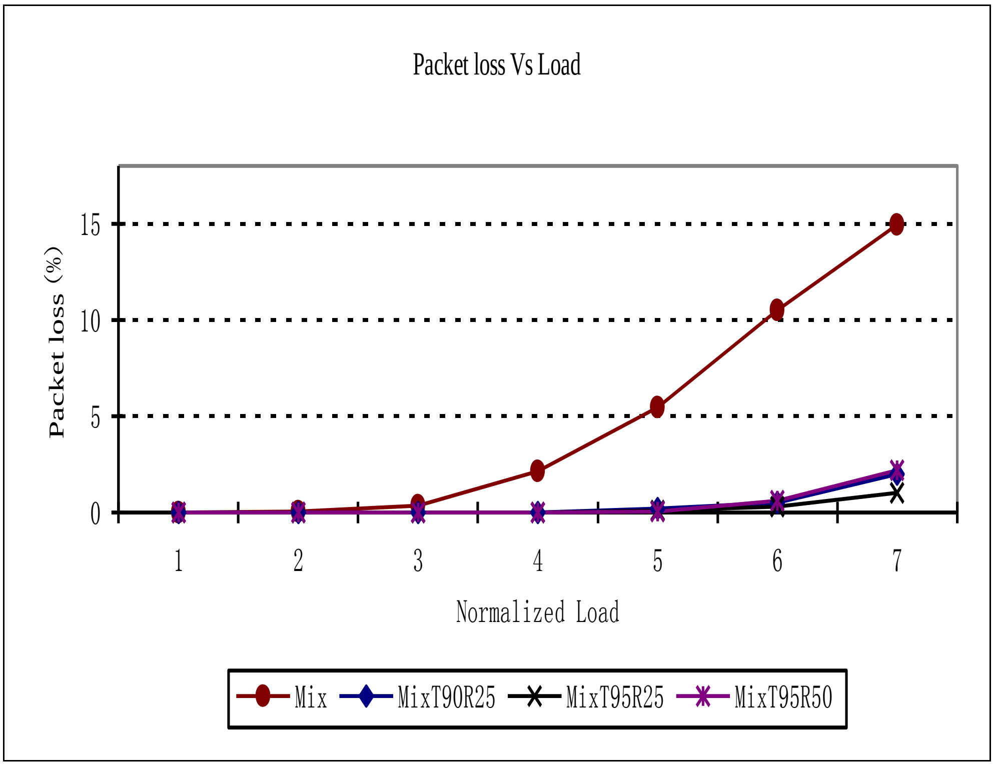

Fig. 13 : SRR and Packet Loss Ratio

Packet Loss Ratio and Performance Analysis

As depicted in Table 5 and Fig.13, among *Mix*, *MixT90R25*, *MixT95R25* and *MixT95R50*, *MixT90R25* has the best performance. The *T* and *R* in the formula can be interpreted as follows: For instance, in the case of *MixT90R25*, rerouting is triggered when the link utilization is above 90% and 25% of the LSPs within the congested path will be rerouted. On the other hand, with *MixT95R25*, rerouting is triggered when the link utilization is above 95% and 25% of the LSPs within the congested path will be rerouted.

Three stages are defined to aid the discussion: light, moderate and heavy. The traffic is light when the normalized load is 1, 2 or 3. It is moderate when the normalized load is 4 or 5 and heavy when the normalized load is 6 or 7.

When the load is light, two of the algorithms with rerouting, i.e., *MixT95R25* and *MixT95R50* perform the best, they experience zero packet loss. *MixT90R25* and *Mix* experience maximum packet loss of 0.001% and 0.358% respectively. The results at this stage indicate that rerouting is useful even when the traffic is light.

At stage 2, when the normalized load is moderate, *MixT90R25*, *MixT95R25* and *MixT95R50* have the best performance with a maximum packet loss of 0.201%. In terms of performance, *MixT95R50* ranked first. However, *Mix*, the algorithms without rerouting capability, begins to show sign of over utilization of network resources. The packet drop is as high as 5.4%.

At stage 3, when the normalized load is heavy, *MixT95R25* outperforms the rest. The ranking of *MixT95R50* drops from the first position to second. One possible explanation of the drop of performance of *MixT95R50* is that, as the network becomes more congested, the available bandwidth of more links becomes smaller. Rerouting of half (50%) of the LSPs within the congested path may overload the non-congested paths. At this stage, the gap between algorithms without rerouting and algorithms enhanced with rerouting continues to increase. The packet drop of *Mix* is 14.954% while the packet drop of *MixT95R25* is only 1.001%. The difference is about 15 folds. (1.500%).

In terms of overall performance, from Fig. 13, routing algorithms enhanced with SRR have the best performance when the rerouting threshold is set to 95% and the percentage of LSPs to be rerouted is set to 25%. The algorithm performs better with a higher value in the rerouting threshold of 95% instead of the lower value of 90%. This is because the path selected by *Mix* is efficient in terms of hop-count and available bandwidth. The setting of the rerouting triggers to a lower value such as 90% may result in higher percentage of bandwidth being reserved. The remaining 10% of the bandwidth is not put into efficient use. On the other hand, the percentage of LSPs to be rerouted is set to as low as 25%. This percentage of the traffic to be rerouted should not be too high. If a high percentage of the LSPs of the original path are rerouted to alternate route, this may cause congestion in the longer alternate route, and resulting in a lower link utilization rate in the original path. When the next trigger is activated, the traffic is shifted back to the original path. This may cause routing instability. The simulation results show that a smaller rerouting percentage such as 25% is more desirable.

In general, simulation results show that algorithms equipped with SRR mechanism have a lower packet loss ratio. The improvement can be as high as about 15 orders of magnitude when the network is heavily congested.

5.0 Conclusions

In this paper the problem of long-lived connections is tackled by the use of a simple rerouting mechanism, namely Simple Re-Routing (SRR), to route part of the LSPs in a congested path to a non-congested path.

The advantages of incorporating the SRR rerouting mechanism are as follows:

- Dynamic Traffic reallocating.
- Traffic is reallocated in an aggregated manner.
- Simple mechanism and no additional signalling mechanism are required.

SRR reroutes congested long-lived aggregated flows when better paths are available. Simulation results revealed that SRR is able to effectively reroute packets away from congested hot spots and significantly reduce packet loss in bottleneck links. This rerouting mechanism is simple and scalable and is in-line with the implementation philosophy of the DiffServ and MPLS networks.

References

.

[1] T.C. Ling, K.K. Phang, T.F. Ang, G.P. Lim, “Multi-Area Qos Provisioning Using Hierarchical Bandwidth Brokers Architecture”. *Malaysian Journal of Computer Science*, 2008, Vol. 21, No. 2, pp. 88 - 100.

[2] Tanvir, M., Said, A.M. “Decreasing packet loss for QoS sensitive IP traffic in DiffServ enabled network using MPLS TE”. *International Symposium in Information Technology* , Volume: 2, 2010

[3] Zhang, G. and Mouftah, H.T. “End-to-end QoS guarantees over diffserv networks”. *Proceedings. Sixth IEEE Symposium on Computers and Communications*, 2001, pp. 302-310.

[4] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, X. Xiao. “Overview and Principles of Internet Traffic Engineering”. RFC 3272, IETF, May 2002

[5] Ling Teck Chaw, Ahmed Riza, Phang Keat Keong, Hoh Kian Pee, “Improving QOS In WLAN Using Dynamic Weighted Fair Scheduling”. Malaysian Journal of Computer Science, Vol. 23. No. 2, 2010: 85-101.

[6] K.K. Phang, T.C. Ling, T.F. Ang. “QoS Routing in Diffserv MPLS Networks”. *Malaysian Journal of Computer Science*, Vol. 22(2), 2009

[7] Dijkstra, E. “A Note on Two Problems in Connexion with Graphs”. Numerische Mathematik, 1959, Vol. 1, pp. 269-71.

[8] Lim, S. H., Mashkuri Yaacob, Phang, K. K and Ling, T.C. “Traffic engineering enhancement to QoS-OSPF diffserv and MPLS networks”. *IEE Proceedings - Communications*, 2003.

[9] G.R. Ash, R.H. Cardwell and R.P. Murray. “Design and Optimization of Networks with Dynamic Routing”. *Bell System Technical Journal*, 60(8). 1981, pp. 1787-1820

[10] R.J. Gibbens, F.P. Kelly. “Dynamic Alternative Routing - Modeling and Behaviour”. In *TELETRAFFIC SCIENCE for New Cost-Effective Systems, Networks and Services*, ITC-12. Elsevier Science Publishers B. V. (North-Holland), 1989, pp. 1019-1025.

[11] D. Mitra, R.J. Gibbens, and B.D. Huang. “Analysis and Optimal Design of Aggregated-Least-Busy-Alternative Routing on Symmetric Loss Networks”. In *TELETRAFFIC AND DATATRAFFIC in a Period of Change*, ITC-13, Elsevier Science Publishers B. V. (North-Holland), 1991.

[12] S. D. Patek, R. Venkateswaran, J.Liebeherr. “Simple Alternate Routing for Differentiated Services Networks”. *The International Journal of Computer and Telecommunications Networking*, Vol. 37, Nov. 2001, pp. 3-4.

[13] Y. Bernet, S. Blake, D. Grossman, A. Smith. “An Informal Management Model for Diffserv Routers”, RFC3290. IETF, May 2002.

[14] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang. “An Architecture for Differentiated Services”, RFC 2475, IETF, Dec. 1998.

[15] J. C. Oliveira, C. Scoglio, I. F. Akyildiz, and G. Uhl. “A New Preemption Policy for DiffServ-Aware Traffic Engineering to Minimize Rerouting”, *IEEE Infocom*, 2002

[16] Mohamed, S.I. and Elsayed, K.M.F (2006). “Distributed Explicit Partial Rerouting (DEPR) Scheme for Load Balancing in MPLS Networks”. *11th IEEE Symposium on Computers and Communications*, 2006. Pp. 884 - 889

[17] C. Li, P. Li, T. Mohammed. “An Optimal MPLS-TE Solution to Route Selection and Redistribution on Congested Networks”. *International Conference on Networking, Architecture, and Storage*, 2007, pp. 69-76.

[18] M. Zhu, W. Ye, S. Feng, X. He. “An improved adaptive preemption policy in MPLS Networks”. *International Conference on Wireless Communications*, *Networking and Mobile Computing*, 2008, pp.1-4

[19] J. Zhang, J. Zhou, J. Ren, B. Wang. “A LDP fast Protection switching scheme for concurrent Multiple failure in MPLS network”. *International Conference on Multimedia Information Networking and Security*, 2009, pp. 259-262

[20] D. Hock, M. Hartmann, M. Menth, C. Schwartz. “Optimizing unique shortest paths for resilient routing and fast reroute in IP-based networks”. *Network Operations and Management Symposium*, 2010 , pp. 309 - 316

[21] Y. Liu, D. Tipper, P. Siripongwutikorn. “Approximation Optimal Spare Capacity Allocation by Successive Survivable Routing”. *IEEE/ACM Transactions on Networking*, Volume: 13 , Issue: 1, 2005.

[22] Philips, S. J. and Westbrook, J.R. “Approximation Algorithms for Restoration Capacity Planning”, Lecture Notes in Computer Science 1643, Springer, 1999, pp. 101 – 115.

[23] Autenrieth, A and Kirstadter, A. “Fault Tolerance and Resilience Issues in IP based Networks”. *Second International Workshop on the Design of Reliable Communication Networks*, München, April 10-12, 2000.
[24] Shaikh, A. “Efficient Dynamic Routing in Wide-Area Networks”, Ph.D. Paper, University of Michigan, May 1999.

BIOGRAPHY

K. K. Phang is an Associate Professor at the Faculty of Computer Science & Information Technology, University of Malaya, Malaysia. He obtained his PhD from University of Malaya. His current research interests include high speed computer network, network QoS, traffic conditioning, traffic engineering, and distributed systems.

T.C. Ling obtained his PhD from University of Malaya, Malaysia. He is an Associate Professor at Faculty of Computer Science & Information Technology, University of Malaya. His research areas include core network research, inter-domain Quality of Service (QoS), Voice over IP (VoIP), grid computing, cloud computing and network security.

T.F. Ang obtained his PhD from University of Malaya. He is a lecturer at the Faculty of Computer Science & Information Technology, University of Malaya, Malaysia. His research areas include web services, Voice over IP (VoIP), cloud computing and network QoS.