Material for lecture #8 & #9

MPLS (MULTIPROTOCOL LABEL SWITCHING) - SOME PRELIMINARY THOUGHTS

According to RFC2702 "Requirements for Traffic Engineering Over MPLS" (http://www.ietf.org/rfc/rfc2702.txt)

> Multiprotocol Label Switching (MPLS) integrates a label swapping framework with network layer routing. The basic idea involves assigning short fixed length labels to packets at the ingress to an MPLS cloud. Throughout the interior of the MPLS domain, the labels attached to packets are used to make forwarding decisions (usually without recourse to the original packet headers).

One of the main applications of MPLS is traffic engineering and as the RFC states, Traffic Engineering is concerned with performance optimization of operational networks. However, In general, the future of MPLS is an open question, and cannot necessarily be compressed into one term. Thus, in order to give some more general outlook over the possible application area of MPLS, let us start with the check list we have used to evaluate the possible advantage of resource reservations.

1. Applications' need of minimum bandwidth
   Perhaps, but then we have to be ready to use MPLS for individual flows - and that is quite unlikely approach.

2. Virtual Private Network or Leased Lines
   Yes, this could be a relevant issue (we will discuss about this later)

3. Pricing model [that indicates that resource reservations are used inside the network]
   This might be possible if the pricing model is related to items 2 or 10 (but definitely not in case of end-user pricing).

4. Need to know service availability beforehand
   Perhaps in case of very large reservations

5. Emergency calls
   This cannot be the main reason for MPLS

6. Need to limit traffic sent into the network
   Depends what is the *reason* for limiting incoming traffic

7. Technological reasons (reservations may improve the use of network resources in some specific cases)
   This might be possible, but even then the use of MPLS for reservations should be justified somehow. Anyway, the original idea of MPLS is to design a protocol that is independent of underlying technology and protocols. Thus, even though it might turn out that MPLS is advantageous with certain network technology, it hardly can be the primary reason and guideline for developing MPLS.

8. The need to favour existing connections over new attempts
   Perhaps, but then we have to be ready to use MPLS for individual flows, and once again my personal opinion is that MPLS should not be aimed to individual flows.

9. Common beliefs
   Common beliefs may lead to think that something is important and advantageous, though we personally do not entirely understand the justification of the technology or protocol. There seems to be an MPLS school that has strong faith on the technology, see e.g.

10. Contracts between operators
    Yes, this could be relevant (relates to item 2)

As to the common opinions about MPLS let us just take two more examples:

> " It is expected that MPLS will be a crucial strategic element in addressing the ever-present scaling issues faced by the Internet as it continues to grow." But this is from a MPLS conference advertisement; so let us not take it took seriously.
>
> "Until MPLS is fully supported it brings nothing new to our customers," said [Mika] Uusitalo [Sonera]. "We will not implement it until it really brings us value," he said, confirming a growing Scandinavian trend against MPLS.
> http://www.totaltele.com/view.asp?ArticleID=32975&Pub=CWI&CategoryID=705

The next pages are partly based on an article "MPLS: the Magic Behind the Myths", G. Armitage, IEEE Communications Magazine, Jan. 2000. According to Armitage the possible justifications for MPLS are (actually this list is essentially the same that can be found on http://www.ietf.org/html.charters/mpls-charter.html):

- IP QoS
- Gigabit forwarding
- network scaling
- traffic engineering

However, gigabit routers have practically removed the forwarding problem, and also a common, though not undisputed, opinion is that MPLS is not the right tool to solve IP QoS problems in general. Therefore, a tentative conclusion could be that if MPLS really provides some significant advantages they are related to two aspects:

- Ability to forward IP packets over arbitrary non-shortest paths
- Emulate high speed tunnels

Although these are quite concrete issues, they may require some more general justification. If we take the second item, the question is what is the point of tunnels. Armitage expressed the possible reason by stating that "Relative or absolute protection from other traffic is desired". What is "protection" and what is "other traffic"?

An emotionally appealing idea is that other traffic means everything else than *my traffic* and (absolute) *protection* means that my traffic remains (totally) intact whatsoever happens inside the network. What then is relative protection, else than prioritization? Higher priority traffic should be protected from the lower priority traffic - but unfortunately, this principle is not as attractive as the absolute protection, because someone else may have higher priority than I. Maybe this is the reason why protection sounds better than prioritization. On the other hand, if my traffic has high priority then someone else must have low priority (or if everyone has the same priority we are once more in the starting position).

After going through these considerations a couple of time, we may be ready to soften the requirement of absolute protection and introduce some new concepts like isolated traffic classes (see, e.g. http://members.home.net/garmitage/things/ngn-diffserv-110399r2.pdf, slide 4 that illustrates nicely the way of reasoning). But what is the point here, either from individual user or service provider viewpoint? Because a traffic class does not mean an individual user, the protection is inevitably relative for one user though the protection could be hard for the traffic class. The only additional value for the user has to somehow emerge from the grouping of users or applications. Indeed, you may rely more on the members of your group than outsiders.

But what is group or class and why should it be isolated? I can see two main classification principles:

- certain customers form a group (independent of the application)
- certain applications form a class (independent of the user)

The most apparent group is formed by the employees of an organization when total network capacity is divided among several organizations (note that the organization does not necessarily mean an independent company but it can be, for instance, a unit inside a large company). The main requirement for this grouping is related to the pricing model of the system: if the organization pays a monthly fee and all employees then share the resources bought by the organization, it is reasonable to suppose that the employees should form an isolated group. However, this reasonability does not imply that a hard isolation between all groups is the most efficient implementation. Rather, I would say that this grouping indicates that the utility function is somehow tied to the whole group rather than individual members of the group.

Whether or not this shift from individual utility functions to a group utility function is advantageous from the viewpoint of individual user, depends on the behavior of the users and the method used to divide the capacity inside the group.

Then the other possibility, classification based on applications, is somewhat more difficult to comprehend. Why should I rely on a user just because he or she is using the same application as I? What is the connection between me and another arbitrary VoIP user and how this connection changes when the other users change from a VoIP to a web application? The answer may be that web applications are using TCP and therefore tend to use any available capacity while VoIP is intrinsically non-adaptive. Or, alternatively, the answer may be that VoIP packets are, on average, more valuable for the service provider than web packets. Finally, we may be tend to think that other users of the same application are somehow similar to us, and therefore, are more reliable than other users.

From service provider viewpoint the last item is somewhat irrelevant, because it does not give any clue how much capacity should be isolated (or reserved) for each traffic class. As to the middle item, a better approach seems to be to use prioritization rather than isolation, because the isolation may well lead to a situation in which less valuable TCP packets are transmitted instead of VoIP just because the isolation algorithm has not been able to make optimal reservations.

What about the first item - it seem to offer reason for isolation because the behavior of TCP is inappropriate from the viewpoint of VoIP? Yet, it is not clear that isolation is the best tool to solve the problem. Moreover, if VoIP users tend to think that TCP applications behave inappropriately (since they do not adapt), in the same way TCP users may think that VoIP applications behave inappropriately (since they generate packet losses). Thus these opinions do not provide any clear conclusion how the traffic should be treated, or how much isolated bandwidth each user is entitled.

But let us return to MPLS. What is the relationship between these considerations and the supposed needs for MPLS, that is

- Ability to forward IP packets over arbitrary non-shortest paths
- Emulate high speed tunnels

If the operator can decide the right size for each tunnel, then it can use MPLS to realize the tunnels. But that does not mean that MPLS somehow clarifies the question how the sizes are determined.

The clearer case is the high-speed tunnel reserved for specific user group, and further we may well call a coupling of tunnels as VPN. In this case the task of the operator is just to establish the tunnels between each pair of end-points based on the contract made with the user group or organization. In addition, the operator shall control the traffic sent by each group in order to protect other users and groups. Although this model appears straightforward, it does not necessarily use resources efficiently (most of the remarks made in lecture #4 are valid here as well). Note also that a model in which certain capacity is reserved for group of applications is quite problematic in the light of previous considerations.

Then as to the other item, non-shortest path forwarding, the apparent goal seems to be load balancing. But what is the connection between load balancing and grouping of users or grouping of applications. Although someone may argue that certain route is more suitable for certain applications (e.g., because of significantly differing delays), in high-speed core network that is quite unlikely situation. Then if the shortest path is much slimmer than a longer path, the OSPF should primarily select the longer path with higher capacity. Only if there are a number of slim links, it is clearly advantageous to balance the load between them. But then we, once again, lose the connection between alternative routing and traffic or user classes.

But what could be the expected advantage of load balancing? In a short document and without real traffic it is impossible to assess exactly the advantages of load balancing. However, we can try to understand the fundamental limits and most likely outcome even by simple example. Let us look at the figure 1, which shows a network with 5 nodes. Actually we may think that there are more nodes in the network but they do not provide realistic routes between the 5 nodes presented in the figure.

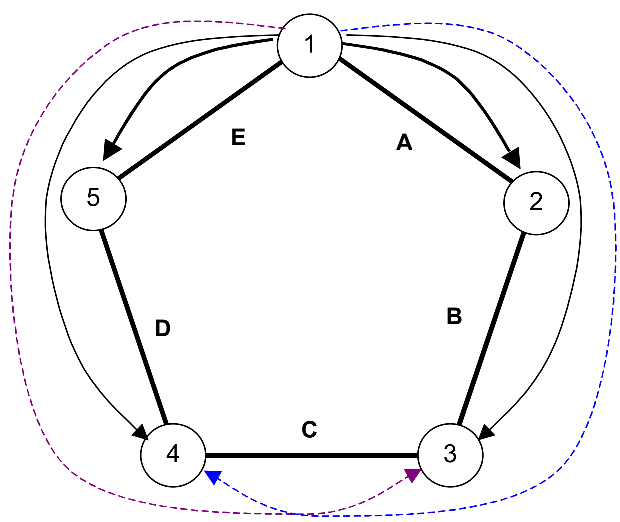

Fig. 1. A network with 5 nodes, primary and secondary paths

The advantage of the simple ring topology is that both the primary and secondary paths are clear. For instance, the shortest path from node 1 to node 2 is, of course, via link A and to node 5 via link E. Because of the large difference between the number of hops (1 vs. 4) the alternative path is not considered realistic unless the short path is totally unavailable. Therefore, we concentrate here on the routes from node 1 to nodes 3 and 4 (and similar cases between other node pairs). Because in these cases the difference is only one hop, it is quite likely that sometimes it is advantageous to transmit some part of the traffic through the longer route with 3 hops.

The most significant issue that determines the gain available by using load balancing is how evenly the load is divided among pair of nodes. Or actually, if the link capacities are different the main matter, apparently, is the variations in average load between links. However, in this brief study the capacity is the same for each link. In our case there are together 20 different traffic streams, that is traffic from each node i to each other node j. For simplicity we may suppose that the average traffic for each of these streams is a random variable with a given mean (A) and standard deviation (σ). For simplicity the average traffic of a stream is here always the same (10) while the unevenness of traffic distribution is defined by standard deviation (σ). If σ is small (say, less than 0.5*A) the demand could be considered even, and if σ is large (say, more than A) the demand could be considered uneven.

It is intuitively clear that load balancing is likely to be useful when the load is unevenly distributed. But how should we define the possible usefulness. There are various possibilities. Here the approach is as follows. First we fix the traffic demand matrix by using lognormal distribution for given A and σ. But because the cases must be random, we of course get some other values for A and σ, and those number are more relevant than the original values used to define the demand matrix. Then we fix also the link capacity for each link (C) in a way that the packet loss ratio is approximately 2%. The packet loss ratio is roughly approximated by a model in which the load distribution on a link is always log-normally distributed with standard deviation of 0.25 times mean traffic, and the packet loss ratio is supposed to the same as the probability that traffic exceeds the link capacity.

However, the actual model for traffic distribution and packet loss ratio is not very critical because what we do is that we compare the required capacity needed to achieve the same packet loss ratio. For instance, if the load balancing approach yields 2% packet loss ratio with capacity 50 units, we search the capacity that produces the same packet loss ratio without load balancing. If that capacity is, say, 58 units, we may conclude that the saving achieved by load balancing is 8/58 or 14%.

One of the problems is to find the optimal load balancing approach. Here we simply use MS Excels Solver tool. Of course, there is no guarantee that solver can find the ultimate optimum, but that is anyway the situation in reality. Moreover, the results given by the solver-tool appear consistent and reasonable.

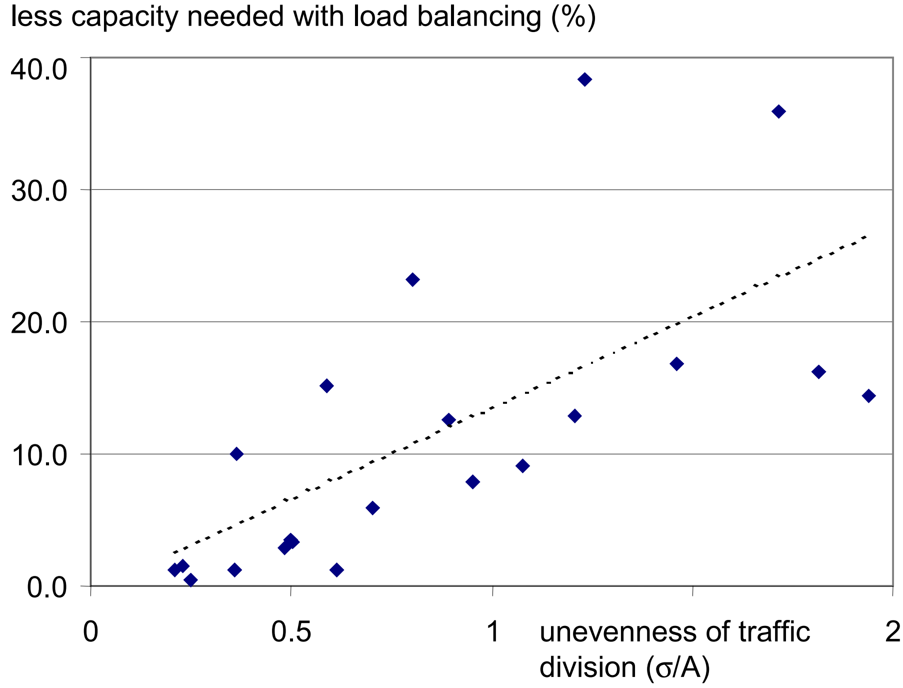

Fig. 2. Capacity gain obtained by load balancing as the function of unevenness of traffic distribution

Figure 2 illustrates the strong correlation between evenness (or unevenness) of traffic division and the gain attainable by load balancing. With a uniform load division ($\sigma/A < 0.25$), there is practically no gain at all, whereas with very uneven load division ($\sigma/A > 2$), the attainable gain could approach 50%. A realistic region could be somewhere between from 0.5 to 2 (though I have no real information about this - see also slide #6).

A special example is that one traffic stream is much larger than the average traffic stream. For instance, the magnitude of all traffic streams can be 10 exept one stream with magnitude of 35.2, in order to produces $\sigma/A = 0.50$. In this case the optimal (or nearly optimal) load balancing yields tte following packet loss ratio with link capacity of 60

$E\{P_{loss}\} = 2.5\%$ if the exceptional stream is between adjacent nodes

$E\{P_{loss}\} = 3.8\%$ if the exceptional stream is between non-adjacent nodes

With OSPF we need more capacity to obtain the same average P_{loss}. According to my calculations the corresponding figures are

C = 69.4 for adjacent nodes -case

C = 71.5 for non-adjacent nodes -case

Thus the gain obtained by load balancing is about 15%, which is more than what Fig. 1 indicates. The reason probably is that there is not so clear gain to be achieved when all streams are differing as in cases where only one stream is exceptional. Then if we want to have an unevenness of $\sigma/A = 2$, one traffic stream should be as high as 172, if all others are 10. Though this is possible, an evident approach is that the link capacities should be adjusted accordingly.

But how relevant actually are these results? There are a couple of reasons that may considerably weaken the benefits of load balancing.

Firstly, the above calculations were based on the assumption that all traffic streams were exactly known in advance, in a way that optimal load balancing can be found. Of course, that cannot be true in reality, because load distribution varies all the time. Still, in the core network with large traffic aggregates the load figures likely are stabile and predictable enough to make it possible to balance the load between different links.

Secondly, we assumed that the traffic can be infinitesimally divided between two routes. This is usually not true, because the packets of one flow should normally use the same route. Further, with MPLS and similar schemes, the distribution of traffic among differing labels is usually fixed, and the granularity of this distribution limits also the possibilities of load balancing.

Thirdly, no extra cost whatsoever is assumed for the implementation and management of the load balancing system. Evidently, this is quite unrealistic assumption because MPLS (or anything similar) requires a management system and qualified personal to use the system.

Fourthly, the above calculation does not include the possibility that the capacity of individual links (and routers) were updated independently, instead all links had to have equal capacity.

Consequently, if we rely on the results shown in Fig. 1 we may assume that the average gain of load balancing (measured in the saving of link capacity) could be 15% compared to OSPF. However, because of the unpredictable traffic variations and other practical problems, it is quite unlikely that the real saving could be more than 10% even if we ignore the costs related to the use of the load balancing system. In consequence, the final question is about the cost structure, for instance, which one is more expensive, 10% overall increase in link capacity, or the implementation and management of a new system with advanced load balancing. My impression is that the latter option is reasonable only if link (or perhaps router) capacity really is an expensive resource. Note also, that 10% savings in capacity is very rapidly consumed by incessantly increasing traffic demand, perhaps in less than 1 month.

Of course, there are basically different approaches between a pure OSPF and "optimally managed" load balancing. One example is presented in

http://nero.informatik.uni-wuerzburg.de/cost/FinalSeminar/Internet/roch.pdf

in which a traffic-aware routing system is briefly described. Another simple and attractive system could be that there is permanent division of traffic when there are two routes with equal or almost equal length. In our case, we may route 80% of the traffic from node 1 to node 3 via node 2 and 20% via nodes 3 and 4. Unfortunately, this approach does not appear to provide any clear advantage compared to OSPF; sometimes OSPF is better sometimes 80/20 division, but it is hard to find any systematic difference. Probably the situation is somewhat different if there are two routes with equal length, so that 50/50 division between the routes is probably the best approach if there is no real-time knowledge about load situation.

Finally, it is usually assumed, or at least that has been my assumption, that alternative routing is most useful with moderate load. On the one hand, with very high overall load, evidently the most efficient way of using resources is to use the shortest path (if any) for all packets. On the other hand, if there is no lack of resources, there is no gain to be achieved by any other more complex method than OSPF. Based on the same model used to draw Fig. 2, the first assumption related to high load appears to be valid (although the model used for the evaluation is not very good with high load). On the contrary, the difference between required capacity (C_{OSPF}/C_{LB}) is basically independent of C_{LB} for a given traffic demand, if packet loss ratio is less than 1%. Of course, we may argue that if the packet loss ratio is negligible with OSPF there is no need to use load balancing - but still if we keep the expected packet loss ratio constant, it is possible to save some resources by load balancing. The practical point, however, is that there is no *real* saving of resources if the resources have already implemented and available without any extra cost. Therefore, from utility perspective there is only a negligible gain to be attained if the packet loss ratio is small.