A Design of Bi-Directional Path Setup to Enhance Fast Network Restoration Support for MPLS in Optical Networks

Hen-I Yang
heni@andrew.cmu.edu

Problem Definition

Great amount of current protocol research effort in optical networking has been placed in adapting traditional switching protocols such as MPLS, but ignoring some of the original performance characteristics promised by optical networks. One such key characteristic is fast restoration capability that remains barely untouched by researchers. Some extensions needed to be made to MPLS in order to make optical networks offer competitive performance, and this project will study the necessary changes, with a proposal and implementation on one such modification to MPLS. This project will study one of the promising changes necessary, with the proposal and implementation modification to MPLS protocol.

Background

Restoration can be generally stated as the process of re-routing label switched paths (LSP) between hosts in an optical network when parts of the link go down. Thus, it involves automated re-routing and re-setup of circuits when a part of the link fails. Fast and automated restoration is a desired characteristic of any networking system – it aids the return of the system to a normal balanced state of operation. Restoration involves the following major steps:

1. Detect the state of adjacent links near the node
2. Find a suitable re-routing path circumventing the link that has gone down.
3. With control messages to the intermediate nodes, setup the link with channel assignments etc.

The first step is very important in any networking system. Early detection of a network failure is essential to achieve reasonable performance goals. The second step involves finding an alternate route to the destination based on the topology/map known about the network or dynamic network topology mapping. The third step involves the automated signaling of the intermediate nodes to setup the new connection. Because of the bi-directional characteristics of optical links, MPLS protocol has to be modified to support such fast
bi-directional restoration setup.
Original MPLS implementations do not support such fast restoration, and most of the work in restoration is done manually. Thus an automated system for fast restoration is needed if the technology has to be competitive in the long run.

Implementation Approaches

According to Doverspike and Yates [1], there are four major categories of solutions being brought up, based on the relative timing of executing three major restoration activities compared to the time network failure occurs. Total-Redundant method will calculate path, assign channel and setup a separated backup channel in advance, and set aside until event of failure triggers the cross-connect to switch toward backup channel, it is simple but very inefficient. Pre-planned Cross-connect Maps would calculate path and channel assignment before hit by failure, but mapping channel to cross-connect after failure has been identified, which makes it extremely difficult to implement. Fully Dynamic Restoration takes all measures only if a failure actually hits, but it is in general too slow for efficient recovery in optical network. The forth category calculates the path before failure detected, but performs all others after, and this might be the most promising method. There are also various methods about how to calculate a restoration path, varied from optimized calculation to adapt first available then re-optimize later.

Literature Research

According to Doverspike and Yates [1], there are three categories of crucial issues need to be addressed when adapting MPLS to distributed optical network restoration. First, how to make restoration process work when failure occurs, how to complete restoration as rapid as possible so as to match the expectation toward optical network, and how the operation of network settle to normal after failure has been repaired. One interesting difference is that optical connections are bi-directional, different from unidirectional connection in ordinary IP networks, and conflicts may occur in some scenarios involving optical networks that would not happen in IP networks. Second, connection admission control (CAC) is the process about path selection and network restoration assessment. There are many crucial issues about how to diversely rerouted the connection from original service path to secondary one using physically disjoint connections and make them aggregated into a restoration path. Clear definition of restoration objectives and the assessment of where objectives are achieved are also of great importance. The last but not the least, restoration process has to
take into consideration about capacity of restoration path.

MPLS has many advantages suitable for bridging IP networking into the world of optics. It simplifies packet forwarding, uses efficient explicit routing, and enables traffic engineering and service differentiation, and even provides features such as load balancing and aggregate network bandwidth between any pair of routers [2]. Bernstein, Mannie and Sharma [4] points out that SDH and SONET networks already offer various protection options at transportation level, such as unidirectional path-switched ring protection. Many of these rings and line protections also allow extra traffic to be carried over the protection lines when they are not used, when no failure has occurred and traffic has not been switched away from original service path. Due to variation in nature of packets transmitted, it may be desirable to route some connections over lines that support protection of a given type, while others could be routed over unprotected lines or take route for extra data over restoration lines. All these protection mechanisms are useful features for SDH/SONET in transportation layer, but we still need some modifications and enhancement to MPLS that can deal with failure in service level, such as routing failure and other services.

Many documents address some fundamental differences between optical network and traditional IP network [1, 2, 3, 4]. Some important ones are discussed as following. Technically speaking, it is not feasible to establish zero-bandwidth paths for later use in optical networks, while it is trivial to think about what to handle predefined restoration path before failure hits in traditional packet switching. It only requires a simple and fast switch to restoration path once failure hits. As mentioned in previous discussion, optical networks require bi-directional connection setup, which is one of the major differences compared to traditional network. The bi-directional connection setup is likely to introduce a flood of setup messages within short period of time, which should be stochastically unlikely to happen if following regular path setup, but almost unavoidable during path restoration, hence can potentially lead to conflict in channel requests. Also, in regular IP packet networks, tags are placed on each packet for identification of different channels, and each tag is only meaningful locally as in most virtual circuit routing algorithm. When MPLS is introduced to optical network, wavelength is the most natural and effective equivalent to tags, but if neighboring OTS does not have the capability to convert signals from one wavelength to another, the channel selection process becomes nontrivial since the wavelength has to be carried out throughout the entire transmission process. The tag (wavelength in optical networks) has to be defined somewhat with global acceptance to avoid conflicts due to repeatedly used tag by different paths on some segments along the way. Because people expect fast failure recovery at the same order of performance as in SONET, the restoration process applying original RSVP protocol, which requires building forward path first, then the
reverse one is too slow.

Doverspike and Yates also categorize different approaches of providing optical network restoration into four groups, based on when to perform each of these three crucial tasks, calculate path, calculate channel assignment, and cross-connect. Each has its pros and cons, and each vendor chose to pursue different strategies that make standardization a challenge. Baroni and his colleagues [3] take the approach of doing everything about restoration in advance by plotting an overlay networking architectures, with separately designed service and transport layers, with protection and restoration mechanisms deployed and executed independently from each other. Under this layout, IP virtual topology and MPLS path layouts are designed first, with MPLS based protection/restoration used to protect against router failures. No additional protection/restoration mechanisms were implemented in transport layer (except those provided by SONET/SDH) therefore everything related to protection/restoration is performed in service layers using IP/MPLS.

In service layer, several MPLS-based protection and restoration solutions can be implemented in additional to ordinary OSPF routing restoration. Two of those explored by Baroni and his colleagues are 1+1 and 1:N dedicated protection with shared restoration. With this solution, it calculates the restoration paths in advance, and actually uses them to transmit data, and the difference is that 1+1 dedicated protection sends duplicates of whatever being transmitted in service path on restoration path too, causing inefficiency in bandwidth utilization. While in 1:N dedicated protection, preemptable data is sent via restoration path. Destination edge router gets both primary and preemptable data if both path works correctly, and also N primary paths share the same restoration path, both measures can maximize the bandwidth utilization efficiency. Once original service path failed, destination router would switch after requesting the source edge router to switch to restoration path too. If any preemptable traffic is being delivered on secondary path, it is dropped. 1:N dedicated protection comes with cost of slower restoration process and possibility of data loss during restoration process. The restoration time is expected to be much longer than 1+1 case, with the order of several seconds compared to the order of microseconds when using 1+1 dedicated protection scheme, which is considered to be a penalty incurred for increased bandwidth efficiency.

It can be concluded that knowledge of the fiber layout is the key of implementing the service layer architectures. In additional to that data channels, a carefully designed mechanism for signaling is needed, because in case of complex network, signaling could flood the network if a fiber cut occurs. According to Baroni and his colleagues, restoration performance and scalability of service layer networking architectures of their proposals are crucial issues still to be determined, but the advantage of adapting architecture that make
restoration process diversely routed with aggregated channels and link disjoint physical paths can already be shown in their research. Other measurement of performance include the protection time, which is expected to be proportional to the difference in delay offered by primary and secondary (protection) LSPs. 1+1 dedicated MPLS protection is achieved by diverse LSPs, and no sharing of restoration capacity is attained. It has a low cost for restoration because intermediate nodes only perform static wavelength routing.

Both of these dedicated protection mechanisms have limited packet-level reconfigurability via the routers at the source and destination point-of-presences (POPs), but neither of them have wavelength reconfigurability available, although this feature is becoming important. Dedicated MPLS protection reduces cost of traditional overlay design by 35%, and provide a much better restoration time as compared to traditional overlaying structure [2].

Among all four categories of restoration strategies, Baroni and his colleagues were concentrated on first category (Total-Redundant Method), and have concluded with some preliminary result on its design concerns, engineering tradeoffs except the final performance comparison. Category two restorations also known as pre-planned cross-connect maps has been proven to be very difficult to implement as indicated by Doverspike and Yates. Category four known as fully dynamic restoration is flexible in dealing scenarios of complex failure and cost less in planning, but its performance is too slow to meet the requirement on optical network. This leads to a belief that a restoration mechanism in Category three that calculate path before the failure, but assign channels and cross-connect to them only after it happens is a more promising option.

Bi-directional Path Setup Process

Compared with all the other options such as a full scale comparison on efficiency between different categories of restoration strategy, or designing a more effective algorithm for restoration path discovery, the further investigation of bi-directional path setup process seems to be more elegant and effective when considering the amount of changes needed and compatibility to original version of MPLS and RSVP.

Original path setup process in MPLS using RSVP works as following, the upstream INGRESS router initiates a PATH request message to the next hop along the path. The transit router that receives this message checks on the available resources in the incoming interface, outgoing interface and within the router itself against the requirements encapsulated in PATH message. If all the required resources can be satisfied, the PATH request message is passed to the next hop along the path. If the request can not be met, a
PATHERR message is generated and transmitted along the incoming path back to INGRESS router. If all the resource along the path can be satisfied, the PATH message will eventually get to its destination EGRESS router that also checks whether resources can be granted. EGRESS router then transmits a RSEV message back to INGRESS router along the established path. The entire path setup process is completed once RSEV message reaches INGRESS router, and the following packets can be label switched to its destination after this setup process. The entire process takes a round-trip time plus the processing delay from all the routers along the way.

The optical network poses a different challenge to original MPLS path setup process. The optical links are bi-directional in nature, as mentioned in the section of literature review. To allow optical network working properly, setting up a bi-directional path is a necessity. The only way to create bi-directional path using original RSVP messaging would be to generate an extra request of PATH message in reverse direction initiated by Egress router once the original one hits the destination. Even with intervening RSEV and PATH messages in upstream direction, the path setup time would still takes 1.5 times of round-trip time plus processing delay in optimal situation. Not to mention the high overhead of doing PATH-RSEV messaging twice, as well as unpredictable path setup result caused by resource exhausting along the routers on the path or possibility of upstream path being re-routed to other routers.

This could become a problem during optical path provisioning, but what it really hurts is the performance during restoration process. The extra 50% of time needed to establish an optical path could means an extra 26 ms if the link is across the continent of North America, resulting in an extra lost of 1G bits of information during the restoration process in optimal situation. The overhead of doubled messages in both upstream and downstream directions for each link could cause a chaos and contention that could result an even longer delay in restoration.

Our solution to attack this problem is to come up with a bi-directional path setup process that can be used to enhance the path setup process in provision and restoration time. The new bi-directional setup process works as follows. The INGRESS router would still initiate a PATH message just as original RSVP, but instead of waiting for RSEV until the PATH message pass all the way to EGRESS router, it includes a suggested reverse wavelength that can be used for reverse link as a part of PATH message. The transit routers, which receive the PATH message, check for the resources requested in the PATH message, but this time allocate double the bandwidth for paths on both directions, and check if suggested reverse wavelength appended to PATH message is available at the interface. If any of these resources is not available, or there is a conflict in suggested reverse
wavelength, the router issues a PATHERR message directed toward the INGRESS router. If all the requests can be fulfilled, the transit router then forwards the PATH message to next hop, while attaching a suggested reverse wavelength for the link to its next neighbor. It will also send a RSEV message to its upstream router in order to confirm its success in reserving resources for the path. When PATH message reaches EGRESS router, it does exactly the same checking as any transit router, and send RSEV message to its immediate upstream neighbor. Since the resource needed for both directions are confirmed along the PATH messaging, there is no need for further delay on RSEV messages.

This approach guarantees such a promising result. It reduces the setup time to exactly one round trip time plus processing delay at each node, which is 50% less than prior solution. The amount of messages exchanged is reduced to half from 4(n-1) to 2(n-1) assuming there are n routers along the path. It already looks good at provision time, but what it really helps is the restoration process. The 50% reduction is setup time is magnified by the half the number of message exchanges needed, which could greatly reduce possible contention and control packet dropping problem if a link or node fails and all related paths try to restore the path at the same time. The detection and notification of failed link is also further reduced. If the process of setting up two unidirectional links is completed, both setup procedure would take the same amount of time to notify all the routers along the path would be the same. However, if a router fails while the reverse unidirectional link is still in setup process, the time to notify all could be more than twice larger than bi-directional procedure.

Simulation

Simulation Tools

Several simulation tools were being considered for this research project, OMNET++, JavaSim, NS-2 are some of the tools had been considered seriously. After examination of documents and tutorials, with hand-on trial on potential simulation tools, it is decided that a simulator will be written in C++ with specific features to perform RSVP path setup simulation would be used instead of any of the tools mentioned above.

The reason is because that it is confirmed that OMNET++ does not provide RSVP installation, and doubtful whether OMNET++ has the capability of correctly perform real RSVP-TE simulation. Similar conclusion is reached against JavaSim, although it claims providing complete RSVP protocol suite, but it is nowhere to be found. It also has some reliability issue. NS-2 is a highly appraised simulation tool providing an open-source RSVP implementation. But the complexity of compilation and setup is an obstacle. During our trial process, the latest version of NS-2 has been successfully installed, but was not able to
integrate RSVP patches. It is then decided that it would be easier if implementing a simulator written in C++ that has only those features relevant to this study. It would be easier to control over the simulation process and highlight the main points to be discussed.

### Assumptions and Simplifications

The simulation is built upon some assumptions.

- This test tool is designed specifically for optical network. Therefore all the fields used for non-optical network are omitted. All the error messages, primarily RESVERR messages that reject certain types of non-optical links have not much to do with our simulation.

- It is assumed that link delay specified in routing table takes care of both propagation delay, and processing delay.

It also incorporates the following simplification.

- Removal of unnecessary fields: The simulator is built to demonstrate the idea of bi-directional path setup, all the irrelevant fields not affecting the effectiveness of simulation is removed.

- The use of Path_State_Removed flag: This simulator adopts the idea of Path_State_Removed flag from Internet draft on RSVP extension. Which means the PATHERR message raise the flag that asks all the routers along the way that have reserved resources for this path should invalidate all the reservation without further PATHTEAR message from Ingress router.

- Routing algorithm is built-in and well known throughout the network: Since this simulation is mainly focused on showing effectiveness of bi-directional path setup, it is assumed that some how the routers along the way knows which is intended next hop.

- The only “resource” being considered is availability of wavelength, which is used as tagged label. All the other resources in real network such as bandwidth, availability of protection link, security and privacy issue, etc is not considered in this simulation.

- No end host routers are presented. The only routers that are shown in the simulation are edge routers and core routers. Everything outside of core network island is not of the concern in this study.

- Assume that smart algorithm already tells which wavelength is available at next hop unless there’s none available, This assumption is realistic because in RSVP-TE, there are objects that can suggest label or identify available label set that can be passed along PATH message, which greatly reduces the possibility of
label mismatch and cost of re-negotiation. In this simulator, it is pushes further that assume this will not happen; therefore PATHERR will not occur in the situation where re-negotiation of label is necessary.

- Path setup only. There is no real data packet following the completion of path setup in this simulation.

**Measurement**

The primary objective is to simulate and observe the performance of regular RSVP and modified version (bi-RSVP) that is capable of setup bi-directional path at once when used in setting up light paths during route path provision and restoration. We will simulate the performance of both RSVP and bi-RSVP when setting up bi-directional and unidirectional paths at provision and restoration time, while applying applications that are bi-directional or unidirectional in nature.

The measurements that we are interested are setup time, including path calculation time, switching time, preoperational delay, and protocol interactions. The second measurement is the number of messages exchanged during setup. Teardown is not exclusively included because in this simulation we adopted the use of Path State Removed flag, which essentially begins tear down process automatically once a PATHERR is received. If an explicit PATHTEAR message is used as in original RSVP, the performance should be comparable to setup time. Both will be tested under both unidirectional and bi-directional optical path RSVP. An analysis of restoration in scenarios including link failure is also part of measurement. The primary goal of measurement is to demonstrate the cost and benefit of using regular RSVP versus modified RSVP for optical bi-directional path setup.

**Remark**

This is a simplified simulation of actual path setup process, where there are a few points worth mentioning.

- Routing tables are given manually to setup the topology of the simulated network, where an individual OSPF link state table is given to each router. When a router receives a message (as class PACKET in the source code), it checks the destination address in the message, and then decides which hop to forward to. In real MPLS using RSVP, many networks are actually setup using source routing, because it allows network administrators to conduct traffic engineering route selections.
- There is no explicitly defined event called FAILURE. The way to simulate link failure is to manually alter the routing table at either side of the link, so the
neighboring node has infinite distance, and the route to any other destinations will not go through neighboring node as “next hop”. The FAILURE simulation also includes sending PATHERR messages from routers connected to failed link to routers that are still connected. This is one of the reasons it is the design choice have OSPF routing table at each router.

**Result and Discussion**

The results of shown as diagrams below, figure 1 shows the path setup time needed assuming the setup process goes smoothly, while figure 2 shows the time needed to detect path setup has failed in worst case. Figure 3 and 4 show number of messages exchanged and time needed for path restoration from failure happened until the whole path being restore, assuming the alternative route is calculated before hand. The setup assumes that there is only a linear chain of routers identified as nodes along the path, and each of them are 5 time units away from each other. Because they are equidistant, the measurement of time consumed is actually the as the message exchanged. The simulator is simple but can tailored to demonstrate the benefit of bi-directional path setup. The output of the simulator is a list of the events triggered after initial PATH request is inserted into event queue. In the failure and restoration cases, related PATHERR and router updates were inserted to queue and activated at preset time.
Figure 1. Path setup time

Figure 2. Time needed to detect failed path setup
Figure 3  Number of Messages Exchanged for Path Restoration

Figure 4  Time needed for Path Restoration

I would like to integrate some of the ideas from Internet draft MPLS-SIG with the observation I gained from my own simulation results as following:

12
1. The latency for setup a bi-directional path using RSVP is one round-trip time plus one initiator-terminator messaging delay, which is at least 3 times more than using bi-RSVP. Timing is the crucial criteria during restoration process.
2. The worst-case latency for discovering unsuccessful path setup is more than twice for original RSVP. The speed of detection contributes to another major element of fast restoration.
3. The saving of time during restoration process is not due to notification process of the failure but due to the time it takes to setup a new route and detect whether the route to be established has succeed.
4. The control message overhead is close to twice for original RSVP because of separate setup of forward and reverse links. This would be a major problem for contention and packet dropping if multiple paths pass through the same route, and all wants to restore the links.
5. Separate routing can result because the forward and reverse links are not established at the same time. Resource availability has a greater possibility to fluctuate between the time PATH message passed and the time RSEV came back from Egress. This will make the provision/restoration process endure a greater degree of uncertainty.

It was October when we decided to proceed with this enhancement scheme on bi-directional path setup and started to design possible protocol for message exchanges. Although there were confusion about how every state of the path can be passed or how to gain agreement about which wavelength to use, as well as concerns about constraints such as availability of wavelength converters in current optical network. In mid-November it is found that Ashwood, et al, published two Internal drafts on RSVP-TE (which we called bi-RSVP in our previous reports) and GMPLS-SIG in October, which already incorporates the idea of bi-directional path setup as well as notifying available wavelength, etc into this draft. The careful examination shows that the fundamental idea is extremely similar, but their design is more firmly built on original MPLS and takes greater care about various details.

Although this may appear that publication of these documents reduce the originality of this work. It is actually a sign that our approach was in the right track, and many experts in the fields feel the same way as we do. It is well accepted as reinforcement to our idea by me.

**Future Work**

Many researchers have already observed the benefit of bi-directional path setup process, and it is being verified here in this project. With the publishing of Internet drafts on
RSVP-TE and MPLS-SIG, our intuition of attacking fast restoration problem from the bi-directional path setup has been verified by the industry. The algorithm for computing alternative restoration path and timing for conducting entire tasks of switch over would have a great impact on the performance of the restoration. Further enhancement will definitely have to follow these routes.

The simulator implemented for this project, although still relatively primitive compared to other tools evaluated, but it does serve better than any other software for this particular case of study. Obviously with further enhancement, such as automating setup of topology, implementing a more complete set of functionalities to handle PATHERR and RSEVERR that were not necessary for the objective of this project, and creating comprehensive FAILURE event or RECOVER event would be of great benefits. This enhanced version of simulator should also be able to contribute as a good simulation environment of enhancements mentioned in previous paragraph.

Conclusion

Fast restoration is one of the fundamental requirements for optical networks. However, in the process of bring IP to optical networks, people are more focused on provisioning of initial path, but ignore some of the fundamental differences between traditional network and optical ones. Several strategies addressing these issues in restoration process has been identified, and a solution has been proposed for a specific problem as bi-directional link for optical path. The bi-directional RSVP messaging shows great benefit in path provision and restoration, and the simulation verifies this claim.

Acknowledgement

I would like to acknowledge the value of discussion and help from two of my colleagues, Haifeng Zhu from ECE department and Karthik Rajan from INI. Haifeng offers his expertise in building network simulation, and he generously offers one of his earlier versions as template when building software tool. He also brought up some stimulating idea during many of our group discussions. Karthik Rajan joined us for early half of the meetings, and contributed to some of the discussions. I am also grateful to Dr. Tonguz. He gave us so much insight into the problem in office hours, and gave crisp comment on previous progress report of this work.

References