Reconfiguration in Multi Class of Services Networks

Robert Suryasaputra*, Alexander A. Kist† and Richard J. Harris‡
* Centre for Advanced Technology in Telecommunications (CATT Centre)
School of Electrical and Computer Engineering
RMIT University Melbourne, GPO Box 2476V, VIC 3001, Australia
Email: robert@catt.rmit.edu.au
† Faculty of Engineering & Surveying
University of Southern Queensland, Toowoomba Qld 4350, Australia
Email: kist@usq.edu.au
‡Institute of Information Sciences and Technology
Massey University, Private Bag 11 222, Palmerston North, New Zealand
Email: R.Harris@massey.ac.nz

***Abstract*— This paper presents an optimisation method to traffic engineer networks carrying different classes of traffic. The novel Mixed Integer Programming model is based on the classical multi-commodity flow problem. Although some initial input data (such as topology and traffic matrix) is required, the method can be directly applied and no protocol or router modifications are necessary. The method reconfigures the routing pattern in a way that allows the network to carry more traffic and capacity expansion can be delayed. It is shown here that the method brings up to 50% improvement in the maximum link utilisation when it is compared to Inverse Capacity metric, the default Cisco routing metric.**

I. INTRODUCTION

Traffic engineering has become an integral part of the network operations. Its aim is to facilitate efficient and reliable network performance while at the same time optimising network resource use. Intra-domain traffic engineering is concerned with optimising performance of a network which under administration of one service provider. The service provider can influence how flows are routed inside its own network by tuning network configurations. Native IP networks usually run a shortest path based protocol such as OSPF (Open Shortest Path First) or IS-IS (Intermediate Systems - Intermediate Systems). Service providers can influence the routing by setting or changing link metrics to achieve given performance objectives, e.g. minimising the maximum link utilisation or distributing the traffic load across the links as evenly as possible [1]. The problem of finding a suitable suitable link metrics has been studied extensively in [2] [3] [4] [5].

Whilst the Internet operation is still largely based on best effort delivery, service providers are now required to handle different kinds of traffic. For example, mission critical applications require close to 100% packet delivery rate and cannot tolerate delay more than a certain threshold. Real time applications such as VoIP or video streaming require minimum delay and jitter while they are more resilient to loss. On the other hand, non-time critical applications, such as email, can tolerate delay and scarce bandwidth. Hence, we can classify mission critical and real time traffic as high priority traffic, and non-critical traffic as Best Effort (BE) traffic.

The traffic engineering problems are further complicated by the requirements for networks handling multiple classes of traffic [6]. Differentiated Services (DiffServ) is developed to support Quality of Service (QoS) in a scalable manner. Multi-Protocol Label Switching (MPLS) technology can be used to deliver the traffic engineering requirements [7] [8].

In this paper, we look at a problem of reconfiguring the network carrying multiple classes of traffic. We model high priority traffic as Expedited Forwarding (EF) traffic, which requires a minimum outgoing rate to be specified. A virtual lease line, in which the ISP has to conform to Service Level Agreement (SLA), can also be categorised as high priority traffic. Non-critical traffic will be modelled as Best Effort (BE traffic). At any time, the network has to be able to carry the EF traffic which must be fully restored and BE traffic which should be restored as much as possible above a certain percentage in the event of a link failure or when an optimisation takes place on an operational network. Given that the number of networks migrating to MPLS technology, we utilise the explicit routing feature in MPLS to route EF traffic. Best effort traffic is routed using a native intra-domain protocol such as OSPF, which works based on shortest path paradigm.

We formulate a two-phase optimisation model. The first phase is to determine a suitable path for every traffic classes and every origin and destination pairs (OD-Pairs). A suitable path is a path that is able to carry traffic higher than its restoration percentage. An additional constraint is also added to impose the single path routing requirement. The second phase is to determine a suitable weight set to route best effort traffic.

The main contribution of this work is a novel Mixed Integer Linear Programming (MILP) model formulation for optimising multi-classes of traffic networks. The model is based on the classic multi-commodity flow problem. It is applicable for optimising or expansion of operational networks running IGP/MPLS. In addition, it is also shown that non-linear constraints can be transformed into linear constraints given the problem's understanding. Furthermore, the model can be directly applied to networks with MPLS support without requiring any router or protocol modifications.

The remaining of the paper is organised as follows: Section II outlines the model in detail. It also shows the transformation of non-linear constraints to linear constraints. Section III explains the experimental studies, results and discussion. Section IV concludes the paper.

II. MILP / LP MODEL FOR RECONFIGURATION

As briefly mentioned above, the model consists of two phases. The first phase (Phase I) is a path selection process for EF and BE traffic. The second phase (Phase II) is a weight set for BE traffic shortest path routing. The first subsection introduces the notation which is used throughout the paper. The second subsection describes the Phase I MILP model and the last describes the Phase II LP model.

A. Notation

Constants:

d^{tk} is the demand of traffic type t of demand k.

r^{tk} is the restoration percentage, EF traffic will have 100% restoration percentages. BE traffic will have a user defined restoration percentage.

δ_{ej}^{tk} is the boolean indicator if link e lies on path j.

Variables:

u_j^{tk} is the binary variable that indicates if path j is used to carry traffic t of OD-Pair k.

x_j^{tk} is the amount of traffic on path j carrying traffic t of demand k.

y_e is the additional capacity needs to be purchased to restore traffic above restoration level.

h_j^{tk} is the hop count of path j is used to carry traffic t of OD-Pair k.

E is the set containing edges in the network.

K is the set containing OD-Pairs in the network.

T is the set containing different traffic classes in the network.

B. Phase I: Path Allocation Model

The first phase model is an extended multi-commodity flow problem [10], with additional constraints. The basic formulation of the erestoration model is given in [9]. In the model, high restoration percentages may yield an infeasibility condition. In this work, an variable y_e that represent "additional capacity required" is used. This "elastic" capacity is heavily penalised in the objective function to discourage its use.

$$\sum_K \sum_T \sum_J \delta_{ej}^{tk} x_j^{tk} \leq c_e + y_e \qquad \text{for } e \in E \qquad (1)$$

Both EF and BE traffic carried on the corresponding paths should be at least restored above the restoration percentages.

$$\sum_J x_j^{tk} \geq r^{tk} d^{tk} \qquad \text{for } t \in T, k \in K \qquad (2)$$

Restored traffic carried on the corresponding paths should not be exceeding the end-to-end demand.

$$\sum_J x_j^{tk} \leq d^{tk} \qquad \text{for } t \in T, k \in K \qquad (3)$$

EF traffic is a special case of restoration, which can be used to simplify equation (2) and (3). In this special case, where the value of $r^{tk} = 100\%$, the equation (2) and (3) become a single equality constraint.

$$\sum_J x_j^{tk} = d^{tk}$$

It is desirable that the demands are carried on a single path. Although, load balancing (using multiple LSPs and ECMP) can be done, it is not desirable to have packets belonging to the same OD-Pair travel different path. Hence, single path requirement needs to be imposed on EF and BE traffic. To accommodate this, equations (1-3) needs to be modified as follows:

$$\sum_K \sum_T \sum_J \delta_{ej}^{tk} u_j^{tk} x_j^{tk} \leq c_e + y_e \qquad \text{for } e \in E \qquad (4)$$

$$\sum_J u_j^{tk} x_j^{tk} \geq r^{tk} d^{tk} \qquad \text{for } t \in T, k \in K \qquad (5)$$

$$\sum_J u_j^{tk} x_j^{tk} d^{tk} \leq d^{tk} \qquad \text{for } t \in T, k \in K \qquad (6)$$

$$u_j^{tk} \text{ is binary}$$
$$x_j^{tk} \geq 0$$

However, these constraints become non-linear constraints, due to a product of u_j^{tk} and x_j^{tk}. To overcome the problem, a deeper analysis is carried out and it can be shown that the relationship between x_j^{tk} and u_j^{tk} can be written differently. Consider the following inequality:

$$x_j^{tk} \leq d^{tk} u_j^{tk}$$

If u_j^{tk} is equal to zero, this indicates that the path does not carry any flow at all, the value of corresponding x must be zero. However if u_j^{tk} is equal to one, the amount of flow of the corresponding path must be non-zero and its minimum is determined by the restoration percentage and it is bounded by the d^{tk}. For a particular traffic t in OD-Pair k, there must be only one non-zero x_j^{tk} (single path requirement).

The following model is equivalent to the above non-linear problem:

$$\sum_K \sum_T \sum_J \delta_{ej}^{tk} x_j^{tk} \leq c_e + y_e \qquad \text{for } e \in E \qquad (7)$$

$$\sum_{J} x_j^{tk} \geq r^{tk} d^{tk} \qquad \text{for } t \in T, k \in K \tag{8}$$

$$\sum_{J} x_j^{tk} \leq d^{tk} \qquad \text{for } t \in T, k \in K \tag{9}$$

$$\sum_{J} u_j^{tk} = 1 \qquad \text{for } t \in T, k \in K \tag{10}$$

$$x_j^{tk} \leq d^{tk} u_j^{tk} \qquad \text{for } j \in J, t \in T, k \in K \tag{11}$$

$$u_j^{tk} \text{ is binary}$$
$$x_j^{tk} \geq 0$$

Since we want to maximise the throughput of BE traffic and at the same time not to use additional capacity, the problem becomes multi-objective optimisation. The first objective is to maximise the BE traffic throughput and the second is to minimise the cost of purchasing additional capacity to accommodate EF traffic. The third term is required to ensure that the paths with the smaller number of hop counts are preferred.

$$Max\Big(\sum_{K}\sum_{T}\sum_{J} x_j^{tk} - M \sum_{E} c_e y_e - \sum_{K}\sum_{T}\sum_{J} h_j^{tk} u_j^{tk}\Big) \tag{12}$$

A large number M in the objective function (12) is introduced to discourage traffic to use the additional capacity, unless the whole network cannot sustain traffic any longer.

On the classic multi-commodity flow problem, the feasibility limit of the problem is determined by the capacity of the network (i.e. c_e values). This means if we slowly increase the demand vector d^{tk}, a point exists where the problem becomes infeasible. However, in this modified formulation (with variable y_e being introduced), it should be noted that the feasibility condition is not restricted by the demand level.

C. Phase II: Determining a weight set (Weight Setting)

A set of link weights needs to be determine to route the BE traffic. Work in [12] states that for any arbitrary set of routes (routing pattern in our case), as long as they are not loopy, they can be converted to shortest-paths with respect to some set of positive link weights. However, this only guarantees that the path is one of the shortest paths. A set of link weights that yields to a unique shortest path, where in there is one and only one shortest path, for every demand cannot be guaranteed to exist.

A pure LP formulation [11] can be used to determine the set of link weights. Given that the routing pattern for BE traffic is known from Phase I, the problem of determining a unique shortest path weight set can be formulated as a pure LP problem as (P_n is defined as a path that carries non-zero flow based on Phase I solution):

$$\sum_{e \in P_n} w_e + 1 <= \sum_{e \in P_j} w_e \qquad \text{for } j \in J, j \neq n, t \in T, k \in K \tag{13}$$

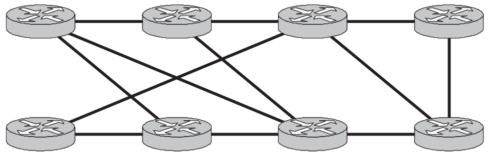

Fig. 1. Test Network Topology

$$w_e \geq 1 \qquad \text{for } e \in E$$

The constraints (13) force the path that carries the flow should (P_n) be at least one unit shorter in length than other paths belonging to that OD-Pair. However, this might not always to feasibility due to the reason stated above. By removing the “+1” coefficient in the left-hand side of constraints (13), the model becomes an equivalent method to find “non-unique” shortest path as in [12].

It should be noted that the objective function in the Phase II is not important. One could have minimising sum of link weights (as what is done in this work). Different objectives such as minimising the range of link weight that should be used in practice or minimising number of weights that need to be changed can be used.

III. RESULTS AND DISCUSSIONS

This section introduces the experimental setup, the solving procedures and discussions of the results.

A. Experimental Setup

The test network consists of 8 routers which are connected by 24 uni-directional links as shown in Fig. 1. The demand is generated from one router to every other routers in the network, hence there are 56 OD-Pairs. Every OD-Pairs will have two different traffic classes, EF traffic and BE traffic, with 20% and 80% proportion, respectively. In this experiment, we set the restoration percentage for EF and BE traffic to 100% and 95%, respectively.

The experiment is done with a number of different traffic matrix instances. There are 3123 instances of traffic matrices. They are then grouped into different 41 different groups according to their total demand, with the first group having the lowest total demand and the last group having the highest total demand. Roughly, they can be categorised as light, moderate and heavy loads. Details of how these traffic matrix elements are generated are explained in [13].

B. Solution Procedure

The MILP/LP model requires a set of paths for every demand in the network. Theoretically, all possible paths in the network should be included. However, in reality, only paths with an acceptable delay or hop count will be used. Based on this knowledge, we can generate a “limited” number of paths. For this particular network, we generate 10 paths for each demand.

Once the Phase I problem is formulated, it is then solved using *glpk, GNU LP solver*. For this problem size, the solver is able to get within 0.1% of the optimal value in less than 5 seconds.

Once the Phase I solution is obtained, the Phase II problem can be formulated. A similar approach to [11] can be adopted. To solve the Phase II problem, one can start with only two shortest paths and do an iterative computation. In the first iteration, the first path, P_n is the one that carries the flow and has the total metric less than P_j, where $j \in J, j \neq n$. Once the problem is solved, Dijkstra's algorithm can be used to determine whether there are more shortest paths that are not included in the Phase II problem. If there is one, then it will be added to the Phase II problem and it is solved again. If the inclusion of the new path causes infeasibility, this indicates that no weight system exists for this routing pattern. The iteration stops when no more shortest paths can be found.

C. Performance Evaluation

In this work, the concern is BE traffic. EF traffic is routed using MPLS and has a guarantee minimum bandwidth. Moreover, once the solution from the Phase I problem is available, ER traffic can be readily routed. Our main concern here is the maximum utilisation of the links in the network.

Figure 2 depicts the resulting maximum utilisation when BE traffic is routed based on three different routing schemes, namely, unit weight, inverse capacity and weight setting (based on the Phase II solution) with increasing total demand. The error bars on every data points indicate 95% confidence limits.

Using unit weight in this simple network, i.e. setting the weight of all links to one, will generate many multiple shortest paths for most of the OD-Pairs. In this network, the splitting mechanism is effective to spread the load through out the network. In a real network running OSPF, Equal Cost Multi Path (ECMP) functionality is used to split aggregate traffic to a particular destination evenly.

Using an inverse capacity, the Cisco recommendation, as the shortest path routing metric gives the worst performance of all. This is true across different network loads in our experiments. The utilisation reaches 100% when the average demand is 57 Mbps. With the unit weight and weight setting, the 100% utilisation is reached when the total demand is 72 Mbps. Furthermore, due to the capacity constraints in the model, the weight setting can restrict the maximum utilisation at 100%, whilst the other results 120% and 155% (highest traffic load). Variation in the maximum link utilisation in the Unit Weight and Inv Cap case is higher because none of these models have capacity restriction.

Although the unit weight may look to perform reasonably well in comparison to weight setting, a complication with multi-path routing arises because it is impossible to split traffic evenly in a general practice. Splitting can degrade TCP performance because packets belonging to the same OD-Pair may be arriving out of order (packet level splitting). Splitting can also cause problem in debugging network problem. Test packets can travel via several different paths and conceal problematic paths.

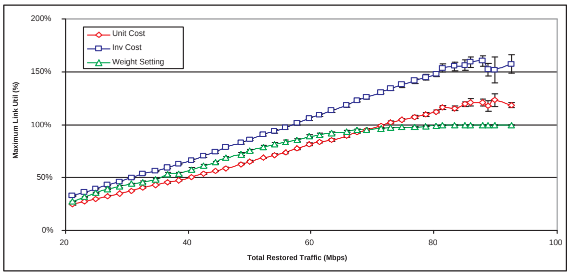

Fig. 2. Resulting Maximum Utilisation for BE Traffic Routing

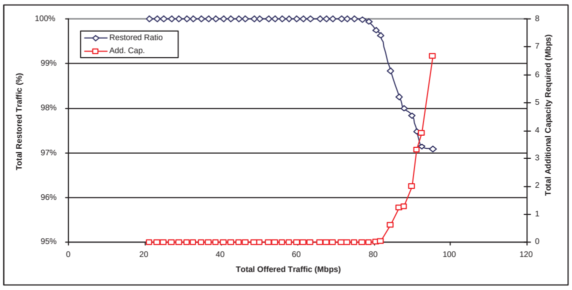

Fig. 3. Percentage of traffic restored and Additional Capacity required

One of the goals of Traffic Engineering process is to delay required capacity expansion although the traffic demand is growing without performance degradation. Herein, using weight setting, we show that the capacity expansion can be delayed for awhile. The weight setting can maintain feasibility without having have to put in additional capacity until the total demand reaches 84 Mbps (see Fig. 3 red curve). When it is required, we also show that the amount is very small in comparison to the demand that can be accommodated. The model is able to identify the bottle-neck links in the network. Upgrading these links will greatly increase BE traffic throughput.

The blue curve in Fig. 3 shows the fraction of total BE traffic restored to total BE traffic offered. Recap that BE traffic restoration percentage is set to 95%. In all scenarios, the percentages are well above the target restoration percentage. A long OD-Pair (the one that has a significant number of hops away to the destination) will be just restored based on the restoration percentage lower bound (in this case 95%) because of network bottle-necks. A short OD-Pair (the one that only has one or two hops to travel to destination) will be greatly benefited because they will be restored up to 100% on the ”non bottle neck” links.

In practice, it is impossible to have 100% utilisation or higher. Utilisation figures in this work are to be normalised with the utilisation of a running network. It should be noted that the utilisation results depicted in Fig. 2 are calculated after the additional capacity has been added. Fig. 2 and 3 are the experimental results after the data points are grouped together based on their total demand. For the “ungrouped” results, please refer to the Fig. 4 and 5.

IV. CONCLUSIONS

In this paper, we have presented a model to reconfigure DiffServ networks with two classes of traffic. MPLS technology can be used to route the high priority traffic, whereas native IP routing, such as OSPF, is used to route best effort traffic. The model maximises the best effort traffic throughput and minimises cost of expansion, whilst at the same time fulfilling bandwidth requirements for high priority traffic. It is shown that in this particular model non-linear constraints can be transformed into linear constraints. Experiment results show that the model can reduce the maximum link utilisation in the network by as much as 50% and accommodate 44% more traffic without needing capacity expansion.

Heuristics to solve MILP model could be developed to solve larger network problems. For large networks, general purpose LP solvers usually run very slow and may not find any solutions. Future work includes the comparison of heuristic method with the MILP solution generated from LP solvers for large networks.

ACKNOWLEDGMENT

The authors would like to thank the Australian Telecommunication Co-operative Research Centre (ATCRC) for their generous financial support. Contributions from Rob Palmer (former Telstra Research Laboratories) who provided the data sets for the experiment is also gratefully acknowledged.

REFERENCES

[1] A. Riedl, “Optimized routing adaptation in IP networks utilizing OSPF and MPLS,” in *Communications, 2003. ICC ’03. IEEE International Conference on*, vol. 3, 2003, pp. 1754–1758 vol.3.

[2] B. Fortz and M. Thorup, “Optimizing OSPF/IS-IS weights in a changing world,” *Selected Areas in Communications, IEEE Journal on*, vol. 20, no. 4, pp. 756–767, 2002.

[3] E. Mulyana and U. Killat, “An Alternative Genetic Algortihm to Optimise OSPF Weights,” in *15th ITC Specialist Seminar. Internet Traffic Engineering and Traffic Management*, 2002.

[4] J. Murphy, R. Harris, and R. Nelson, “Traffic Engineering Using OSPF Weights and Splitting Ratios,” in *Interworking 2002*, ser. IFIP Conference Proceedings, C. McDonald, Ed., vol. 247. Perth, Australia: Kluwer, 2003, pp. 277 – 287.

[5] A. Nucci, B. Schroeder, S. Bhattacharyya, N. Taft, and C. Diot, “IGP Link Weight Assignment for Transient Link Failures,” in *International Teletraffic Congress ITC18*, 2003.

[6] P. Trimintzios, L. Georgiadis, G. Pavlou, D. Griffin, C. F. Cavalcanti, P. Georgatsos, and C. Jacquenet, “Engineering the Multi-Service Internet: MPLS and IP-based Techniques,” in *ICT2001. International Conference of Telecommunications. Proceedings. IEEE*, 2001.

[7] F. L. Faucheur and W. Lai, “Requirements for Support of Differentiated Services-aware MPLS Traffic Engineering,” 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3564.txt

[8] E. Horlait and N. Rouhana, “Differentiated Services and Integrated Services Use of MPLS,” in *ISCC 2000. Fifth IEEE Symposium on Computers and Communications. Proceedings. IEEE*, 2000.

[9] R. J. Harris, “A Mathematical Programming Model for Service Protection in a Telecommuncaitons Network,” in *International Teletraffic Congress ITC13*, 1991.

[10] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms and Applications*, 1st ed. Prentice Hall, 1993.

[11] M. Pioro and D. Medhi, *Routing, Flow and Capacity Design in Communication and Computer Networks*, 1st ed. Morgan Kaufmann, 2004.

[12] Y. Wang, Z. Wang, and L. Zhang, “Internet traffic engineering without full mesh overlaying,” in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 1, 2001, pp. 565–571 vol.1.

[13] M. J. Dale, H. L. Ferra, and R. A. Palmer, “Fast MPLS Network Optimisation using Machine Learning,” in *TENCON 2005. 2005 IEEE Region 10 Conference. IEEE*, vol. 1, 2005, pp. 971–976.

APPENDIX

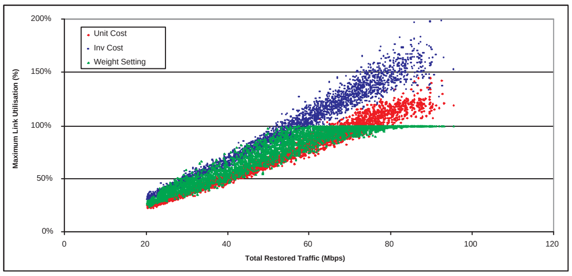

Fig. 4. Resulting Maximum Utilisation for BE Traffic Routing (Ungrouped data points)

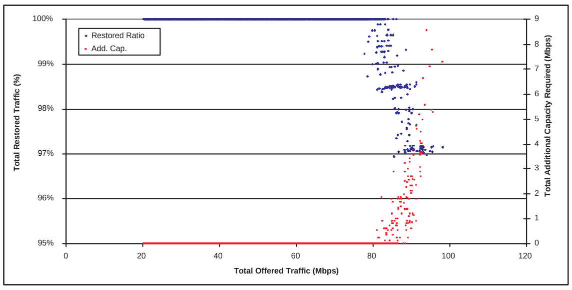

Fig. 5. Percentage of traffic restored and Additional Capacity required (Ungrouped data points)