WAN - VPN Network Design

David Tipper
Graduate Telecommunications and Networking Program
University of Pittsburgh
Telcom 2110 Network Design, Slides 11

WAN Network Design

• WAN typically have a mesh or ring design
• Mesh topologies introduce the problem of routing traffic
• Many algorithms/optimization formulations/design tools for WAN packet network design
  – Tend to be imbedded in network topology/data rate layer
  – Different design techniques and metrics at different layers
    • IP
    • MPLS – VPN design
    • WDM – circuit switched routing and wave length assignment
  – In general techniques are either
    • Graph theory based
    • Optimization based
    • Heuristics
  – Considered Routing heuristics for IP layer design
  – Consider Optimization techniques for VPN design
Optimization Based Design


Maximize (or minimize): \( f(x_1, x_2, \ldots, x_n) \)

Subject to:

\[
\begin{align*}
  & g_1(x_1, x_2, \ldots, x_n) \leq b_1 \\
  & g_2(x_1, x_2, \ldots, x_n) \leq b_2 \\
  & \vdots \\
  & g_m(x_1, x_2, \ldots, x_n) \leq b_m
\end{align*}
\]

... where \( x_1, x_2, \ldots, x_n \) are the *decision variables*

Formulations usually either

- Linear Programming problems (objective and constraints linear)
- Integer Programming problems (linear objective and constraints but integer design variables)
- Mixed Integer Programming problems
- Nonlinear MIPs, etc.

Optimization Based Network Design

Input Data

- Node Locations
- Potential Links
- Traffic Demands
- Cost function/parameters

Network Design Optimizer

- Find working path for traffic demands
- Find backup paths for given failure scenarios
- Survivability requirements
- Technology, QoS requirements, etc.

Network topography & Link capacity assignment

Network design strategy/Network optimization model

Output Results

- Network Cost
- Network Topology
- Capacity of Links
- Working Paths
- Backup Paths
WAN Network Design

- Many optimization formulations and design tools for WAN network design
  - Optimization Techniques usually form the initial basis of the formulation
  - Often use a heuristic or meta-heuristic solution technique
- Formulation depends
  - Network layer (e.g., WDM, SONET, MPLS, etc.),
  - Technology,
  - QoS requirements
  - Reliability goals
  - Other constraints
- Consider MPLS VPN design as an example

Virtual Networks

- Virtual Network refers to
  "A class of service that uses a shared network infrastructure to emulate the characteristics of a private network."
- The virtual network should for all intensive purposes appear to the users as a fully functional physical network ("network within a network").

Typically have many Virtual Networks over a common shared network infrastructure

Normally separate traffic among virtual networks - termed Virtual Private Networks (VPNs)

Each VPN can have different service levels (i.e., bandwidth, availability, packet loss, etc.)
Virtual Networks

- Why Virtual Networks?
  - **Cost**
    - For example, in WANs cheaper to lease virtual bandwidth connectivity than string fiber
    - Service Providers can support many virtual networks simultaneously
      - Workforce optimization
  - **Security/Legal/Regulatory Compliance**
    - Provides separation of traffic – can quarantine non compliant hosts
  - **Manageability**
    - VN network manager can concentrate on managing their infrastructure
      - Links through service provider managed in part or in total by service provider
    - Consolidation of multiple networks onto a single infrastructure
  - **Flexibility**
    - Service Providers can provide different service levels (QoS, Security, availability) to VNs - can rearrange logical VN links to support demand changes, etc.
    - In LAN user can move offices and not change VLAN

Virtual Networks

- **Virtual Network Technology**
  - Virtual Networks can be implemented at several network layers in different technologies
  - Some technology options
    - **WAN**
      - Circuit Switched Logical Virtual Trunks
      - Frame Relay
      - ATM – Virtual Paths
      - MPLS - LSPs
      - WDM – lightpath based VN
    - **MANs**
      - Sonet, Frame Relay, WiMAX
    - **LANs**
      - Ethernet, 802.11
  - Overlay Applications (MBONE, VNET, etc.)
  - Tunneling Protocols (GRE, IPSEC, PPTP, etc.)
- Consider one type of WAN of VN technology
  - Multi Protocol Label Switching (MPLS)
IP QoS in WANs

Trend to Use Diffserv on Edge MPLS in Backbone
Most Providers have significant MPLS networks in Core

CE: Customer Equipment, PE: Provider Equipment

In core use MPLS

Provider Core

Diffserv treatment on packets leaving CE

CE

PE

Diffserv treatment on packets leaving PE

CE

PE

MPLS Background

- Multi-Protocol Label Switching (MPLS) is used over IP infrastructure to deliver QoS performance to different service classes, provide security and VPNs.
- MPLS use a label-swapping forwarding technique.
- Labels are assigned when the packet enters into the network.
- MPLS routers forward packets based on the label value.
- Fields: Label, Class of Service, Stack bit, Time-to-Live

<table>
<thead>
<tr>
<th>0</th>
<th>20</th>
<th>32</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>CoS</td>
<td>S</td>
</tr>
</tbody>
</table>

| IP header, Ethernet header, etc. |

MPLS header
Forward Equivalent Class

- Packets entering MPLS domain will be classified into different Forward Equivalent Class (FEC) by an edge router called a label switching router (LSR).

- A Forwarding Equivalence Class (FEC): A subset of packets that are all treated the same way by an LSR – basically separating traffic.

- A packet’s FEC can be determined by one or more of the following:
  - Source and/or destination IP address
  - Source and/or destination port number
  - Destination-based (unicast) routing
  - Multicast routing
  - Traffic engineering
  - VPNs
  - QoS
  - Security, etc...

- Different levels of traffic aggregation are possible.
  - Aggregate traffic based on its IP destination pre-fixed.
  - Aggregate traffic based on its egress node.
  - etc..

MPLS Network

- Customer Edge (CE) Router
- Provider Edge (PE) Router
- Provider Core Router

Entry PE router

“Packet Classification”
MPLS Operation

- At ingress LSR of an MPLS domain, an MPLS header is inserted to a packet before the packet is forwarded
  - Label in the MPLS header encodes the packet’s FEC

- At subsequent LSRs
  - The label is used as an index into a forwarding table that specifies the next hop and possibly a new label.
  - If the label is to be replaced, the field is overwritten with the new label, and the packet is forwarded to the next hop.

- Egress LSR strips the label and forwards the packet to final destination based on the IP packet header

- Note a MPLS router forwards packets to outgoing interface based only on label value (don’t inspect IP address)
  - MPLS forwarding table distinct from IP forwarding tables
  - Provide speed up in the network core

Label Switched Path

- A label-switched path (LSP) is a path through which packets of a particular FEC will be forwarded.

- Two types of LSPs
  1. Prefixed-based
     - LSPs created based on “route” advertisement
     - Controlled by routing or signaling
     - Traditional IP-OSPF routing (No QoS or CoS support)
  2. Tunnel based
     - LSPs created between specific MPLS end-points
     - Controlled by signaling (RSVP-TE, CR-LDP)
     - Used for traffic engineering, QoS provision, VPN service, etc.

- LSPs are always unidirectional
MPLS Packet Forwarding

MPLS Network

IP packet 171.68.10.12
Label=5
IP packet 171.68.10.12
Label=3

MPLS Operation

Can keep LABEL end-to-end – create a tunnel through the network
Inf: Interface or Port

FEC  Inf  Label
  a  1  50

1, 2, 3

In Port  PortPrefixLab  Out Port  PortPrefixLab
3 171.68.10 / 24
Label Assignment and Distribution

- Labels have local significance
- Labels are assigned and exchanged between adjacent LSRs from **downstream to upstream** direction.
  - Unsolicited downstream
    LSRs assign a label to each FEC and distribute labels to all its upstream neighbors.
  - Downstream on demand
    For each FEC, upstream LSR request label binding to its downstream LSRs.

- In MPLS, a forwarding path can be a **sink-tree path** or a multipoint-to-point (MP2P) path ending at an egress node.

---

**Downstream on Demand Label Assignment**

- Use label 5 for Destination 171.68.10 / 24
- Use label 3 for Destination 171.68.10 / 24

171.68.40 / 24 LSR - A LSR - B LSR - C 171.68.10 / 24

- Request label for Destination 171.68.10 / 24
VPN over MPLS backbone

- MPLS is used over a transport backbone network to deliver QoS performance to different classes of service (CoS).
- Using explicit path setup in MPLS, a QoS-based VPN can be efficiently built over the infrastructure.
- Different FECs may be used to classify traffic from different VPNs which may or may not
  - use the same forwarding path.
  - share some portion of network bandwidth.
- VPN point-to-point traffic can be implemented as
  - Full mesh of LSPs \( (N \times N-1) \) LSPs
  - Multi-point to point sink tree LSPs \( N \) LSPs

Logical Network Topology

![Logical Network Topology Diagram]
Point-to-Point LSPs

Multipoint-to-point LSPs
SLAs

- Service Level Agreements (SLAs)
  - *Legal* agreement between customers and service provider for VPN service – specifies cost, security, performance factors, traffic demand, domains of responsibility
  - Performance factors can include delay, bandwidth, delay jitter, packet loss, availability, etc.
  - Traffic demand – peak bit rate, mean rate, burst length
  - Cogent – basically a Business ISP backbone VPN provider - see SLA on class web page

<table>
<thead>
<tr>
<th>Cogent Communications Network Performance Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>North America: Packet Success Rate</td>
</tr>
<tr>
<td>North America: Average Backbone Latency (ms)</td>
</tr>
<tr>
<td>New York to London: Average Backbone Latency (ms)</td>
</tr>
<tr>
<td>New York to London: Average Backbone Latency (ms)</td>
</tr>
</tbody>
</table>

- SLA will define metrics to evaluate service level
  - For example monthly average delay across backbone, monthly packet loss rate, yearly availability, etc.
- Should clearly define
  - Where metrics measured,
  - How measured
  - Frequency of measurement
- Should define penalties for not meeting SLA (usually in $) and traffic enforcement/shaping by provider

- Good References
  - TMFORUM SLA Handbook
Basic VPN Network Design

• Goals of VPN network design same as general network design – just an overlay connecting a subset of network nodes or all nodes
  – Minimize total cost
  – Subject to Constraints … for example
    • Link capacity must exceed some min, and be less than some max
    • Average Packet Delay must be < maximum
    • Reliability requirements
    • Throughput, etc.
  – Variations

• Main difference is physical topology defined
• Multiple VPNs are usually deployed
• Consider MPLS case

VPN Design Problem

• Given
  – Physical network topology
  – Link capacity and its cost
  – VPNs requirements (i.e., traffic demands, QoS parameters)
  – Consider two LSP approaches
    • Mesh based on Point to Point LSPs
    • Mesh multi-point to point sink-tree LSPs

• Find the optimal LSP layouts and their dimension such that
  – Objective:
    • Minimize (total virtual network cost)
  – Constraints:
    • QoS requirements (packet-level parameters)
    • GoS requirements (call-level parameters)
    • Link capacity limitations
VPN Design Procedure

Three main tasks

1) Path generation
   - Candidate path generation
   - For multi-point to point sink trees generate candidate trees – reduce complexity by limiting possible tree selection

2) Virtual link dimensioning
   - Logical link bandwidth sizing of a given demand

3) Network routing optimization
   - Route selection for each VPN logical link

Path/Tree Selection

- Paths
  - Would like to consider all possible paths between a source destination pair
  - Usually much too big a set
  - Standard approach to reduce complexity is to put a hop count limit on the paths considered
  - Doesn’t have to be the same limit for each s-d pair (rule of thumb is shortest path + 2 hops)

- Trees
  - All distinct spanning trees or shortest path trees
  - Usually too big a set
  - Prune from set of possible set of trees
  - Hop-count limit (e.g., max s-d shortest path + 2)
    - Avoid long branch trees in the feasible space
  - Select trees having fewer total number of links
    - Promotes bandwidth sharing
Link Dimensioning

- Must determine LSP capacity
  - In some cases the customer is just leasing bandwidth at a fixed data rate (e.g., OC-3) as specified in SLA so dimensioning is predefined
  - Can use Min-Delay, Min-MAX Delay capacity allocations from previous lecture - depending on QoS requirements and expected revenue from SLA
  - Can use “Effective bandwidth” calculation to determine the link capacity requirement
    - Use parameters from SLA (e.g., mean rate for traffic, mean packet loss rate) and traffic shaper/enforcement at service provider EDGE router.
    - Maps parameters into an equivalent capacity $C$ which is dimensioned for LSP along path.

- Many different effective bandwidth calculations in the literature

Traffic with source characteristic: mean rate $m$, standard deviation $\sigma$, peak rate $R_{\text{peak}}$, utilization $\rho$, burst length $b$, ...

Equivalent capacity estimation for each source $\hat{c}_i$:

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

where

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

Packet loss rate
Buffer size in sec at peak rate
Link Dimensioning Example

• Consider the sink tree LSP 1 shown in the MPLS network on slide 22
• At nodes B and C the offered load to node A has the characteristics
  \[ R_{\text{peak}} = 1 \text{Mbps}, \quad b = 300 \text{ msec}, \quad \rho = 0.2, \quad B = 1500 \text{ msec at peak rate} \Rightarrow 1.875 \text{ Mbytes}, \quad \varepsilon = 10^{-5}, \quad m = 0.2 \text{Mbps}, \quad \sigma = 0.24 \]
• Determine the amount of capacity that should be allocated to the LSP at the entry nodes of LSP 1
  • First compute \( a \)

\[
\begin{align*}
\lambda &= -\frac{b}{B}(1-\rho)\ln(\varepsilon) = -\left[\frac{300}{1500}(1-0.2)\ln(10^{-5})\right] = 1.8421 \\
\hat{c}_i &= R_{\text{peak}} \left( a - 1 + \sqrt{(a-1)^2 + 4\rho a} \right) = 0.626958 \text{ Mbps}
\end{align*}
\]

Link Dimensioning (cont.)

• At a merged link in a multi-point to point tree one can aggregate bandwidth among branches of traffic flowing to destination

For \( \eta \) traffic connections multiplexed

Allocated BW : \( \hat{C} \leq \eta \cdot R_{\text{peak}} \)

\[
\hat{C} = \min \left\{ \eta \cdot m + \alpha' \cdot \sigma, \quad \eta \cdot \hat{c}_i \right\}
\]

\[
\alpha' = \sqrt{-2\ln(\varepsilon) - \ln(2\pi)}
\]

• Bandwidth efficiency and reduction in number of LSP is achieved when traffic is merged into the same pipe.
Link Dimensioning Example

- One of the motivations for using sink trees is save bandwidth where branches of a tree merge. Determine the equivalent capacity needed for the traffic from nodes B and C to A in LSP 1 at the merger point.

Determining the bandwidth requirement at the merge point we have

\[ \alpha' = \sqrt{-2 \ln(\varepsilon) - \ln(2\pi)} = 4.603 \]

\[ \hat{C} = \min \{\eta \cdot m + \alpha' \cdot \sigma, \quad \eta \cdot \hat{c}\} = \min \{1.5047, 1.259\} = 1.259 \text{Mbps} \]

In this case the merged bandwidth is just twice the individual flows but note it is much less than twice the peak rate of 2Mbps.

Design Optimization Model

- Set of links in the network.  \( L \)
- Set of nodes in the network.  \( N \)
- Set of edge nodes in the network.  \( M \subset N \)
- Maximum capacity of link  \( l \in L \)
- Utilization factor of link  \( l \in L \)
- Cost of a capacity on link  \( l \in L \)
- Demand set index  \( K \subset M \)
- Set of candidate paths for demand pair  \( k \)
- Set of point-to-point demand pairs in demand set  \( k \in K \)
- Bandwidth requirement of demand pair  \( d \in D_k \)
- Maximum Limit on number of LSPs on link  \( l \in L \)
Decision Variables

\( Y_l \)  
Sizing decision variable  
\( \sum \text{capacity assigned for VPN traffic on link } l \in L \)

\( X'_k \)  
Routing decision variables  
\( = \begin{cases} 1 & \text{if path } p \in P_k \text{ is used for demand set } k \in K \\ 0 & \text{otherwise} \end{cases} \)

Also have from the pre-computed path set

\( \gamma'_{p,d} \)  
Link path incidence matrix  
\( = \begin{cases} 1 & \text{if demand pair } d \in D_k \text{ of set } k \in K \text{ that uses path } p \in P_k \text{ is directed using link } l \in L \\ 0 & \text{otherwise} \end{cases} \)

Mathematical Formulation

Minimize

\[ \sum_{l \in L} \psi_l \cdot Y_l \]

Total VPN capacity cost

Given

\( \alpha_l, C_l \)  
Link utilization factor, Maximum link capacity

\( D_k, B^d_k \)  
Traffic demand, Bandwidth requirement

\( P_k, \gamma'_{p,d} \)  
Set of precomputed sink-tree paths, Link path incidence matrix

Obtain

\( Y_l \)  
Link BW allocations

\( X'_k \)  
Routing path
**Mesh VPN with Point-Point Links**

### Formulation

**Minimize**

\[ \sum_{l \in L} \psi_l \cdot Y_l \]

**Subject to:**

1. \[ \sum_{p \in P_k} X_{k}^{p} = 1 \quad \text{for all } k \in K \]
   - **Path selection criteria**

2. \[ \sum_{k \in K} \sum_{d \in D_k} B_{k}^{d} \left( \sum_{p \in P_{k}} \gamma_{p}^{l} \cdot X_{k}^{p} \right) \leq Y_{l} \quad \text{for all } l \in L \]
   - **Link capacity requirement**

3. \[ Y_{l} \leq \alpha_{l} \cdot C_{l} \quad \text{for all } l \in L \]
   - **Link capacity limitation**

4. \[ \sum_{p \in P_{k}} \sum_{d \in D_k} \gamma_{k}^{d} \leq SP_{l} \quad \text{for all } l \in L \]

---

### Sink Tree Case

- Point-to-point traffic demand is considered.
- A traffic demand is classified/routed based on an exit node.
- Two cases:
  - No bandwidth aggregation
  - Bandwidth aggregation is possible when traffic is merged/multiplexed on a sink-tree paths.
- Sample notation as before
  - except candidate path set – now a set of trees rooted at destination nodes

\[ P_{k} \quad \text{Set of candidate sink-trees ending at node } \quad k \in N \]

\[ Eqv(B, T, QoS) \quad \text{Equivalent bandwidth of traffic requires capacity } B \text{ with traffic descriptor } T \text{ and QoS parameters} \]
Without Bandwidth Aggregation

**Formulation**

Minimize \[ \sum_{l \in L} \psi_l \cdot Y_l \]  
\[ \text{Total VPN capacity cost} \]

Subject to:

(1) \[ \sum_{p \in P_k} X^p_k = 1 \] ; for all \( k \in K \)

(2) \[ \sum_{k \in K} \sum_{d \in D_k} Eqv(B^d_k, T, QoS) \cdot \left( \sum_{p \in P_k} \gamma^l_{p, d} \cdot X^p_k \right) \leq Y_l \] ; for all \( l \in L \)

(3) \[ Y_l \leq \alpha_l \cdot C_l \] ; for all \( l \in L \)

(4) \[ \sum_{p \in P_k} \sum_{d \in D_k} \gamma^d_{k, l} \leq SP_l \] ; for all \( l \in L \)

---

With Bandwidth Aggregation

Minimize \[ \sum_{l \in L} \psi_l \cdot Y_l \]  
\[ \text{Total VPN capacity cost} \]

Subject to:

(1) \[ \sum_{p \in P_k} X^p_k = 1 \] ; for all \( k \in K \)

(2) \[ \sum_{k \in K} Eqv\left( \sum_{d \in D_k} B^d_k \cdot \left( \sum_{p \in P_k} \gamma^l_{p, d} \cdot X^p_k \right), T, QoS \right) \leq Y_l \] ; for all \( l \in L \)

(3) \[ Y_l \leq \alpha_l \cdot C_l \] ; for all \( l \in L \)

(4) \[ \sum_{p \in P_k} \sum_{d \in D_k} \gamma^d_{k, l} \leq SP_l \] ; for all \( l \in L \)
Numerical Study

• The design model is a mixed Integer programming problem of \textit{NP-hard} type.

• The problem is translated using AMPL model description language and solution is obtain using CPLEX solver.

• Benchmark is obtained for a small network where LP bound is easy to establish.

Tested Networks

- 8-node Network
- 10-node Network
Sample Results

<table>
<thead>
<tr>
<th>Topology</th>
<th>Full-Mesh Design</th>
<th>Sink-Tree(s) Design (w/o BW aggregation)</th>
<th>Sink-Tree(s) Design (with BW aggregation)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Optimal Cost</td>
<td>Simplex Iterations</td>
<td>No. of LSPs</td>
</tr>
<tr>
<td>8-node</td>
<td>104</td>
<td>54</td>
<td>56</td>
</tr>
<tr>
<td>10-node</td>
<td>174</td>
<td>65</td>
<td>90</td>
</tr>
</tbody>
</table>

Multiple VPN Design Problem

- **Given**
  - Physical network topology
  - Link capacity and its cost
  - Number of VPNs and their requirements
    - Traffic demands, QoS, etc.
    - May have 100 or 1000s of VPNs to provision over infrastructure
- **Find the logical layout and their dimension such that**
  - **Objective**:
    - Minimize (total virtual network cost, number of LSPs)
  - **Constraints**:
    - QoS requirements (packet-level parameters)
    - Link capacity limitations
- **Can modify optimization models to add set of VPN to be designed - difficult to scale problem to large networks and large number of VPN**
Multiple VPN Overlays

Can solve optimization model in one at a time fashion

Does the order in which the VPNs are designed matter?

YES

By modifying optimization model to include multiple VPNs and solving optimization model for small cases can see how to rank VPNs

Multiple VPN Design

- **Heuristic Solution**
  1. Assign each VPN a weight = \( \text{Sum (traffic demand x distance (i.e., shortest path))} \)
  2. Sort VPNs on weight in descending order (i.e., largest weight first) and set VPN index \( i=1 \)
  3. Use a standard algorithm to layout VPN \( i \) (Mentor, Routing approach, optimization formulation, etc.)
  4. Subtract VPN \( i \) bandwidth requirements from physical network capacity where used.
  5. Increment \( i = i+1 \), repeat steps 3 and 4 until all VPNs are provisioned.
     - Note VPNs with small weight may have non-shortest path routes/trees through the physical network.
Summary

- Optimization based WAN Design
  - Virtual Network Design in WANs
  - MPLS
    - LSPs, Point-to-Point vs. Sink Trees
  - Single VPN Optimal Design Formulation
    - Point-to-Point
    - Sink Trees
    - Sink Trees with bandwidth aggregation
  - Heuristic for Multiple VPN Design