TOPOLOGICAL DESIGN OF MULTIPLE VPNS OVER MPLS NETWORK

Anotai Srikitja
David Tipper

Dept. of Information Science and Telecommunications
University of Pittsburgh
135 N. Bellefield Avenue,
Pittsburgh, PA 15260

ABSTRACT

With the deployment of MPLS over a core IP backbone, it is possible for a service provider to built Virtual Private Networks (VPNs) supporting various classes of services with QoS guarantees. Efficiently mapping a logical layout of VPNs over a service provider network is an important traffic engineering procedure. The use of sink-tree (multipoint-to-point) routing paths in a MPLS network makes the VPN design problem different from traditional design approaches where a full-mesh of point-to-point paths often is the choice. Clear benefits of using sink-tree paths are reduction in the number of label switch paths and bandwidth saving due to larger granularities of bandwidth aggregation within the network. In this paper, the VPN design over MPLS, using sink-tree routing, is formulated as a mixed integer programming problem to simultaneously find optimal VPNs logical topologies and their dimensions to carry multi-service, multi-hour VPNs traffic from various customers. The model exploits a pre-computed set of sink-tree paths over which VPNs traffic is routed in a MPLS network. It is clearly shown that such a problem formulation yields a NP-hard complexity. Here, we propose tree selection heuristics aiming to scale the VPN design problem by choosing a small-but-good candidate set of feasible sink-tree paths. Numerical results are given showing the advantages of the proposed approach.

INTRODUCTION

Virtual Private Networks (VPNs) provide a private and dedicated environment over a shared private or public network infrastructure. The deployment of QoS-based VPNs supporting integrated services for voice, data and video applications over a public Internet is economically appealing since it allows a high-speed access with performance and Quality of Service (QoS) guarantees. MultiProtocol Label Switching (MPLS) [1,2] with traffic engineering capability allows QoS-based VPNs to be efficiently built on top of IP network architecture.

To guarantee performance to VPN services, a service provider has to be concerned with capacity provisioning and routing coexisting VPNs having different service classes and topologies over the same network infrastructure. Generally, a well designed VPN must be easy to manage while maintaining bandwidth efficiency. Over a MPLS network, this implies that the number of label switched paths (LSPs) and required labels must be kept small. In term of capacity efficiency, different granularities of traffic aggregation may be considered, for example, aggregation of traffic from different VPNs within the same Classes of Service (CoS), aggregation of traffic from same VPNs exiting at the same egress router, etc.

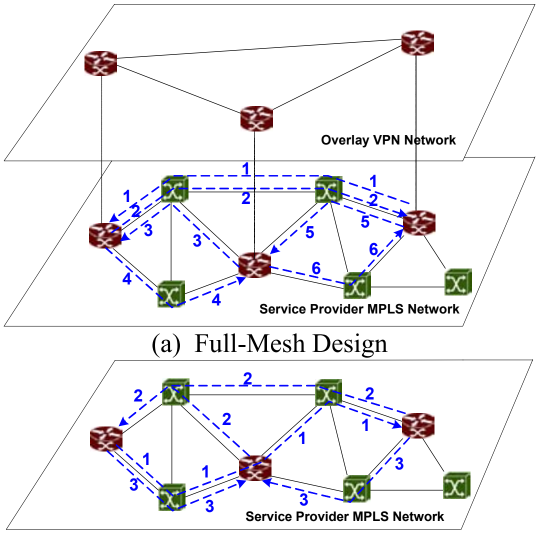
Traditionally, VPNs over a connection-oriented network such a circuit-switched or an ATM network, are often viewed as a logical full-mesh network with point-to-point demand between node pairs. More specifically, in a logical full-mesh topology, each demand pair is independently given a logical link which is a point-to-point path going through multiple switched-points. In MPLS networks, however, a route can be a point-to-point path as well as a multipoint-to-point path or a sink-tree path ending at an egress router. This attribute makes the VPN design problem different from those in traditional connection-oriented networks. Exploiting sink-tree paths, traffic demand can be merged/aggregated not only at the edge router where the traffic demand enters the network but at the core router inside the network. Overall, the use of sink-tree paths not only reduces the number of label switch paths but promotes the bandwidth aggregation within the network, thus, allowing the possibility of bandwidth savings. In this manner, traffic of different VPNs with the same QoS requirement may/may not be carried on the same routing tree and may/may not share a portion of network bandwidth. A big question is how to construct a tree and how to incorporate it in the network design model.

Only recently has work appeared on optimization models to solve traffic engineering problems over MPLS networks. In [3], authors provided integer programming formulations for flow assignment problem given a set of point-to-point LSPs and it can be extended to solve a capacity planning problem. However, several trivial assumptions are made including : one-to-one relationship between traffic trunks and LSPs, no aggregation, de-aggregation and merging of LSPs. Sato et al. [4] proposed the use of multipoint-to-point LSPs in flow assignment problem. A set of pre-selected LSPs is forced to include at least two routes which do not share any single node to each ingress/egress node. The optimization model aims to minimize the maximum link load without considering cost of link capacity.

Realizing a sink-tree routing path, this paper proposes a mathematical formulation for the problem of VPN design in order to simultaneously find optimal logical topologies of multiple VPNs and their dimensions over a service provider IP/MPLS core network. The proposed model incorporates multi-service, multi-hour design with the objective in minimizing total capacity cost. The pre-computed set of candidate sink-tree paths (multipoint-to-point paths) will be used in the model to determine a route assignment for VPNs traffic. In this paper, we have developed the proposed design model and compared it against a full-mesh design. However, the optimal solution can not be obtained within a reasonable time due to the complexity of the problem. Therefore, a tree selection heuristics is developed aiming to scale the VPN design problem by choosing a small-but-good candidate set of feasible sink-tree paths. Numerical results show the advantages of the proposed approach.

SINK-TREE LSP PATH

As previously mentioned, a label switched path (LSP) in MPLS can be a multipoint-to-point path referred to here as a sink-tree path. One clear benefit of using a multipoint-to-point path is its scalability since fewer LSPs must be created as compared to using a point-to-point path between each demand pair. This also results in a smaller number of forwarding labels needed at each MPLS router and thus makes it simpler to manage for a medium to large network. For example, applying a full-mesh design, where there are point-to-point paths between all node-pairs, to a N –node network, the total number of LSPs required is $N \cdot (N-1)$. However, this number can be reduced to N using a sink-tree design. Figure 1 displays a full-mesh versus sink-tree design for a 3-node VPN over an 8-node MPLS network. Assume that there is a directional demand of one unit between 3-node pairs in the VPN network and each link in the MPLS network has one unit cost. A full-mesh design requires 6 LSPs compared to 3 LSPs in sink-tree design. Both designs use the same links in MPLS network. However, the first yields a unit cost of 14 while, in the latter, the cost is reduced to 12. The cost saving results from the capacity efficiency gain attained when traffic is merged in a sink-tree design.



(a) Full-Mesh Design

(b) Sink-tree Design

Figure 1. Full-mesh versus sink-tree design

MPLS-VPN DESIGN MODELS

Given a network topology, node location and link capacity, an optimization model is formulated for VPN design over MPLS network. A physical network is represented by a graph $G(N, L, C)$ where N , L , C is a set of nodes, links and link capacities respectively. M ($M \subset N$) is the set of edge nodes (edge routers) where there is a demand traffic entering or exiting. Thus, $N - M$ represents the set of core nodes (core routers). The complete notation of the formulation is given below. For each link $l \in L$, utilization factor α_l limits the proportion of the link capacity C_l to be allocated for all VPNs traffic. This utilization factor may be used to protect certain links from being overly subscribed or subjected to potential congestion. For example, a smaller value of α_l may be assigned to links connecting to core-routers than ones connecting to edge-routers. Here, the complete matrix of VPNs traffic demand is assumed to be given. For each source-destination (ingress-egress) node-pair, the matrix of each VPN specifies the required bandwidth and its QoS parameters (i.e., end-to-end delay requirement and packet loss probability). In the model, traffic demand $D_{V,s,k}$ will be assigned a route based on its egress node $k \in K$, where $K \subset M$, called the demand set index.

A. Notation

α_l Maximum utilization factor of link $l \in L$

K Demand set index, $K \subset M$

$P_k^{v,s}$ Set of feasible sink-trees ending at node $k \in N$ spanning all nodes $m \in M$ of service class $s \in S$ of VPN $v \in V$

$D_{v,s,k}$ Set of point-to-point demand pairs in demand set $k \in K$ of service class $s \in S$ of VPN $v \in V$

$B^d_{v,h,s,k}$ Bandwidth requirement of demand pair $d \in D_{v,s,k}$ of service class $s \in S$ of VPN $v \in V$ during hour-period $h \in H$

Y_l Sizing (topology) variables, capacity assigned to VPN traffic on link $l \in L$

ψ_l Cost of a capacity on link $l \in L$

$U^l_{v,h}$ Capacity at link $l \in L$ allocated to VPN $v \in V$ during hour-period $h \in H$

$X^p_{v,h,s,k}$ Demand-path routing decision variables

$= 1$ if path $p \in P_k^{v,s}$ is used for demand set $k \in K$ of service class $s \in S$ of VPN $v \in V$ during hour period $h \in H$

$= 0$ otherwise

$\gamma^l_{p,d}$ Link path incidence matrix

$= 1$ if demand pair $d \in D_{v,s,k}$ of set $k \in K$ that uses path $p \in P_k^{v,s}$ is directed using link $l \in L$

$= 0$ otherwise

$EB^l_{v,h,s,k}$ Estimated BW requirement of all demand type $k \in K$ on link $l \in L$ of service class $s \in S$ of VPN $v \in V$ during hour-period $h \in H$

$Eqv(B, T_s, Q_s)$ Equivalent bandwidth of traffic in service class $s \in S$ with requirement of bandwidth B (with traffic descriptor T_s and quality of service requirement Q_s)

B. Bandwidth Calculation

The bandwidth requirements at each link will be estimated based on an effective bandwidth calculation [5] where traffic parameters such as connection peak rate and its burstiness are taken into account. Two classes of services in a differentiated service model are considered including premium/guaranteed service and assured service. In the premium service class, applications require an absolute bandwidth guarantee. Thus each traffic connection in this class is allocated a bandwidth equal to a source peak rate R_p. Assuming that η connections are multiplexed within one link, the total allocated bandwidth is

$$Eqv = \eta \cdot R_{peak} \tag{1}$$

where η is derived from an inverse Erlang formulation such that a grade of service constraint (GoS) of the connection (i.e., connection blocking probability - P_b) is met. When a is the source utilization or an offered load of a connection,

$$\eta = InvErlang\ (a, P_b) \tag{2}$$

In the assured service class, applications are expected to have the ability to tolerate a certain amount of delay and loss. For this traffic class, a mean bandwidth guarantee is sufficient along with a statistical delay bound. In bandwidth calculation, source traffic in the assured service class is assumed to be characterized by its source peak rate - R_p, utilization factor - ρ, and mean burst period - b. In this case, the allocated bandwidth may be less than $\eta \cdot R_p$.

$$Eqv = \min\{\eta \cdot m + \alpha'\sigma, \quad \eta \cdot \hat{c}_i\} \tag{3}$$

where $\alpha' = \sqrt{-2\ln(\varepsilon) - \ln(2\pi)}$ given m – a mean bit rate, σ – a variance bit rate, and ε – buffer overflow probability. Equivalent capacity estimation for each source $\hat{c}_i$ is

$$\hat{c}_i = \frac{\alpha b(1-\rho)R_p - B + \sqrt{(\alpha\, b(1-\rho)R_p - B)^2 + 4B\alpha\, b\rho(1-\rho)R}}{2\alpha b(1-\rho)} \tag{4}$$

where $a = \ln(1/\varepsilon)$, assuming that B – buffer size and ε – packet loss ratio are known. The number of connections η multiplexed can be found as before from an inverse Erlang formulation.

C. VPNs Design Formulation

The problem of multi-hour VPNs design can be formulated as VPWBA model shown below. The formulation simultaneously seeks to find VPN link capacity allocation $U^l_{v,h}$ and route assignment $X^p_{v,h,s,k}$ for all VPNs periodically every hour. Using a precomputed sink-tree routing path, traffic demand can be merged within the network, thus the required bandwidth after the merged point can be allocated by aggregated bandwidth of multiplexed traffic within the same service class.

VPWBA : Minimize $\sum_{l \in L} \psi_l * Y_l$

Subject to :

$$\sum_{p \in P_k^{v,s}} X^p_{v,h,s,k} = 1 \quad ;\forall : v \in V\ ,\ h \in H\ ,\ s \in S\ ,\ k \in K \tag{5}$$

$$EB^l_{v,h,s,k} = Eqv\left(\left(\sum_{d \in D_{v,s,k}} B^d_{v,h,s,k} * \sum_{p \in P_k^{v,s}} \gamma^l_{p,d} * X^p_{v,h,s,k}\right), T_s, Q_s\right)$$

$$;\forall: \nu \in V,\ h \in H,\ s \in S,\ k \in K,\ l \in L \qquad (6)$$

$$\sum_{s \in S} \sum_{k \in K} EB^{l}_{\nu,h,s,k} \le U^{l}_{\nu,h} \qquad ;\forall: \nu \in V,\ h \in H,\ l \in L \qquad (7)$$

$$\sum_{\nu \in V} U^{l}_{\nu,h} \le Y_l \qquad ;\forall: h \in H,\ l \in L \qquad (8)$$

$$Y_l \le \alpha_l \cdot C_l \qquad ;\forall: l \in L \qquad (9)$$

$$X^{p}_{\nu,h,s,k} \in \{0,1\} \quad ;\forall: \nu \in V,\ h \in H,\ s \in S,\ k \in K,\ p \in P^{\nu,s}_{k} \qquad (10)$$

$$Y_l \ge 0 \qquad ;\forall: l \in L \qquad (11)$$

The objective of the formulation is to minimize the total capacity cost in allocating network bandwidth to all VPNs. For each VPN, service class, and hour period, constraint (5) selects only one path from a pre-computed set of feasible sink-tree paths ending at egress node P_k for each demand set k. Constraints (6) – (9) impose that capacity assigned at each link is not greater than a utilization limit of link capacity ($\alpha_l \cdot C_l$). Note that, in constraint (6), the capacity is allocated for aggregated traffic within a demand set destined to the same egress node of a VPN. Constraints (10) and (11) require that routing variables and capacity assignment variables are positive. This formulation yields different route assignment and capacity allocation at different hour-periods.

TREE SELECTION HEURISTICS

The VPN design model for MPLS, where a routing path is a sink-tree, often involves enumerating a set of all possible routing tree paths. For example, if a set of candidate tree paths includes all distinct trees spanning over all nodes, the path set can be as large as N^N for an *N*-node network [6]. One can easily see that the problem size (e.g. the number of technical variables) grows exponentially with the number of nodes. Therefore, standard approaches for solving such problems are difficult and sometimes prohibitive due to the complexity of the problem. In order to obtain a solution for a realistic-size network within a feasible amount of time, tree selection heuristics is proposed to limit the set of candidate sink-tree paths to be searched over.

In our extensive study, numerical results obtained from the VPWBA problem reveal patterns of optimal solution showing that optimal trees agree on using less number of links when possible. Besides, a large demand is routed through the shortest possible route when capacity cost can be justified over a spanning tree route. Nonetheless, a small demand may be routed through a longer hop on a sink-tree route when traffic aggregation yields savings in bandwidth. These findings give us some guidelines in choosing candidate tree paths. Typically, in the point-to-point routing path, the size of precomputed path sets can be reduced by imposing a hop-count limitation. However, both hop-count and the number of links in a tree are critical factors in selecting a good candidate set of trees. By imposing the hop-count constraint, we simply avoid choosing single-branch trees which may introduce an undesirable delay violation. In general, choosing a tree having less number of links will promote bandwidth sharing at as many common links as possible. Thus, in the proposed path selection heuristics, the algorithm will start by enumerating distinct trees spanning over all edge routers that satisfy a hop limitation constraint and ranking them based on the number of links used in each candidate tree. Then, only a fixed number of candidate sink-tree paths will be chosen for an optimization procedure. Essentially, trees having less number of links are preferred.

NUMERICAL RESULTS

Backbone networks under study have 10 and 15 MPLS routers (shown in Figure 4 and 5) where 7 and 10 of those are edge-routers respectively. Each link has an OC-12 capacity of 622 Mbps. Traffic from 4 VPNs is assumed to be symmetric with a demand varied uniformly within {10,15,20,25,30} T1 rate. Characteristics of traffic source are assumed to be known (R_p=300 Kbps, b=300 msec, ρ=0.2). Capacity will be allocated for each traffic sources such that the connection blocking probability at the edge router will not exceed 10^{-5}. The VPNs design problem is solved by using AMPL with CPLEX 7.1 InP solver implementing branch and bound solution technique.

Sample results shown in Table 1 compare a full-mesh design versus the proposed design model. By utilizing sink-tree paths in the design, a cost can be reduced approximately by 10 percent. It is also shown that the optimal sink-tree paths tend to share many links in common as previously mentioned. Next, the study of the proposed path selection heuristics is conducted. Figure 4 illustrates the optimal capacity cost and computational time obtained using different sizes of precomputed-path sets for a 10-node network. In this case, the true optimal solution can be best approximated with 180 sink-tree paths and takes 300 seconds of computational time. It is clearly shown that the proposed heuristics path selection algorithm can largely reduce the candidate set of sink-tree paths to be searched over and allows the optimal solution to be obtained quickly. The effect of imposing different values of hop limitation in the heuristics path selection is shown in Figure 5. The hop-count limit of 7 and 10 are used for a comparison. With a hop-count limitation of 7, the obtained solution will converge faster to the true optimal solution using a smaller set of candidate paths.

This implies that choosing the right value for hop-count limitation in the heuristic path selection algorithm can affect the goodness of candidate sink-tree paths.

CONCLUSIONS

In this paper, we have formulated the MPLS based multi-hour VPNs design problem considering sink-tree routing path where traffic aggregation is possible. The model aims to simultaneously find optimal layouts of multiple VPNs so as to minimize the total capacity cost while satisfying QoS requirements. A precomputed set of candidate sink-tree path is used in the formulation to be searched over. Such a problem clearly yields an NP-hard complexity; therefore, a heuristics path selection algorithm is proposed to choose only a small-but-good set of candidate paths to be fed to an optimization model. The numerical results show that the proposed algorithm can reduce computational time significantly.

REFERENCES

[1] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture," RFC 3031, January, 2001.

[2] D. Awduche, et al., "Requirements for Traffic Engineering Over MPLS," RFC 2702, September, 1999.

[3] K. M. Girish, B. Zhou, and J.-Q. Hu, "Formulation of the Traffic Engineering Problems in MPLS based IP Networks," Proceedings ISCC 2000. Fifth IEEE Symposium on Computers and Communications. , Los Alamitos, CA, USA, pp. 214-219, 2000.

[4] H. Saito, Y. Miyao, and M. Yoshida, "Traffic Engineering using Multiple multipoint-to-point LSPs," IEEE INFOCOM 2000, pp. 894-901, March, 2000.

[5] R. Guerin, H. Ahmadi, and M. Naghshineh, "Equivalent Capacity and Its Application to Bandwidth Allocation in High-Speed Networks," 7th ITC Seminar, Morristown, NJ, October, 1990.

[6] N. Christofides, Graph Theory and Algorithmic Approach, London: Academic Press Inc., 1986.

Nodes	Point-to-Point LSPs			Sink-Tree LSPs		
	Cost	# Label / # LSPs	# links used Avg. (Max)	Cost	# Labels / #LSPs	# links used Avg. (Max)
Symmetric Demand						
10	8121	120 / 47	6.42 (8)	7354	23 / 24	6.04 (8)
15	7587	120 / 87	5.79 (8)	6958	24 / 24	5.67 (8)
Asymmetric Demand						
10	7202	103 / 43	5.67 (8)	6722	24 / 21	5.46 (7)
15	5467	80 / 64	5.65 (8)	5083	20 /20	5.45 (7)

Table 1. Point-to-point and sink-tree LSPs comparison

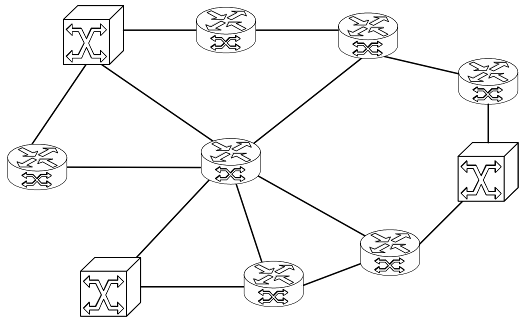


Figure 2.Network with 10 routers and 14 links

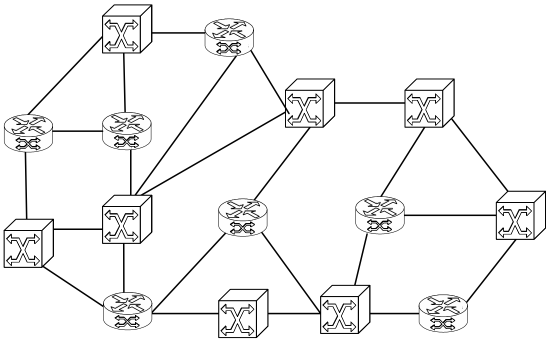


Figure 3. Network with 15 routers and 24 links

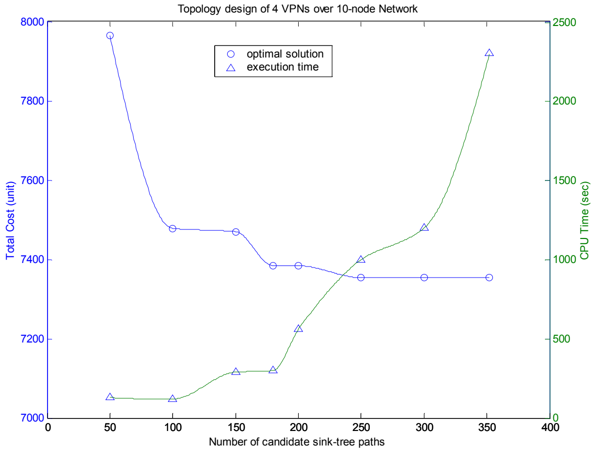


Figure 4. Performance of heuristics path selection.

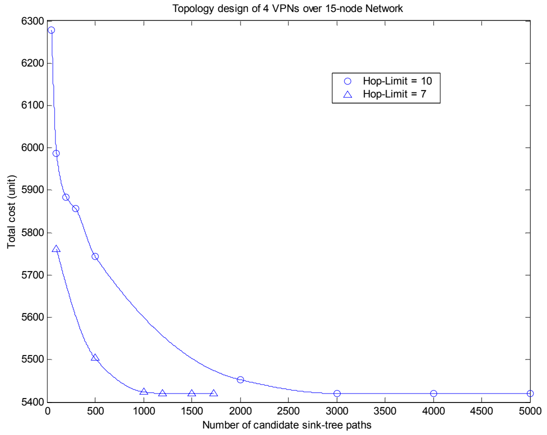


Figure 5. Effect of hop-limit to heuristics path selection.