Internet Technology

The “inner network“ view, part 2 (B): MPLS

Michael Welzl http://www.welzl.at

DPS NSG Team http://dps.uibk.ac.at/nsg
Institute of Computer Science
University of Innsbruck, Austria

Using RSVP for MPLS

- Originally designed for QoS in the context of IntServ
  - Per-(end-to-end)-flow resource reservation
  - Heavyweight protocol with multicast support
  - Extended for use with MPLS
    - create/maintain LSPs
    - associated bandwidth reservations
    - number of flows is much smaller (concerns LSPs, not end-to-end paths)
    - still, state grows as network grows (proportional to number of LSPs)
- Important property which is different from LDP: explicit routing
  - Can ignore IGP
- Ingress router can specify
  - Entire path, or
  - Transit nodes that must be contained in the path
  (like strict or loose IP source routing)

Consequences of RSVP explicit routing

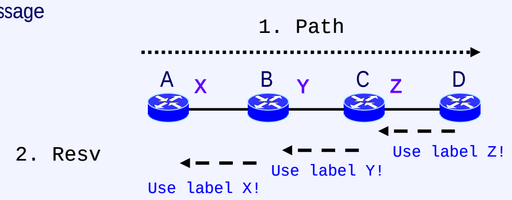
- Path that complies with constraints that differ from IGP can be chosen
  - E.g. maximum capacity, not minimum of hops
- Path computation
  - online (by the router)
    - Typically only the ingress router needs to be aware of LSP constraints
    - No need for consistent routing information database in routers
    - No need for consistent route calculation algorithm in routers
  - or offline (by path computation tool)
- Removing reliance on IGP removes IGP domain restriction
  - RSVP-LSPs can leave AS boundaries
- Possible to establish path which can only be changed by the head end
  - With LDP, any intermediate LSR can change path due to IGP
  - Important for traffic protection schemes such as Fast Reroute

How an RSVP LSP is set up

- Path message from ingress LER with Router Alert option (IP header) (semantics of this option: “routers should take a closer look“)
  - Label Request Object
    - requests an MPLS label for the path
    - causes transit + egress routers to allocate a label for their section of the LSP
  - Explicit Route Object (ERO)
    - contains addresses of nodes which LSP must traverse
    - can be complete path
  - Record Route Object (RRO)
    - requests routers to add their address to the list in this object
    - records path of Path message, i.e. path taken by LSP
    - routers can detect loops if they see their own address
  - Sender TSpec
    - Bandwidth reservation request for LSP from ingress LER

How an RSVP LSP is set up /2

- Egress LER answers with Resv message
  - Not addressed to ingress but to upstream neighbor (which will do the same)
  - This way, the same path is used (IGP does not interfere)
- Content
  - Label Object
    - Contains label to be used for that section of the LSP
  - Record Route Object
    - Similar to Path message
- Soft state: refresh messages needed
  - Path and Resv messages sent periodically



RSVP for MPLS: some more details

- Periodic refresh messages cause significant overhead
  - “Refresh Reduction Extensions“ scheme to reduce this traffic e.g. Summary Refresh Extension: refresh multiple RSVP sessions (LSPs) with a single message
- Optional node failure detection mechanism
  - Hello messages periodically exchanged between neighbors
  - Faster than RSVP session timeouts
- Note: no ECMP in RSVP
  - Once traffic has entered an RSVP LSP, there is no splitting and merging of traffic as it can happen with LDP

To RSVP or not to RSVP?

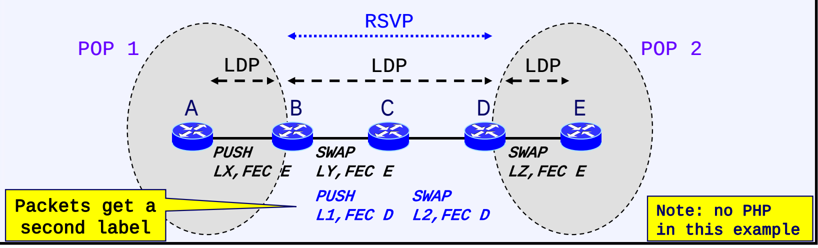
	LDP	RSVP
Ease of configuration	Very easy (automatic neighbor detection, routing from IGP)	Explicit configuration of the LSPs at the ingress router, must know all routers to which LSPs should be established
Scalability	•State proportional to number of LDP neighbors (fully meshed topology: O(n)) •Keepalive / hello messages for limited number of neighbors / sessions •Forwarding state in core LSRs for all FECs, plus additional labels for resilience or ECMP	•State proportional to number of LSPs (fully meshed topology: $O(n^2)$) •Refresh messages for all LSPs •Forwarding state in core LSRs for LSPs traversing them
Support of Traffic Engineering	No	Yes
Support of Fast Reroute	No	Yes

To RSVP or not to RSVP: Applications

- L3 VPN
  - Typical properties
    - No stringent SLAs regarding outage time when a link fails
    - DiffServ classes may be offered, but without related bandwidth reservation in the core
  - Protocol commonly chosen: LDP
- Emulation of Layer 2 services (e.g. ATM) over MPLS
  - Typical properties
    - Bandwidth guarantees required (as promised by ATM)
    - Fast restoration needed when a link fails
  - Protocol commonly chosen: RSVP
- Services requiring fast restoration (e.g. voice)
  - RSVP obvious choice - but since traffic engineering not required, maybe only for some parts of the net!

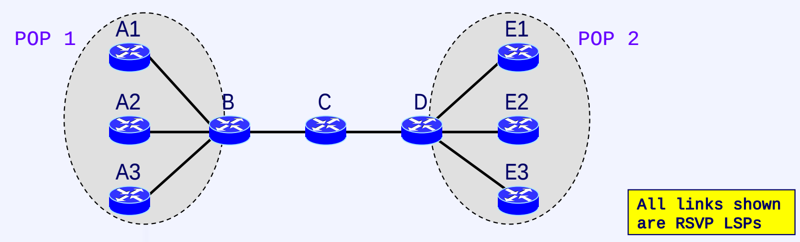
Combining LDP and RSVP

- Sometimes: RSVP needed for its features (e.g. fast reroute) in the core, but not everywhere
  - Practical setting: Point-of-Presence (PoP) consisting of several access routers and one or two core-facing routers
  - LDP can be used instead of RSVP within the PoP
  - Greatly reduces number of LSPs to be considered in the core
    - Relevant if memory becomes a problem, but also easier management



Nesting RSVP-signaled LSPs

- LDP over RSVP not suitable when properties of RSVP needed from edge to edge
  - hence, alternative: stick with RSVP, but aggregate LSPs
    - LSPs in the core are called Forwarding Adjacency (FA) LSRs
    - Core LSPs are unaware of LSPs "inside" it
  - solves the scalability problem, but makes configuration hard again



BGP label distribution

- Border Gateway Protocol (BGP) – most common EGP
  - has to supports multiple address families (prefixes advertised)
  - new address family added for advertising prefix + associated label(s)
  - essential for inter-AS MPLS/VPNs
- Benefits of using BGP
  - ability to establish LSPs crossing AS boundaries
    (e.g. for MPLS-based VPNs having attachment points with multiple providers)
  - BGP is already used; better to add labels to it than to use (and configure) heavyweight RSVP between AS in addition to it
  - plenty of protocol capabilities automatically reused: filtering routing information, controlling the selection of exit points, loop prevention, ..

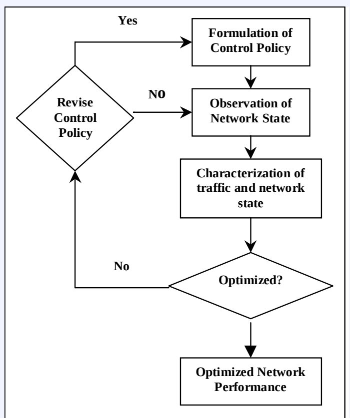
Traffic Engineering

Network vs. Traffic Engineering

- **Network Engineering**
  - Manipulating the network to suit the traffic flow i.e. predicting the traffic flows across the network and subsequently ordering the appropriate circuits and network devices
  - Note: network traffic never match the predictions 100%
  - "Put the *bandwidth* where the *traffic* is"
    - Physical cable deployment
    - Virtual connection provisioning
- **Traffic Engineering**
  - Manipulating the traffic flow to suit the network i.e. moving traffic from a congested link onto the unused capacity of another link
  - "Put the *traffic* where the *bandwidth* is"
    - On-line or off-line optimization of routes
    - Implies the ability to explicitly route traffic

Traffic Engineering Objectives

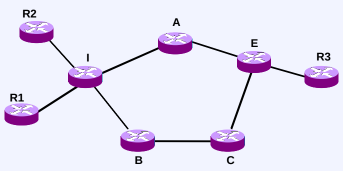
- Traffic Engineering Objectives
  - Map actual traffic efficiently to available resources
  - Controlled use of resources
  - Redistribute traffic rapidly and effectively in response to changes in network topology - particularly as a consequence of line or equipment failure
- Traffic Engineering complements Network Engineering
  - Putting the network where the traffic is
  - Performance oriented: avoid underload and congestion, minimize packet loss and delay, maximize throughput, enforce SLAs
- Adaptive and Iterative Process



Traffic Engineering Process Model

Two application scenarios

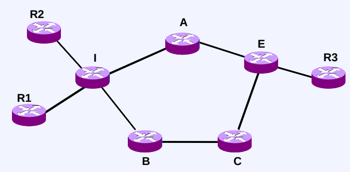
- In example on the right, standard IGP path is always via A
- Various possible reasons and methods for shifting some traffic to the B-C path



1. E.g. assume: satellite link along path via A, customers connected to R1 expect low latency to R3
   - I must distinguish sources: R1 vs. R2
2. Assume capacity is 150 Mbit/s, R1 sends 120 Mbit/s to R3, R2 sends 40 Mbit/s to R3
   - IGP shortest path routing: 160 Mbit/s over A ⇒ congestion
   - Solution: I splits, e.g. 80 Mbit/s across both paths
     - But what if the capacity of B-C is only 50 Mbit/s?
       ⇒ I must know about this!

Application scenario 3

3. Assume: all links 150 Mbit/s except B-C: 50 Mbit/s
   - R1 sends 100 Mbit/s to R3, R2 sends 40 Mbit/s to R3
   - Customers connected to R1 bought standard service
   - Customers connected to R2 bought service with strict guarantees



- Normally, total traffic of 140 Mbit/s can be sent via shortest path
  - Assume link A-E fails ⇒ alternate path cannot support the whole load
- Possibility: protect traffic from R2 via DiffServ – but:
  - Under normal conditions, R1 and R2 traffic should get the same treatment
  - Generally operators try to avoid introducing DiffServ classes (management overhead)
- Alternative: only let R2 traffic use alternate link

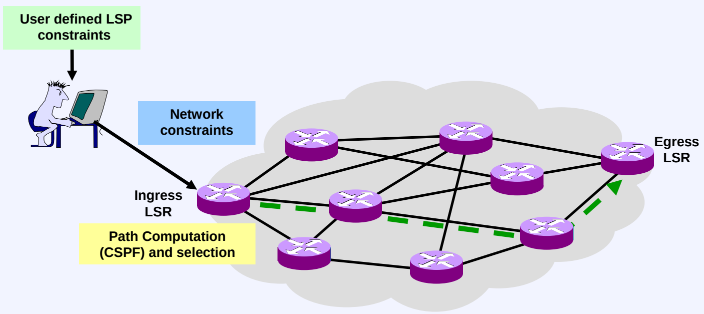
MPLS-TE

- How to solve the problem in scenario 3?
  - (only let more important traffic use alternate path)
- Answer: LSP priorities
  - More important LSPs can preempt less important LSPs (use their resources)
  - 8 priority levels (0 = best, 7 = worst)
  - Setup priority: controls access to resources when establishing LSP
  - Hold priority: controls access to resources for established LSP
  - Preemption: check: setup priority of new LSP > hold priority of existing LSP?
- How to set these priorities by default?
  - High hold priority, low set priority: stable network (no preemption)
  - High set priority, low hold priority: can lead to oscillations
  - Most implementations therefore disallow setting hold < setup for one LSP
- Often, long LSPs are given better priorities than short ones
  - Short LSPs: better chance of finding the necessary resources over an alternate path

Path Computation

- TE consists of two steps:
  1. compute a path that satisfies constraints ("constraint based routing")
  2. forward traffic along this path
- Possible constraints
  - Link properties: bandwidth, administrative attributes ("colors" – e.g. for avoiding high-latency or unstable links), ..
  - LSP properties: max. number of hops, LSP setup priority, ..
- Consider application scenario 2: I must know about small B-C capacity ⇒ information must be advertised throughout the network
- Done by adding TE-specific extensions to IGP: IS-IS and OSPF
  - In addition to link up/down, advertise bandwidth + "colors"
  - Information stored in Traffic Engineering Database (TED) on each router
  - When to send updates?
    - Standard 30 second interval may not be enough
    - Signaling every change (e.g. available bandwidth) may be too much
    - Only signal upon significant changes
      ⇒ trade-off between TED accuracy and overhead

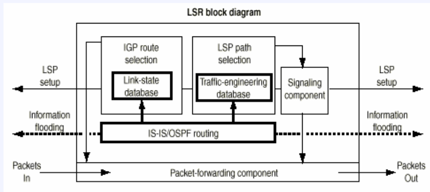
Constraint-Based Routing



- Operator configures LSP constraints at ingress LSR
  - Bandwidth reservation
  - Include or exclude a specific link(s)
  - Include specific node traversal(s)
- Network actively participates in selecting an LSP that meets the constraints

Constrained Shortest Path First (CSPF)

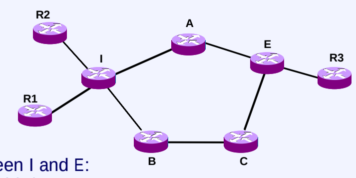
- Calculation of shortest paths like conventional SPF, but with rules
  - E.g.: exclude blue links, include red links, min. bandwidth 50 Mbit/s
  - As with SPF, by default only one shortest path chosen
- Based on data in TED, which is built from IGP information
  - Computation restricted to an AS
  - Result can be wrong if TED is outdated (slow advertisements)



- Paths change, updates arrive periodically ⇒ recompute
  - This is called reoptimization; trade-off: path optimality vs. net stability
  - Stability + foreseeable behavior are usually the most important goals; hence turned off by default in most vendor implementations
  - Can usually be enabled with different granularity levels (periodic, event-driven or manual)

CSPF: using multiple paths

- If CSPF derives multiple equal cost paths: rules ("tie-breaking") apply
  - Random, least-fill, most-fill



Least-fill example:
- Assume all links are 150 Mbit/s
- Metrics set such that paths via A and via B-C have equal cost
- Assume that three LSPs must be set up between I and E: LSP1: 75 Mbit/s; LSP2: 75 Mbit/s; LSP3: 150 Mbit/s
- Bandwidth suffices, but will not be used with least-fill algorithm if LSPs must be set up in order LSP1, LSP2, LSP3:
  - LSP1 placed on I-A-E
  - LSP2 placed on I-B-C-E
  - LSP3 cannot be placed

- Reason for this problem: lack of information about future reservations
  - Solution: offline path computation

The TE Control Plane: RSVP-TE

- RSVP with Traffic Engineering extensions used for setting up path
  - Reminder: path specified in Explicit Route Object (ERO) of Path message
- For TE, more information is needed
  - TE information that intermediate nodes must keep track of (e.g. bandwidth requested by LSP)
  - Information for path setup such as LSP setup and hold priorities
- Resv message (= reply to Path message) causes admission control at each node because
  - LSP may not have been computed with CSPF
  - Even if it was, state of resources may have changed in the meantime
  - CSPF may have been based on outdated information in TED
- If successful, information fed back to IGP to update state in all other nodes
  - May not be immediately distributed
- Note: goal is, of course, to make data plane match control plane
  (Does not always have to be an exact match - e.g. overbooking: announce higher available resources in control plane than data plane if resources are never fully utilized)

What if the resources do not suffice?

- Try preemption
  - If that fails, send error message to head end
- Upon failure, head end recomputes path
  - If TED still outdated: same result as before, reservation will fail again
  - IETF standards foresee no solution to this problem
  - Practical solutions:
    1. Exclude the link from CSPF computation for a while
       ⇒ simple, localized to head end, but TED is not updated, failure propagates
    2. Announce admission control failure via IGP, regardless of throttling mechanism (which should reduce flooding load)
       ⇒ does not have problem above, but
       1. computation must happen after delay (make sure TED is up to date)
       2. relies on help from a downstream node which may not implement the same behavior (no standard)
       3. generates extra flooding traffic

    … and they can be combined.

Make-before-break

- Reoptimization finds better path based on TED
- Switching must happen without traffic loss
- This is done by
  - first setting up new LSP
  - then tearing down old one
  - then shifting traffic
- This means that both paths must be kept for a while
  - reserve twice the resources? Likely to fail
  - Let these two paths LSPs share the resources they reserve
  - LSRs must be informed: shared explicit reservation style in RSVP

The TE data plane

- As in examples, easiest way to map traffic: static routing (manual)
  - not scalable (effort per operator scales linearly with LSPs)
- Alternative: incorporate LSPs in routing
  - regard LSPs just like other links, associated with cost metric, in BGP
- LSP can become a shortcut through an AS (for transit traffic) by setting up LSP between AS Border Routers (ASBRs)
  - transit traffic forwarded via MPLS labels; no routers inside an AS need to know about destinations outside (ASBRs know) ⇒ BGP-free core
  - tight control over path of transit traffic in domain, e.g. to have it forwarded over dedicated links (to ensure that SLA is kept)
- What if we let IGPs use LSPs?

The TE data plane /2

- Mixing LSPs with other links in IGP = mixing paths determined by constraint-based routing with paths determined by IP routing
  - Even when TE is only applied for a small part of the network, LSPs are taken into account in the whole network
- Two behaviors
  - Let head end use LSP in SPF calculation
  - Advertise the LSP just like any other link via IGP
    - More efficient, as LSP is taken into account by “distant“ nodes
    - Can lead to strange behavior because entire path calculation is a mix between SPF and CSPF
    - Example: consider LSP I → E
    - I advertises LSP with low metric
    - Tells LSR about it...