Epidemic Network Failures in Optical Transport Networks

SARAH RUEPP, DIMITRIOS KATSIKAS, ANNA M. FAGERTUN
DTU Fotonik
Technical University of Denmark
Orsteds plads, building 343, 2800 Kgs. Lyngby
DENMARK
srru@fotonik.dtu.dk

Abstract: - This paper presents a failure propagation model for transport networks which are affected by epidemic failures. The network is controlled using the GMPLS protocol suite. The Susceptible Infected Disabled (SID) epidemic model is investigated and new signaling functionality of GMPLS to support epidemic failure resolution is proposed. The results provide important input to service recovery mechanisms under epidemic failures.

Key-Words: resilience, transport networks, epidemic failures, network modelling, GMPLS, performance evaluation

1 Introduction

Nowadays, transport networks carry extremely large amounts of network traffic, and are widely spread across multiple geographical locations. As a result, any possible connectivity failure could directly impact the service delivery of a vast amount of users. Therefore, the network should be able to recover fast from a failure in order to provide service continuity to the user. Several recovery techniques have been employed by the Internet Service Providers (ISPs) such as adding redundancy to network equipment (e.g. routers, optical cross-connects, etc.), or by provisioning alternate paths (path protection, path restoration) [1] and recovery mechanisms have amongst others been treated in the following works [2]-[6]. Hence, assuming sufficient resources, network resilience can be achieved when a single failure occur (e.g. fiber cut). However, when it comes to simultaneous failures such as cascading and epidemic failures, the available solutions are expensive [7]. For Generalized Multi Protocol Label Switching (GMPLS) transport networks, network survivability under multiple failures has been discussed in [8]-[10]. Virus propagation models from the field of epidemiology have been altered for simulating network failure scenarios and the failure propagation probability within the network [4],[5],[11],[12].

This paper evaluates the reliability of a GMPLS transport network under epidemic failure scenarios. Thus, the aim is to increase the fault tolerance of the GMPLS technology when simultaneous failures occur, impacting a large number of network nodes across an optical transport network (OTN) in order to ensure the service delivery.

The remainder of the paper is organized as follows: Section 2 describes the GMPLS framework. Section 3 deals with epidemic failures. Section 4 presents details about the SID implementation. The simulation study and its results are presented in section 5. Section 6 concludes the paper.

2 GMPLS Architecture

GMPLS is an enhanced version of the MultiProtocol Label Switching (MPLS) architecture. MPLS uses labelled packets instead of using IP addressing for its forwarding decisions. In this way high switching performance is achieved, and at the same time requirements for traffic engineering are satisfied. The path from source to destination is called Label Switched Path (LSP). The network nodes, which support labelled paths, are called Label Switch Routers (LSR). MPLS LSRs have been designed to support only packet switching. GMPLS is extending the concept of label switching in order to enable it to work with optical networks [13]. Thus, switching technologies such as Time Division Multiplex (TDM), Lambda Switch Capable (LSC) and Fibre Switch Capable (FSC) are supported by GMPLS. The support of those additional switching types in the optical domain has driven the extension of the GMPLS control plane, which is now logically and/or physically separated from the data plane. TDM, LSC and FSC introduce new constraints to IP addressing and to the routing models due to the fact that several hundreds of parallel physical links (e.g. wavelengths) are possible to exist between two interconnected nodes [14]]. This separation of the control plane and the data plane introduces extra constraints, as additional control plane signalling

techniques are required for managing the data plane failures. On the other hand failures on the control plane are not necessarily a result of data traffic connection failures. GMPLS details are discussed in the following sub-sections.

2.1 GMPLS Routing

GMPLS networks typically use extended versions of the Open Shortest Path First-Traffic Engineering (OSPF-TE) algorithm for their routing decisions. Usually rerouting is required when a failure occurs along the already established LSP. Under certain conditions it might also be necessary for a LSP to return back to its original tunnels, if the failed resource becomes re/activated (reversion) [18].

2.2 GMPLS Signalling

In order to set up and tear down LSPs, GMPLS is making use of the Resource Reservation Protocol (RSVP) extensions. RSVP was initially designed to support Integrated Services (IntServ) in IP networks for reserving resources on the router in order to satisfy receiver initiated requests for Quality of Service (QoS). Therefore, when a sender wants to set up a connection, it is advertising its status by transmitting a Path message. This Path message traverses the network on a hop by hop basis in the downstream direction to one or more receivers as shown in Figure 1. The Path messages traverse the network towards the destination via intermediate RSVP-capable routers. Once a path message reaches its destination, the recipient node sends a Reservation (Resv) message. While the Resv message traverses the reversed path in the upstream direction to the sender, it is causing each intermediate node to reserve the traffic characteristics advertised in the Resv message.

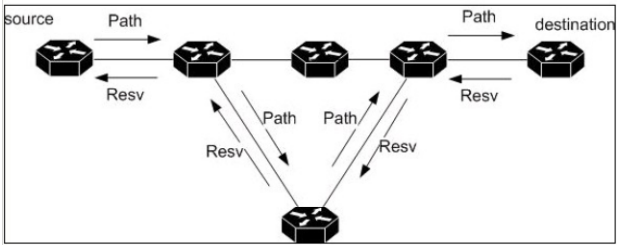

Figure 1: RSVP operation

The development of MPLS required that RSVP should be extended to allow the support for Traffic Engineering (TE) by requesting and distributing label bindings [15]. This resulted to a modified version of RSVP, known as Resource Reservation Protocol-Traffic Engineering (RSVP-TE). RSVP-TE messages must include the following extra information:

- A Label Request object: It is included in the Path message and informs the downstream LSR that it requests a label. In the path message an Explicitly Route Object (ERO) can be included. The ERO objects consist of the route of the nodes until the final destination.
- A Session Attribute object: Indicates the priority of the requested LSP. The downstream node will compare this attribute with the holding priorities of the already established LSPs in order to decide if a new LSP should be established. The session attribute is included in Path messages.
- A Label object: It is included in Resv messages and informs the upstream LSR which label should be used as unique identifier for the forwarding decisions.

The LSP tunnel is established in the same fashion as previously described and data can flow via this path. In order to avoid adding extra load to an already congested path, each node in the LSP tunnel is using the above information also in refresh messages; even if there has been no change in the tunnel's state. In case an intermediate node does not support Label requests or has no resources available it sends a Path Error (PathErr) message back to source node. By the end of the data transmission, if the receiver, or the sender has no more data to send, they can delete the created state by respectively using a message for releasing the allocated resources (ResvTear) and a message for tearing down the path (PathTear). Support for Hello messages has been defined in RSVP-TE extensions for node failure detection between neighbour nodes [15].

2.3 Link Management

The Link Management Protocol (LMP) is a point to point protocol which was defined in [12] It provides a mechanism for creating and managing multiple control channels between adjacent GMPLS nodes. It supports neighbour discovery fault management, thus takes part in the protection and restoration mechanisms of GMPLS optical networks. Some of the most important functions are:

- Control Channel Management: LMP neighbour nodes exchange messages for establishing a control channel connection. Once the control channel is active, maintenance of the channel is achieved by the regular exchange of Hello messages.

- Link Discovery and Verification: Is a procedure for checking the status and connectivity of the data links between two LMP peers. This may be carried out on a timer just to check that everything is functioning correctly, or it may require specific operator intervention due to a possible failure.
- Link Capabilities Exchange: It is an optional phase, which can take place after Link Discovery. So that the LSRs can tell each other about the specific features of the data link that is useful to build TE links out of multiple parallel physical links through the process bundling.
- Fault Isolation: One of the most important features of LMP. Devices such as photonic crossconnects may normally not notice, if there is a disruption to the signal, and LMP helps to isolate and report faults, that may occur. The process is initiated by a downstream node that detects a problem on a data link. A failure can be detected due to Loss of Light (LoL) or due signal degradation.

3. GMPLS Survivability under Epidemic Failures

Network survivability is defined as the set of capabilities that allow a network to recover from failures in a timely manner [19],[20] In GMPLS transport networks the failures can be split into two groups:

1. Control plane failures: for example a controller misconfiguration or a channel failure could result in making new service delivery requests impossible and existing services unmanageable.
2. Data plane failures: directly impact the service delivery and could be caused by a failure of an element across the transmission line i.e. a fiber cut or a power outage.

3.1 General Failure Mechanisms

In general terms failures could occur as a consequence of software or hardware defects, power outages or natural disasters such as earthquakes, floats, etc. Since the early start of the transport networks, service recovery processes have been defined under the term fault management as a key factor for improving the service availability and reliability. In GMPLS, fault management is taking place in the following 3 steps:

1. Fault detection
2. Fault localization and isolation
3. Fault notification and recovery

Depending of the recovery type defined by a Service Level Agreement (SLA) with the service provider, there are certain actions to be performed in order to switch over the traffic to alternate paths for recovering the service. The time it takes for switching the traffic to a working path is the recovery time T, which is calculated as follows:

$$T = T_f + T_l + T_r \quad (1)$$

where

T_f is the fault detection time,
T_l is the fault isolation time,
T_r is the fault recovery time.

In case there are not any service recovery guarantees (unprotected service), then no actions are performed.

In GMPLS networks service recovery can be achieved by the so called protection and restoration mechanisms. The former defines a service recovery class where support for one or more alternate routes is required. An alternate route assumes that at least one redundant path has been provisioned and resources have been allocated pro-actively; before a failure is detected. The restoration mechanism is taking place after a failure occurrence, when for the recovery of the service a new path needs to be calculated, or has already been calculated. Thus, after a failure notification is received it is decided, if resources should be dynamically allocated for serving this new path.

3.2 Epidemic Failures

Epidemic failure propagation has its roots in medical virology and relates to models on how diseases are spread [16]. The Susceptible Infected Disabled (SID) model was proposed as an extension to the SIS model in order to model the behaviour of an epidemic in GMPLS transport networks [22] . The SID model was proposed for dealing with failures, which tend to propagate over the network.

The states listed below represent the possible GMPLS node states according to the SID model:

1. Susceptible (S): State where both the control plane and the data plane are operational.
2. Infected (I): State where the control plane fails, but the already established LSPs continue to function, i.e., data forwarding is not impaired. After a given period the node

either recovers (going to S state) or completely fails (going to D state).
3. Disabled (D): Both control plane and data plain fail representing a complete nodal failure. Thus any provided service stops.

A susceptible node can be infected with probability β. When a node is at the infected state the restoration process starts and lasts a given amount of time, which is proportional to the Mean Time To Repair (MTTR). After this time has expired, the node becomes susceptible with probability δ_1 or disabled with probability τ. In case the node becomes disabled another restoration process will take place, which has a success probability of δ_2. If the restoration process is successful the node will transit to the susceptible state; otherwise, in case of a failure the node will remain disabled [22] . The possible transitions according to the relevant probabilities are depicted on **Figure 2**.

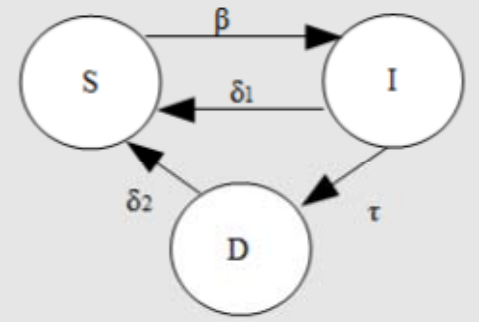

Figure 2: SID Model [17]

The average number of infections produced by an infected node, called basic reproduction number R0, is calculated using the following formula [10]:

$$R_0 = \frac{\beta}{\delta_1 + \tau} \cdot \lambda_1 \quad (2)$$

where $\lambda_1 > 0$ is average nodal degree when a homogeneous network is considered. In case the network is not homogeneous, it has been proven that the largest eigenvalue of the adjacency topology matrix (spectral radius) is a more suitable property for epidemic modelling [21] If R0 < 1, then the infection dies out over time. On the contrary, if R0 > 1 the epidemic sustains while impacting a large number of nodes within the network. In this case the proportion of susceptible (S) nodes is

$$S = \frac{1}{R_0} \quad (3)$$

and the proportion of infected (I) and disabled (D) nodes is given by equations (4) and (5) respectively:

$$I = (1 - S)\frac{1}{1 + R_1} \quad (4)$$

where $R_1 = \frac{\tau}{\delta_2}$ and

$$D = (1 - S)\frac{R_1}{1 + R_1} \quad (5)$$

3.3 Application to GMPLS Control and Data Plane Failures

Failure detection is a vital part of the service recovery. Once a failure is detected it needs to be reported to the relevant nodes that formulate the LSP. In the current work, two control plane failure detection methods are examined:

1. By using a Path timer message for refreshing the LSP state.
2. With the use of a Hello protocol.

When it comes to data plane failures, those can be detected almost instantly by the LoL (Loss of Light on the line) or by monitoring the Bit Error Rate.
In the following subsections the implemented signaling operations are described for the control and data plane separately.

3.2.1 Control plane failure during the downstream LSP establishment

During the operation to the downstream direction there are two possible failure scenarios that need to be handled depending on which mechanism detects the failure and on which plane is impacted. Two mechanisms have been implemented for detecting a control plane failure:

Via a Path timer: By using a timer after a Path message is sent. As shown on Figure 3 the downstream node B has received a Path message from his upstream node A and has available resources to allocate. Thus, it forwards a Path message at time T1 to its downstream node C and starts a timer. This timer is defining the period in which a node waits to receive a Resv message. When the node receives a Resv message it can allocate resources for serving the connection. This period is equal to the round trip time (RTT) starting from the moment that a Path message is send until a Rerv message is received plus the processing delay of each node along the path [19]. Consequently, if node B has not received a Resv within a certain time

period it blocks the connection request and sends a Path error message to its upstream node A which in return terminates the connection request, as it is the initiator of the LSP request.

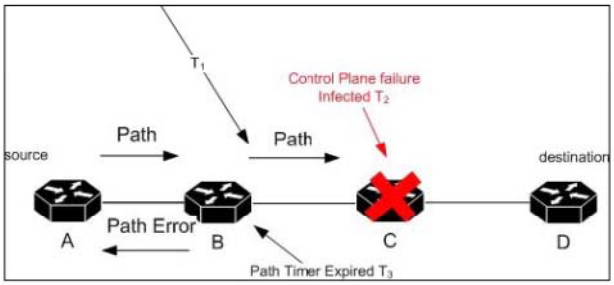

Figure 3: Control plane failure detection via the Path timer at the downstream direction

Via the Hello protocol:

As mentioned before, a node does a look up on its routing table and sends Hello messages to its directly connected neighbors in order to check if they are available. On Figure 4, node B sends a Hello packet to the infected node C at T1 and starts the retransmission timer.

When the timer expires (after 30 sec) it sends another Hello message. If node B does not receive a Hello Ack it will retransmit again the Hello. After 3 unsuccessful attempts node B will realize that node C is not responding and will report to its upstream node with a Path Err message (for every ongoing LSP establishment process). Additionally, Node B updates its routing table by excluding node C as downstream destination. Thus, incoming Path requests that include node C in the ERO message will be blocked.

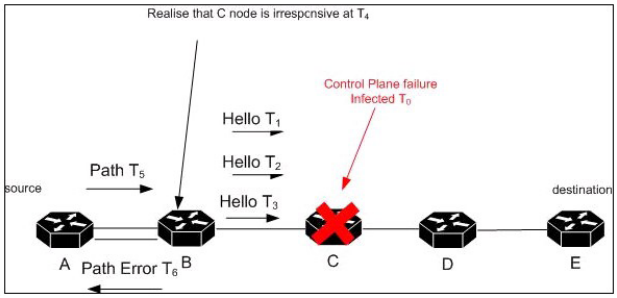

Figure 4: Control plane failure detection via the Hello protocol

3.2.2 Control plane failure during the upstream LSP establishment

In this case a node becomes infected during the LSP establishment in the upstream direction as depicted on Figure 5. Node D has allocated resources for serving the connection request for both directions and sends a Resv message to its upstream node C at time T4. Node C will fail to receive the Resv message because its control plane is not operating. Hence, node

D will realize this failure at a certain time T5 via the Hello protocol, after 3 unsuccessful attempts for receiving Hello Ack. Until the time when this failure is realized nodes D and E have allocated resources for serving the LSP. Therefore, Node D will release its resources for serving this path and send a Path tear message to the downlink direction. A Path tear messag will be forwarded from the node that detected the failure to the downstream direction.

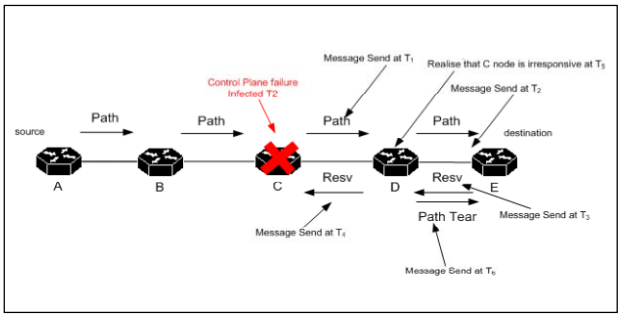

Figure 5: Failure occurrence during upstream LSP establishment

3.2.3 Data plane failures

As previously mentioned the data plane failures can be detected almost instantly by the LoL or by monitoring the Bit Error Rat (BER). In a real network system, this would be detected by the physical equipment. In the model however, this failure detection is mimicked by letting the failed node sends a Failed message to its neighbors.

Therefore the neighbor nodes upon receiving the Failed message are informed about the failure and examine if there is an active LSP that uses the failed node. If there is an active LSP traversing this node then it should be identified whether it is located at downstream or upstream direction.

According to where the failure is located along the path the relevant signaling operations are taking place informing the nodes along the path regarding the broken path.

Data plane failures can happen at any point. Thus, the cases where a data-plane failure of a node on the path happens have to be covered. The following cases are examined:

1. During the path establishment to the downstream direction. If the failed node is downstream node direction a Path err message the detecting node is reporting to upstream direction among the path Figure 8 A. The failed node C is a downstream node to B. In the case B is sending Path Error to the path

initiator node A. On the other side of the path node D detects that there is a failure of its upstream node C. Hence the failure is reported by sending a Notify message the next hop in the downstream direction. In this way all nodes along the path will be notified for the broken path.

2. During the path establishment to the upstream direction. In this scenario Figure 8 B the failure occurs while resources have been allocated for servicing the path. If the failed node is located at the downstream direction then a Notify message is send to the upstream direction until it reaches the initiator node that will drop the connection. In the downstream direction of the broken LSP all the nodes have reserved resources for servicing the path. Therefore the reporting node D reporting the failure by sending a Path tear message to downstream direction.

3. While the LSP is established. This case is handled in the same way to the as in case 2.

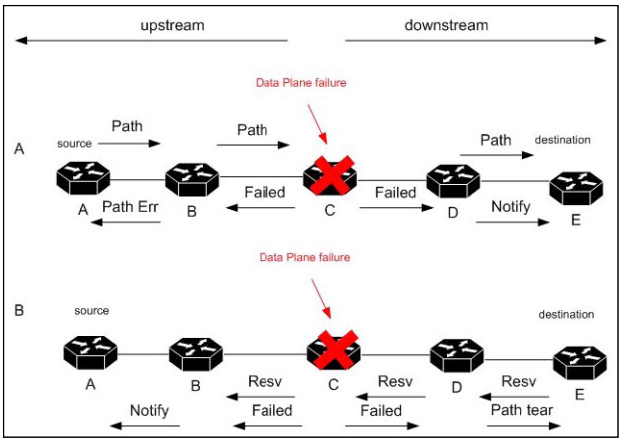

Figure 6: Data plane failure during LSP establishment

3. SID Algorithmic Implementation

For initializing the SID model a node is selected randomly from the network topology map and is set as infected. This node is the starting point for spreading the infection to its neighbor nodes.
Thus, in case the selected node has a high nodal degree it is expected to see wider spread of the infection, as seen in [10].
Whenever a node is entering the infection state the SID algorithm is executed in order to determine the time period a node will remain infected until it transits to the next state. The SID algorithm has been implemented in a procedure called *meaninfectionperiod()* as follows:

```
random                                  //random variable
TimePeriod                              //The time frame in which the node will remain infected.
                                        This value is correlated with the expected MTTR.
delta1                                  //the recovery probability from the infected state to the
                                        susceptible state.
ProbabilityT                            //The probability a node becomes disabled.

If (random<delta1)                      // Next state is susceptible.
Set the infection recovery timer        //the node remains infected for an exponentially distributed
                                        period of (1/random)* TimePeriod
Set infection rate timer                //The transmission rate is exponentially distributed for the
                                        period of time a node remains infected
                                        (1/delta1 + probabilityT))* TimePeriod

Else if (random<delta1+probabilityT)    //Next state is disabled.
Set infection recovery timer            //the node stays infected for an exponentially distributed
                                        period of (1/random)* TimePeriod

Set infection rate timer                //the transmission rate is exponentially distributed for the
                                        period of time a node remains infected
                                        (1/delta1 + propabilityT))* TimePeriod

Else
                                        //Then (random > delta1+probabilityT) the node will return
                                        back to the infected state
Set infection recovery timer            //the node remains infected for an exponentially distributed
                                        period of (1/random)* TimePeriod
Set infection rate timer                //the transmission rate is exponentially distributed for the
                                        period of time a node remains infected
                                        (1/delta1 + probabilityT))* TimePeriod
```

Figure 7: Algorithmic implementation

During the period of time a node is infected and the control plane is failed, the node is stopping the transmission of any signalling messages. Consequently, when a node is entering the Infected state it performs the following actions:

- Stop responding to any signalling messages by dropping all incoming messages.
- Stop generating new connection requests and also stop any control plane signaling.
- Keep the already established connections active.
- Transmit the infection to its neighbours.

As previously described during the description of the *meaninfectionperiod()* procedure, when the infection recovery timer expires the node could possibility:

- Return back to the Susceptible state with a certain probability $\delta 1$ or,
- Transit to the Disabled state with τ probability.

In case none of the above actions occur, then the node remains infected until the infection recovery timer expires. Then, *meaninfectionperiod()* procedure will be executed again creating a loop as long as the result is to remain infected. Hence, there is no guarantee that a node ever exits the infected state.
For the case the procedure results that the next state will be Susceptible the control plane returns back to operation. Additionally, new connection requests can be initiated by re-enabling the ReqGen.
Finally, in the case that the procedure results that the node will transit to the Disabled state the disabled recovery timer is activated. In real life this timer

corresponds to the time period when troubleshooting actions are taking place for repairing a node failure. During the reparation period the Disabled node is irresponsible and both control and data plane are failing. Thus, when the node is entering the Disabled state implements the following actions:

- Release the resources which have been allocated for serving any active LSPs' (connections).
- Stop transmitting the infection.
- Drop all incoming signalling messages.
- Stop generating new connection requests and start signalling.

When the disabled recovery timer expires it should be decided whether the node will either become Susceptible again (probability δ2), or if it will remain at its current Disabled state. In case the node remains Disabled the actions below are taking place:

- Restart the disabled recovery timer.
- Reproduce a random time and check whether it is greater than the value of the δ2 probability.

It is worth noticing that in both cases when a node is Infected or Disabled there is no guarantee that the node will return to Susceptible state after the recovery timer expires. This is one of the main differentiators of the SID model in comparison to previous network epidemic models.

Finally, the time period that a node remains as Infected Ti and Disabled Td is given by the following formulas:

$$T_i = \delta_1 MTTR_i \quad (6)$$

$$T_d = \delta_2 MTTR_d \quad (7)$$

Where MTTRi is the Mean Time To Repair a control plane failure and MTTRd is the Mean Time To Repair a data plane failure.

In general terms the MTTR is defined as the mean period of time needed to repair a failure or to recover a service. The MTTR can vary according to the offered SLAs' by the service providers.

4 Simulation Study and Results

The SID epidemic propagation model is evaluated using the OPNET Modeler [23] simulation software, in both a homogeneous and a heterogeneous network topology.

It is worth noticing that in both cases when a node is Infected or Disabled there is no guarantee that the node will return to Susceptible state after the recovery timer expires. This is one of the main differentiators of the SID model in comparison to previous network epidemic models.

4.1 Homogeneous network topology

As a first step, the model is verified against analytical results for a homogeneous network networks given in [22]. The selected homogeneous network topology is shown in Figure 8.

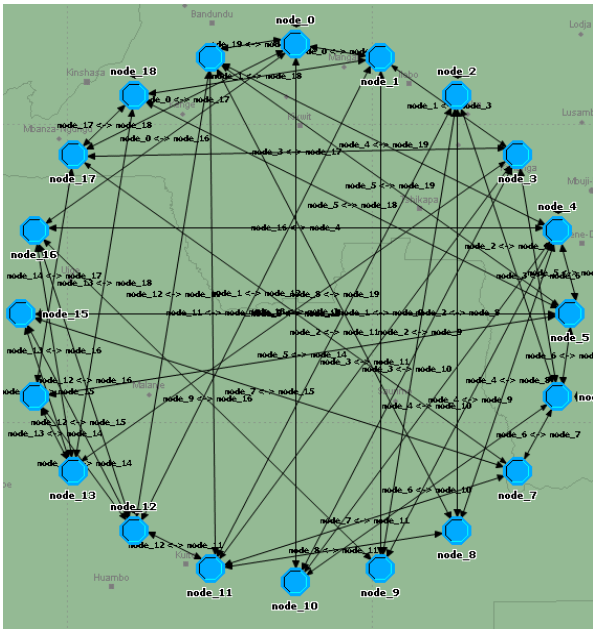

Figure 8: Homogeneous random mesh topology

The network type is random mesh and consists of 20 nodes. In order to verify the model the epidemic should persist, i.e. the basic reproduction number R0 must be greater than 1. Hence, the epidemic is spreading over time impacting a large amount of nodes.

Due to the fact that the average nodal degree (λ) is highly impacting the infection propagation it has been chosen as simulation parameter for comparing the simulation results against the analytical values. Therefore, the infection and the recovery probabilities have been kept as constant parameters with the following values: $\beta = 0.169$, $\tau = 0.1$, $\delta_1 = 0.3$, $\delta_2 = 0.3$. The values of the average node degree are the result of adding more links between the network nodes. The expected fraction of Susceptible, Infected and Disabled nodes has been calculated by using the formulas (2)-(5). The simulated results have been derived by 140 simulation experiments simulating a 2 week period. Both infection recovery and the disable recovery

period have been set t 2 minutes. Those values are intentionally kept low due to the fact that longer recovery times will result in a pandemic when a homogeneous network is considered. The results correspond to the percentage of nodes over time for each state.

Both analytical and simulation results are displayed on. As can be seen on Figure 9, adding more links to the network increases the probability that an infected node will successfully transmit the infection to one of its neighbours. As a consequence the number of susceptible nodes declines while the value of λ increases. The analytical results have been calculated using the formulas for homogeneous networks given in [22].

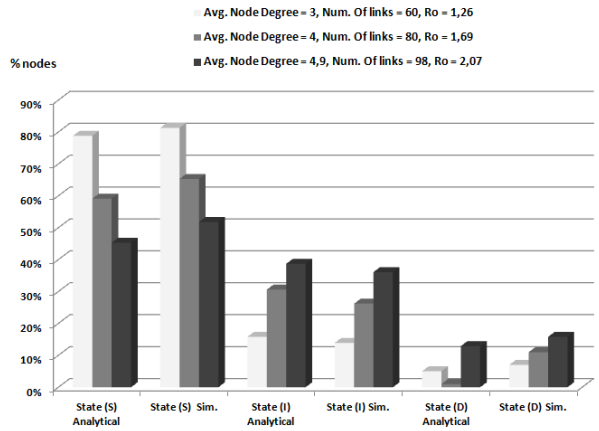

Figure 9 : Comparison of analytical values and simulation results

Next, the performance of the two notification methods, the Hello Protocol method and the Path timer method, is evaluated and the results for average link usage, dropped new connection requests and the number of susceptible nodes are shown in Figure 10 and Figure 11, respectively.

The simulation period is 30 days. The epidemic is initialised on day 15. The number of dropped new connections linearly increases for both implementations. As can be seen the Hello protocol method results in fewer dropped new connection requests (please notice the difference in the y-axis).

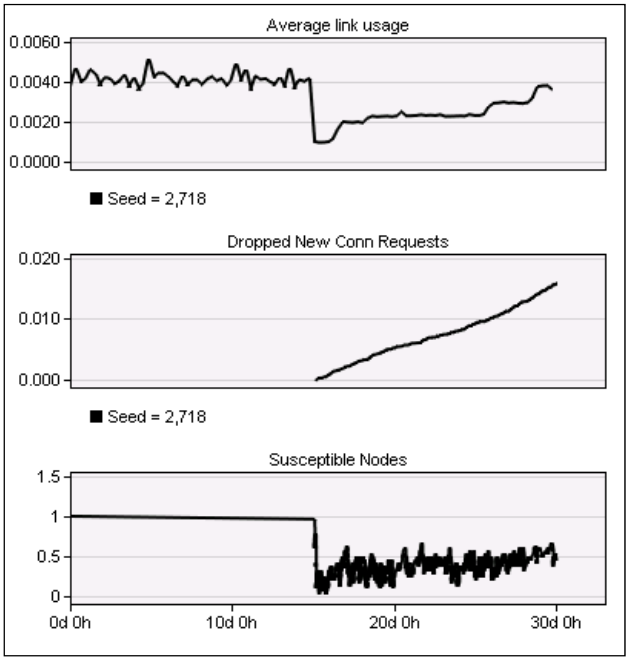

Figure 10: Evaluation of Hello protocol method

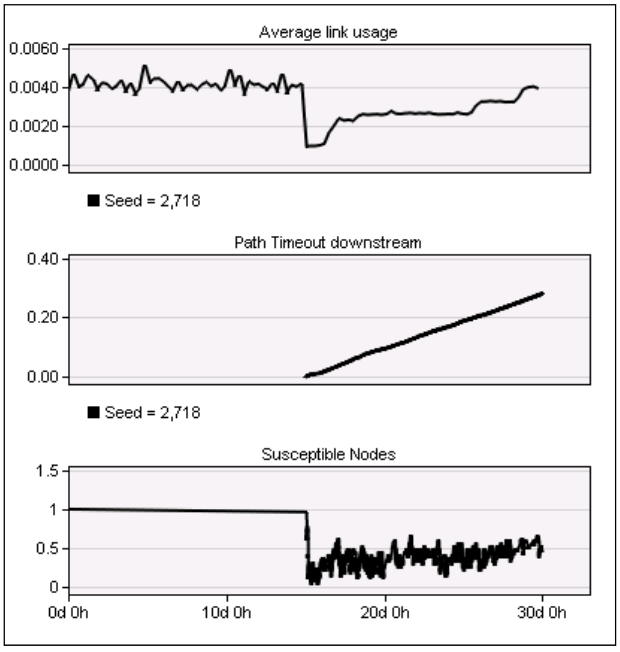

Figure 11: Evaluation of Path timer method

4.2 Heterogeneous network topology

The evaluated heterogeneous network topology is the Pan-European optical network shown in **Figure 12**.

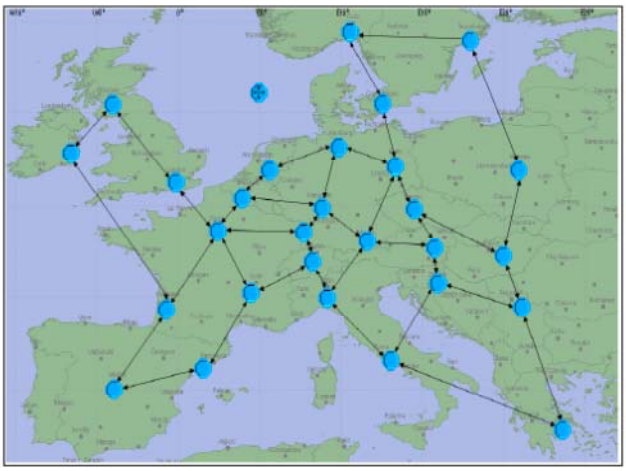

Figure 12: Pan European Network Topology

The network consists of 28 nodes and 78 links interconnecting major cities located in Europe with average nodal degree $\lambda = 2.7$. The eigenvalues of the adjacency matrix have been calculated and the largest value is equal to 3.232. The time periods of the recovery timers are related to a complete node failure, where it might take one full working day (8 hours) for repair. The MTTRi and MTTRd correspond to the different recovery times for the infection and the disabled state respectively. The value of R0 is adjusted by increasing the value of the infection probability β. Thus, starting from $R0 < 1$ (the epidemic dies over time) the basic reproduction number increments by increasing the infection probability as shown in Table 1.

	Analytical Values									
β	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0**
R0	0.8	1.6	2.4	3.2	4.0	4.8	5.7	6.5	7.3	8.1
S	100%	62%	41%	31%	25%	21%	18%	15%	14%	12%
I	0%	29%	44%	52%	56%	60%	62%	63%	65%	66%
D	0%	10%	15%	17%	19%	20%	21%	21%	22%	22%

Table 1: Analytical values as function of beta β

The percentage of the nodes for each state is presented as function of the basic reproduction number (R0). The recovery probabilities have been kept as constant parameters with the following values: $\tau = 0.1$, $\delta_1 = 0.3$, $\delta_2 = 0.3$. The results are presented against the analytical values for each state with a 95% confidence interval over 100 simulation experiments simulating one month timer period. The selected MTTRi and MTTRd are 2 and 8 hour respectively. Figures 12-14 illustrate the percentage of nodes in states S, I and D respectively, compared against their analytical values. Please not that the same formulas for the analytical values were used, but as the work in [22] considers homogeneous networks some deviation is expected between the simulated and the analytical values.

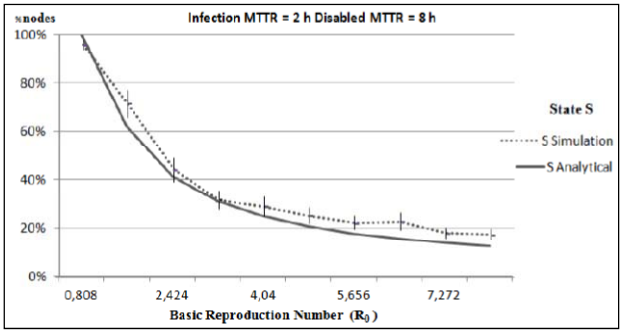

Figure 13: Percentage of nodes in S state as a function of R0

In Figure 13 it is observed that the simulation results for S state present a minor deviation compared to the analytical values however they follow the analytical curve.

By looking at Figure 14 and Figure 15 where state I and D are shown, the simulation results considerably deviate from the analytical ones. The deviation becomes wider for higher values of R0. The reason for this deviation is related to the characteristic of the network. The average nodal degree, the connectivity density, the network diameter and size etc. affect the simulation results, including the heterogeneity of the network topology.

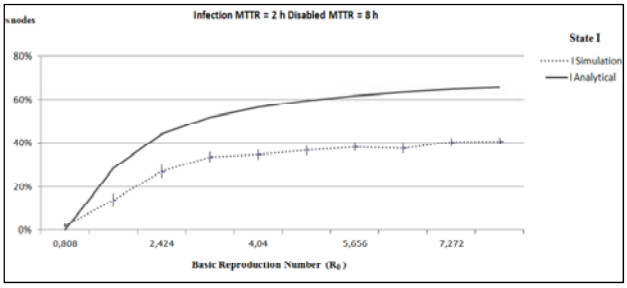

Figure 14: Percentage of nodes in I state as function of R0

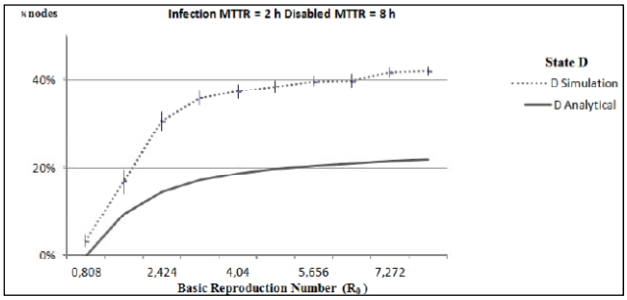

Figure 15: Percentage of nodes in D state as function of R0

A node that becomes disabled could possibly remain in Disabled state during the simulation period. Another important factor is the fraction of time where a node remains infected.

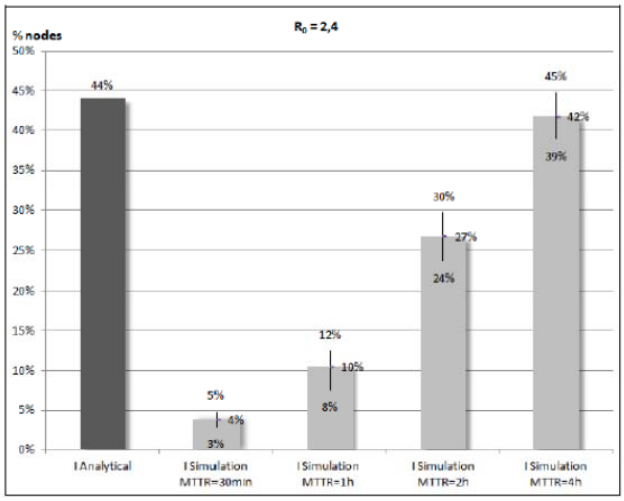

Figure 16: Analytical values against the simulation result for the I state for different MTTRi

Figure 16 illustrates that the chosen MTTRi has a significant impact on the average percentage of infected nodes in the simulation. By using only the MTTRi as a single simulation parameter the experiment took place for 4 different MTTRi while the MTTRd was kept to 8 hours. At an MTTRi of 4 hours the average percentage of infected nodes resulted in 42% with an 95%-confidence interval between [39;45]. Thus the analytical value of 44% lies in the confidence interval given an MTTRi of 4 hours.

5 Conclusion

In this paper, the dynamics of epidemic failure spreading in a Pan-European heterogeneous network is analyzed. The GMPLS framework is extended to accommodate epidemic failure messages and model the SID epidemic model in OPNET. The results show that the dynamic simulation follows the analytical values for the S and I states, whereas some deviation in the D state due to the topological characteristics of the network topology is observed.

References:

[1] Grover, W.; Doucette, J.; Clouqueur, M.; Leung, D.; Stamatelakis, D.: "New Options and Insights for Survivable Transport Networks," IEEE Communications Magazine, Vol 40, No1, pp. 34-41, January 2002.

[2] Ruepp, S.; Dittmann, L.; Berger, M.; Stidsen, T." Evaluating the Efficiency of Shortcut Span Protection", in WSEAS Transactions on Communications, Issue 2, Volume 9, February 2010 Pages:143-152 ISSN:1109-2742

[3] Ruepp, S.; Wessing, H.; Zhang, J.; Manolova, A.; Rasmussen, A.; Dittmann, L.; Berger, M. "Evaluating Multicast Resilience in Carrier Ethernet", in WSEAS Transactions on Circuits and Systems, Issue 2, Volume 9, February 2010 ISSN: 1109-2734

[4] Ruepp, S.; Fagertun, A. M. "Epidemic Propagation of Control Plane Failures in GMPLS Controlled Optical Transport Networks" in proc. of Design of Reliable Communication Networks (DRCN) conference, Budapest, March 2013

[5] Katsikas, D.; Fagertun, A. M.; Ruepp, S. "Survivability Strategies for Epidemic Failures in Heterogeneous Networks", In proc. of 12th WSEAS International Conference on Electronics, Hardware, Wireless and Optical Communications, Cambridge, UK, February 2013

[6] Ruepp, S.; Dittmann, L.; Berger, M.; Stidsen, T. "Capacity Efficiency of Recovery Request Bundling", in proc. of 4th WSEAS International Conference on CIRCUITS, SYSTEMS, SIGNAL and TELECOMMUNICATIONS (CISST '10) Harvard, MA, USA, January 2010

[7] Horie, T.; Hasegawa, G.; Kamei, S.; Murata, M.: "A new method of proactive recovery mechanism for large-scale network failures," 2009 AINA Conference, pp. 951-958, May 2009.

[8] Segovia, J.; Vilà, P.; Calle, E.; Marzo, J. L.: "Improving the Resilience of Transport Networks to Large-scale Failures," Journal of Networks, Vol 7, No 1, pp. 63-72, January 2012.

[9] Yamanaka, N.; Shiomoto, K.; Oki, E.: "GMPLS Technologies Broadband Backbone Networks and Systems," Taylor & F, 2005.

[10] Manzano, M.; Segovia, J.; Calle, E.; Vilà, P.; Marzo, J. L.: "Modelling spreading of failures in GMPLS-based networks," 2010 Intl SPECTS Conference, pp. 244-249, July 2010.

[11] Chakrabarti, D.; Wang, Y.; Wang, C.; Leskovec, J.; Faloutsos, C.: "Epidemic Thresholds in Real Networks," ACM Trans. on Infor. and System Security, Vol 10, No 4, pp. 1-26, Jan. 2008.

[12] Pastor-Satorras, R.; Vespignani, A.: "Epidemic dynamics and endemic states in complex networks," Physical Review E, Vol 63, No 6, p. 066117, May 2001.

[13] Griffith, D.:"The GMPLS Control Plane Architecture for Optical Networks," Emerging Optical Network Technologies: Architectures, Protocols and Performance, Edited by K. M. Sivalingam & S. Subramaniam, Springer Inc., pp. 193-218, 2005.

[14] Mannie, E.: "Generalized Multi-Protocol Label Switching (GMPLS) Architecture," RFC 3945, October 2004.

[15] Berger, L.: "Generalized Multi-Protocol Label Switching (GMPLS) Signaling," RFC 3471, January 2003.

[16] De, P.; Das, S. K.: "Epidemic Models, Algorithms, and Protocols in Wireless Sensor and Ad Hoc Networks," Algorithms and Protocols for Wireless Sensor Networks, Wiley., pp. 51-76, 2009.

[17] Fedyk, D.; Aboul-Magd, O.; Brungard, D.; Lang, J.; Papadimitriou, D.: "A Transport Network View of the Link Management Protocol (LMP)," RFC 4394, February 2005.

[18] Awduche, D.; Berger, L.; Gan, D.; Li, T.; Srinivasan, V.; Swallow, G.: "RSVP-TE Extensions to RSVP for LSP Tunnels," RFC 3209, December 2001.

[19] Papadimitriou, D.; Mannie, E.: "Analysis of Generalized Multi-Protocol Label Switching (GMPLS)-based Recovery Mechanisms (including Protection and Restoration)," RFC 4428, March 2008.

[20] Banerjee, A.; Drake, L.; Lang, L.; Turner, B.; Awduche, D.; Berger, L.; Kompella, K.; Rekhter, Y.: " GMPLS: An Overview of Signaling Enhancements and Recovery Techniques," IEEE Commun Magazine, Vol 39, No 7, pp. 144-151, July 2001.

[21] Lewis, T.G.: "Network Science: Thoery and Applications," Wiley & Sons, Inc., 2009.

[22] Calle, E.; Ripoll, J.; Segovia, J.; Vila, P.; Manzano, M.: "A Multiple Failure Propagation Model in GMPLS-Based Networks," IEEE Network, Vol 24, No 6, Nov-Dec 2010.

[23] OPNET Modeler, www.opnet.com