QoS-based Virtual Private Network Design for an MPLS network

Anotai Srikitja and David Tipper
Dept. of Information Science and Telecommunications
University of Pittsburgh
135 N. Bellefield Avenue,
Pittsburgh, PA 15260
email: anotai@sis.pitt.edu, tipper@tele.pitt.edu

Abstract— In this paper, a VPN design model is proposed for the Next Generation Internet (NGI). With the use of MultiProtocol Label Switching (MPLS) in a backbone network, it is possible to create multiple logical sink-trees (directed multipoint-to-point trees ending at one exit node) carrying traffic of multiple VPNs. This makes the VPN design problem with MPLS different from those of circuit-switched networks and connection-oriented packet networks such as ATM. A big question is how to construct a tree and how to incorporate it in the network design model. Here, the VPN design is modeled as a mixed integer programming (InP) optimization problem to minimize the cost of laying out a VPN supporting different traffic types and service classes on a given topology while meeting QoS requirements. Realizing a sink-tree routing path, the proposed model aims to find an optimal VPN layout supporting multi-service classes during different time periods (multi-hour periods) considering that the traffic demand may vary during the course of the day. Our numerical results show that use of sink-tree approach in VPN design can greatly reduce the amount of bandwidth and the number of label switched paths required.

I. Introduction

Virtual Private Networks (VPNs) provide a private and dedicated environment over a shared private or public network infrastructure. With the advent of broadband technology, a deployment of QoS-based VPNs supporting integrated service for voice, data and video applications together over public data networks appears to be economically appealing since it allows a high-speed access with performance and Quality of Service (QoS) guarantees. Major challenges in deploying QoS-based VPNs over the Internet are delivering a performance guarantee and security assurance to a degree that is comparable to a real private network.

MultiProtocol Label Switching (MPLS) technique proposed by IETF uses a label-swapping forwarding paradigm to expedite a packet-forwarding process [1,2]. MPLS provides a connection-oriented, QoS-based approach to the NGI together with a traditional, connectionless, best-effort approach. Deploying MPLS over a IP network makes it easier for VPN services to provide performance at required levels. Since MPLS is designed to have a traffic engineering capability, provisioning for traffic having different QoS requirements in different Classes of Service (CoS) is possible [3]. In term of security, another no-less important issue, the goal is to protect VPNs data from maliciously or accidentally misconduct. IP Security Protocol (IPSec) [4] is aimed to be used for this purpose.

Using MPLS, multiple VPNs can be constructed on the same network through the use of different MPLS Forward Equivalent Classes (FECs). FEC is used to defined traffic that will be forward in the same manner through a MPLS network. Thus, different FECs will be used to classify traffic from different VPNs which may or may not use the same forwarding path and may or may not share a portion of network bandwidth. The path through an MPLS network can be a multipoint-to-point paths or a sink-tree path ending at one exit node.

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. In addition, in designing a VPN, one must be concerned with scalability issues in order to support a large number of customers. In another words, a well-designed VPN must be easy to manage and attain bandwidth efficiency. Over a MPLS network, this means that the number of label switched paths (LSPs) and required labels must be kept small. In term of capacity efficiency, different levels of traffic aggregation may be considered, for example, aggregation of traffic from different VPNs belonging to the same CoS, aggregation of traffic from same VPNs exiting at the same egress node, etc.

From a design perspective, the concept of a virtual network in general can be applied to VPNs. The notion of virtual networks has long been used to refer to a logical network layout over a network architecture. VPNs over a circuit-switched or an ATM network, are often viewed as a logical mesh network with point-to-point demands between node pairs. A logical full-mesh is a topology where each point-to-point demand pair is independently given a logical link that may be routed over multiple switched-points. The use of sink-tree paths in MPLS makes VPN design problem differ from those in traditional connection-oriented networks previously mentioned. Traffic of different VPNs with the same QoS requirement may/may not be carried on the same routing tree and may/may not share 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 in general over MPLS network. In [5], 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 : (i) one-to-one relationship between traffic trunks and LSPs, (ii) no aggregation, de-aggregation and merging of LSPs. (iii) the model is valid only if one or more feasible solutions exist. [6] 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.

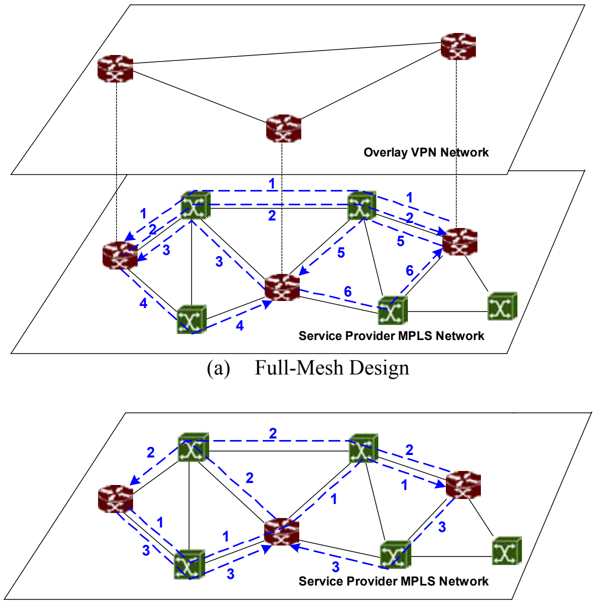
Realizing a sink-tree routing path, this paper proposes mathematical formulations for the problem of VPN design in order to simultaneously find optimal VPNs logical topologies and their dimensions over a service provider IP infrastructure supporting MPLS to carry multi-service, multi-hour VPNs traffic from various customers. Here we exploit pre-computed sink-tree paths (multipoint-to-point paths) over which VPN traffic is routed in a MPLS core network. In the model, different levels of bandwidth aggregation/multiplexing occur across different service classes and routes within one VPN, but not across different VPNs. It is clearly shown that such problem formulations yield a NP-hard complexity. Therefore, preliminary work is conducted for simple cases where only single-service, single-hour VPN traffic is considered. Obtaining the solution to this problem provides a benchmark measure and a guidance to solution feasibility.

II. Sink-tree LSP Path

As previously mentioned, a label switched path (LSP) in MPLS can be a multipoint-to point path referred in here as a sink-tree path. A clear benefit is its scalability since fewer LSPs must be created compared to using a point-to-point path between each demand pair. The number of labels required is also smaller. Thus, management is simpler. For example, applying a full-mesh design where there is a 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 paths 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 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 14 unit cost 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.

III. VPN Design Methodology

The design of VPNs is expected to be a part of the traffic engineering procedure that can be done offline to obtain the VPNs routing plan and virtual network link (VNL) dimension. This is shown in Figure 2. During an operational period, network management will monitor the changes in traffic patterns, network topology or link cost metrics. When it notices any changes that will invalidate the current settings, it will start a global optimization procedure to do an offline recomputation. Note that, the optimization procedure can be done separately for each VPN or jointly for all VPNs to achieve a true optimal solution.



(a) Full-Mesh Design

(b) Sink-tree Design

Figure 1 : Full-mesh versus sink-tree design

For QoS-based VPNs over MPLS, the proposed network design process aims to find the optimal logical sink-tree(s) and its dimension so as to minimize the total network cost while satisfying QoS constraints. Three main tasks are involved in the design process: (i) Tree generation/selection, (ii) Dimensioning and (iii) Routing Optimization. The first task is concerned with generating a candidate set of logical trees for a given source and a set of destinations. The second task is to find a bandwidth that will be allocated to each link in a tree whose bandwidth may/may not be shared by different VPNs traffic. The last task aims to find an optimal route assignment for a given traffic demand. In general, these three main tasks can be solved independently or jointly.

The network topology and node locations (e.g. locations of MPLS edge routers and core routers) will be used to generate multipoint-to-point tree paths which are selected based on traffic QoS constraints. For instance, a bound on maximum delay can be translated into a maximum hop limitation. A precomputed path set is used in the optimization model over which the optimal routes and capacity requirements are determined. The bandwidth allocation/dimension of the virtual network should provide sufficient Grade-of-Service (GoS) (e.g., connection blocking probability) and fairness to different services while satisfying several performance

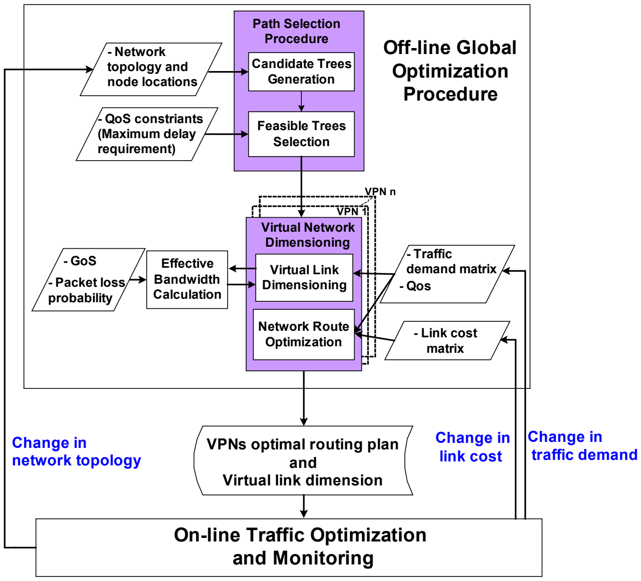
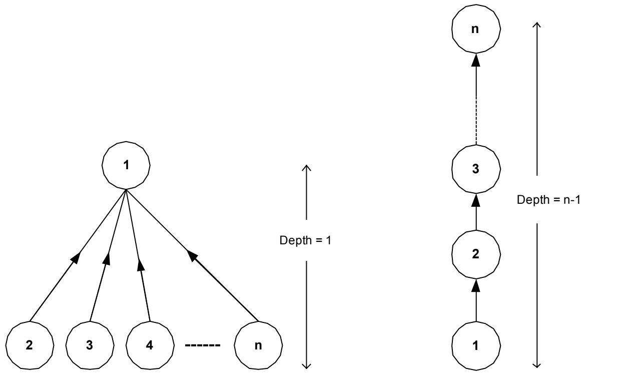

Path Selection Procedure
Candidate Trees Generation
Feasible Trees Selection
Off-line Global Optimization Procedure
- Network topology and node locations
- QoS constraints (Maximum delay requirement)
VPN n
VPN 1
Virtual Network Dimensioning
Virtual Link Dimensioning
Network Route Optimization
- GoS
- Packet loss probability
Effective Bandwidth Calculation
- Traffic demand matrix
- Qos
- Link cost matrix
Change in network topology
VPNs optimal routing plan and Virtual link dimension
Change in link cost
Change in traffic demand
On-line Traffic Optimization and Monitoring

Figure 2 : VPN traffic engineering procedure

constraints at the traffic layer such as packet loss rate and delay. In addition, considering the effects of statistical multiplexing among different connections (when possible), bandwidth allocation can be reduced. Here, we use the concept of "effective bandwidth" to represent service rates required by each traffic connection belonging to different service classes. Effective bandwidth calculations will encapsulate the QoS requirements in term of packet loss rate and delay. Thus, by using the concept of effective bandwidth, traffic flows with different characteristics and QoS requirements can be represented as being steady with a deterministic bandwidth requirement. This simplifies our optimization model. Lastly, the routing optimization will optimally assign a route to a traffic demand, given a set of candidate routes and link capacities. Other than minimizing cost of laying out a given traffic, a route assignment may also aim to balance the load across the network such that the number of over-utilized links and under-utilized link is reduced.

IV. Tree Selection

The choice of a tree is important as it affects the goodness of the solution obtained and the computation time. To reduce the problem size (and thus computation time) for a large network, a precomputed candidate set of trees will be used in the model over which the optimal routes and capacity requirements are determined. A path set will be generated for each source and its destinations given the physical network topology. This set will be limited by a maximum hop-count allowed between each source-destination pair such that the maximum end-to-end delay is bounded. The choice of a routing tree also affects the capacity required and if bandwidth of traffic flows is aggregated and multiplexed when they are merged at one node. For connections within the same class of service, in which a statistical multiplexing can be achieved, a certain part of allocated bandwidth can be shared among them.



(a) Sink-tree with 1-hop depth (b) Sink-tree with (n-1)-hop depth

Figure 3 : Sink-tree routing paths for n nodes

Hence, in tree selection, there is a trade off, between minimizing an end-to-end delay requirement versus minimizing cost of link capacity. Shown in Figure 3 are two different choices of a sink-tree for *n* nodes. The depth of a tree is defined as the distance between the root node and a leave node. Figure 3(a) show a sink-tree of 1-hop paths with a maximum depth of 1. This choice of a tree yields a minimum delay between demand node-pairs, but, since 1-hop paths merge at the root node, no bandwidth aggregation is possible. Oppositely, the tree in Figure 3(b) with a depth of *(n-1)* yields a maximum bandwidth gain due to statistical multiplexing at all links after the merge points. Different types of trees including spanning trees, shortest (distance) path trees and Steiner trees (minimum-cost trees) are among potential choices.

V. MPLS-VPN Design Formulations

Given a network topology, node location and link capacity, an optimization model is formulated for VPN design. 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 of the network respectively. M ($M \subset N$) is a set of edge nodes (edge routers) where there is a demand traffic entering or exiting. Thus, $N-M$ represents a 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 VPN 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. This factor is assumed to be known.

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 $v \in V$

$B_{v,h,s,k}^d$ 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 $l \in L$

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

$X_{v,h,s,k}^p$ 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

$\gamma_{p,d}^l$ 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

$EB_{v,h,s,k}^l$ 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 calculation function for traffic in service class $s \in S$ with requirement of bandwidth amount B (with traffic descriptor T_s and quality of service requirement Q_s)

B. *Traffic Demand*

The complete matrix of VPNs traffic demand is assumed to be given. It can be derived from all SLAs (Service Level Agreements), between customers and service providers. SLAs typically specify various classes of service and how much traffic in each service class a user is allowed to send. In more detail, 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, delay jitter). 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.

C. *Selection of Candidate Paths*

Feasible sink-trees or multipoint-to-point paths are used in the optimization model where a traffic demand may be assigned. The feasible path set $P_k^{v,s}$ can be pre-computed for VPN v and service class s having different QoS requirements (i.e., maximum end-to-end delay requirement). A path $p \in P_k^{v,s}$ is selected from candidate sink-tree paths which are spanning trees rooted at egress node $k \in K$ and spanning over all the edge nodes $m \in M$ or a subset of the edge nodes. Set of candidate paths can be generated by enumerating all distinctive spanning trees. Algorithms to determine these trees can be found in previous works including [7]. The maximum end-to-end delay requirement is translated in to the hop-count limitation constraint. This constraint will limit the path set where only feasible paths are selected from all candidate paths.

D. *Bandwidth Calculation*

The bandwidth requirements at each link will be estimated based on an effective bandwidth calculation [8] where the 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/guarantee service and assured services.

(i) Premium Service

In the premium service class, packet loss, delay and delay jitter must be bounded. The traffic of this class requires an absolute bandwidth guarantee. Thus each traffic connection in this class is allocated a bandwidth equal to a source peak rate R_{peak}. Assuming that η connections are multiplexed within one link, total allocated bandwidth (Eqv)

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

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

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

where a is the source utilization or an offered load of a connection.

(ii) Assured Service

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 only needed 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_{peak} , utilization factor - ρ , and mean burst period - b . In this case, the allocated bandwidth (Eqv) is less than $\eta \cdot R_{peak}$.

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

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

$$\hat{c}_i = R_{peak} \frac{a - 1 + \sqrt{(a-1)^2 + 4\rho a}}{2a} \qquad (4)$$

where

$$a = -\frac{b}{B}(1-\rho)\ln\varepsilon$$

assume 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.

E. *General Case without Bandwidth Aggregation*

Using a sink-tree routing path, traffic demand can be merged within the network, thus the required bandwidth after the merged point can be allocated separately for each demand-pair or multiplexed together within the same service class. The latter yields a reduction in bandwidth requirements especially for traffic in the assured service class, due to a statistical multiplexing gain. The basic formulations are given below. The model assumes that the followings are given: (i) link utilization factor α_l and the link capacity C_l, (ii) set of traffic demand pair $D_{v,s,k}$ and bandwidth requirement $B^d_{v,h,s,k}$, (iii) a precomputed sink-tree path set $P_k^{v,s}$ and a corresponding link path incidence matrix $\gamma^l_{p,d}$. The formulation seeks to find VPN link capacity allocation $U^l_{v,h}$ and its route $X^p_{v,h,s,k}$ for all VPNs in each hour period.

Formulation-I shows the case where there is no bandwidth aggregation. The objective of the formulation is to minimize the total capacity cost in providing service 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) imposes that capacity assigned at each link must not be greater than a utilization limit of link capacity ($\alpha_l \cdot C_l$). Note that, in constraint (6), the capacity calculation is done separately for each traffic demand-pair. Constraints (10) and (11) require that routing variables and capacity assignment variables must be positive. This formulation yields different route assignment and capacity allocation at different hour-periods.

<u>Formulation-I</u>

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

Subject to :

$$\sum_{p \in P_k^{v,s}} X^p_{v,h,s,k} = 1$$

$$; \forall : v \in V,\ h \in H,\ s \in S,\ k \in K \qquad (5)$$

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

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

$$\sum_{s \in S} \sum_{k \in K} EB^l_{v,h,s,k} \le U^l_{v,h}$$

$$; \forall : v \in V,\ h \in H,\ l \in L \qquad (7)$$

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

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

$$X^p_{v,h,s,k} \in \{0,1\}$$

$$; \forall : v \in V,\ h \in H,\ s \in S,\ k \in K,\ p \in P_k^{v,s} \qquad (10)$$

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

F. *General Case with Bandwidth Aggregation.*

Here, we introduce a case where bandwidth of various traffic demand is aggregated at links where possible. The aggregation only occurs within a demand set destined to the same egress node of a VPN. In this case, the objective function and constraints are similar to previous case except for constraint (6) is replaced by (12). The total traffic demand routed on one link is aggregated and bandwidth allocation is done together.

<u>Formulation-II</u>

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

Subject to :

(5), (7), (8), (9), (10), (11), and

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

$$\forall : v \in V,\ h \in H,\ s \in S,\ k \in K,\ l \in L \qquad (12)$$

VI. Preliminary Numerical Study

The mixed-integer formulations for the VPN design problem, shown previously, have a NP-hard complexity. A simplified version of these formulations can be derived when we only consider traffic demand of VPNs having one service class and hour-period. Thus, the formulation-I can be reduced to:

<u>Formulation-III</u>

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

Subject to :

$$\sum_{p \in P_k} X^p_k = 1 \qquad ; \forall : k \in K \qquad (13)$$

$$\sum_{k \in K} \sum_{d \in D_k} Eqv\left(B_k^d , T, Q\right) * \left(\sum_{p \in P_k} \gamma_{p,d}^{l} * X_k^p \right) \leq Y_l$$

$$; \forall : l \in L \qquad (14)$$

$$Y_l \leq \alpha_l \cdot C_l \qquad ; \forall : l \in L \qquad (15)$$

$$X_k^p \in \{0, 1\} \qquad ; \forall : k \in K \text{ , } p \in P_k \qquad (16)$$

$$Y_l \geq 0 \qquad ; \forall : l \in L \qquad (17)$$

In the same manner, the formulation-II can be reduced to :

Formulation-IV

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

Subject to :

(13), (15), (16), (17), and

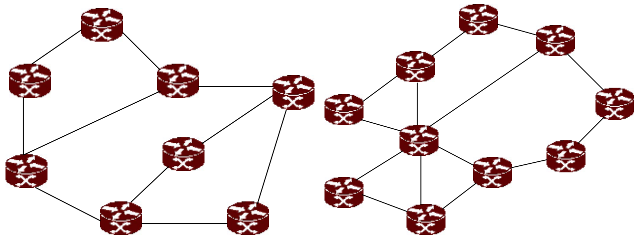
$$\sum_{k \in K} Eqv\left(\left(\sum_{d \in D_k} B_k^d * \sum_{p \in P_k} \gamma_{p,d}^{l} * X_k^p \right), T, Q\right) \leq Y_l$$

$$; \forall : l \in L \qquad (18)$$

Obtaining a solution to problems stated in formulation-III and IV is easier than ones in formulation-I and II. A pilot study was conducted by translating formulation-III and IV using the AMPL model description language and solution is obtained using CPLEX 6.6 optimization solver implementing a branch and bound solution technique. The networks studied were small networks with 8 and 10 nodes, with equal capacity link-cost, shown in Figure 4. The capacity of each link C_l was set to $C_l = 1000 \; ; \forall l \in L$, so that capacity was not a limiting factor. Different cases are shown in Table 1. For case-I, asymmetric load of fixed demands was studied that is there was one unit of demand from each node to every other node. For case-II, a symmetric load of demand was considered with the demand generated from a Uniform(1,5) distribution. For case-III, an asymmetric load of demand was studied with nonzero demand only from a subset of network nodes. The nodes with nonzero demand were randomly selected with a random load drawn from a Uniform(1,5) distribution.

From Table 1, one can see that the optimal solutions obtained from a sink-tree design with no bandwidth aggregation are not different from ones obtained from a full-mesh design. It is observed that a path used in a sink-tree design (with no bandwidth aggregation) is simply a shortest-path tree. This is similar to a full-mesh design where a demand is routed along a shortest path. In terms of cost, when bandwidth aggregation is considered in a sink-tree design, a cost reduction is realized approximately by 30-40 percent in all cases. This is because traffic demand may be routed using a sink-tree path that is different from a shortest path tree. Note that there is a huge difference in the number of LSPs used between the sink-tree and full-mesh design approaches.

In order to obtain a solution to general case problem, where multi-service and multi-hour period are considered, and where multiple VPNs layouts are simultaneously optimized, one seeks to find more efficient solution method due to the problem complexity. Results from the pilot study suggest that a heuristics method may start out its search from a so-called near-optimal solution obtained by routing demand traffic using shortest-path trees, then seek out a better solution as it moves along a projected direction within a feasible search space. One may apply various heuristics techniques explored in the literature, such as greedy algorithm, simulated annealing, or genetic algorithm, to solve this problem.



(a) 8-node network (b) 10-node network

Figure 4 : Networks under study

VII. Conclusion

In this paper we have formulated the MPLS based multi-hour VPN design problem with and without bandwidth aggregation. We modeled a VPN as multiple logical sink trees which reduces the number of label switch paths and allows the possibility of bandwidth savings. Sample numerical results for different studied cases shows that a sink-tree design with no bandwidth aggregation yields the same solution as a full-mesh design where demand was routed along a shortest-path. However, when bandwidth aggregation is considered in a sink-tree design, demand was routed along a tree that is different from a shortest-path tree such that a link capacity assignment can be smaller and the total capacity cost reduction is realized.

References

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

[2] B. Davie and Y. Rekhter, MPLS : technology and applications, San Francisco: Morgan Kaufmann Publishers, 2000.

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

[4] S. Kent and R. Atkinson, "Security architecture for the Internet Protocol," RFC 2401, November, 1998.

[5] 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.

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

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

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

Topology	**Full-Mesh Design**			**Sink-Tree(s) Design (w/o BW aggregation)**			**Sink-Tree(s) Design (with BW aggregation)**		
	Optimal Cost	Simplex Iterations	No. of LSPs	Optimal Cost	Simplex Iterations	No. of LSPs	Optimal Cost	Simplex Iterations	No. of LSPs
Case I : Symmetric fixed-load									
8-node	104	54	56	104	241	8	56	473	8
10-node	174	162	90	174	308	10	90	925	10
Case II : Symmetric variable-load									
8-node	346	64	56	346	99	8	203	2,016	8
10-node	562	112	90	562	242	10	410	4,239	10
Case III : Asymmetric variable-load									
8-node	166	42	33	166	101	7	105	319	7
10-node	212	76	35	212	216	7	129	565	7

Table 1 : Comparison for different design cases