Proceedings of the 5th WSEAS Int. Conf. on APPLIED INFORMATICS and COMMUNICATIONS, Malta, September 15-17, 2005 (pp91-97)

Modelling direct application to network bandwidth provisioning for high demanding research applications

H. Wessing, Y. Yan and M. Berger
Research Center COM
Technical University of Denmark
Bldg. 345V, Oersteds Plads, 2800 Kgs. Lyngby
DENMARK
http://www.com.dtu.dk/staff/hw

Abstract: In this paper, we report on the modelling activities and the first results in the Eurporean IST project MUPBED. The key objectives of the project are outlined, and it is explained how the models in the modelling tool OPNET can be adjusted to establish a modelling platform for evaluating the performance improvement of allowing the application to provision resources directly in the network.

Key-Words: Research networks, experimental test bed, applications, modelling.

1 Introduction

The bandwidth and capacity of the interconnections in the Internet has reached several key-points potentially allowing for transmission and use of high demanding research applications. However, no guarantees exist in the Internet Protocol (IP) for the bandwidth available for specific applications regarding the delay and delay jitter. Such guarantees are required by high demanding research applications like high quality uncompressed video transfer, video conferencing and grid applications. Supporting such demands requires that bandwidth in the transport network for the application is allocated for each group of applications.

The objective of the European IST project MUPBED is both horizontal and vertical integration of networks and applications. Horizontally, the integration is established by interconnecting five larger European testbeds through the common network of GÉANT operating with GMPLS and ASON transport technologies.

Vertically, the objective is to integrate the applications with the transport control layer through establishing a resource communication channel. This enables the applications to trigger bandwidth allocation, i.e., when a storage application initiates transmission, it request for appropriate bandwidth that, if available, is allocated until the transmission is completed. The parameters to use for such communication is determined by the types of applications in future research networks, the application requirements and the communications resources that are required for this interlayer interaction.

As a first approximation, four types of applications have been selected.

Transmission of uncompressed video: The required bandwidth requirements are around 300 Mbit/s with latencies less than 200 ms.

High quality video conferencing demands high bandwidth along with low latency and jitter.

Massive data transfer is expected to benefit from dynamic resource allocation as sessions are well-defined with huge bandwidth requirements.

Grid application, which requires access to resources high flexibility, short session times and resiliency.

The MUPBED project is clearly an experimental project, which utilises the interconnected test bed for identifying the problems and the solutions. However, while the experimental work provides results for the specific network setup, modelling studies are required in order to reproduce, verify and analyse the results. In addition, modelling and simulation activities are required for addressing the scalability of the application for even larger networks.

Hence, modelling activities are initiated in parallel to the experimental work. The objective is to increase the understanding of the obtained experimental results and to guide the deployment of applications in the test bed. Besides the analysis, the modelling work will also indicate how the network and the applications are affected by modifications of the network or applications.

First, in section 2, the objectives of the modelling activities are highlighted and an overview of the proposed modelling scenarios is given. Then, in section 3, the relevant models in the OPNET modelling tool are described, and it is shown how LSP setups, application modelling etc. are implemented into the tool. However, some problems with the integration of the MPLS path establishment and the RSVP for resource reservation have been identified, why this is described in details in section 4. Finally, a conclusion of the work is provided in section 5.

2 Objectives of modelling

With access to a European scale experimental test bed one could ask, whether modelling and simulations offers anything, which could not be addressed through real measurements. While it is doubtless that the measurements of the performance of the applications in the network provide accurate and real world results, they might be disturbed by a number of factors that cannot easily be controlled. Simulation can help in these cases. Sometimes simulations are even superior to measurements because all conditions are known, which would elsewhere require that the measurements are done in a perfectly well known and well controlled environment.
However, it makes no sense to implement simulations if the results are not in correlation with the real world measurements, why the results of the simulations are verified with experimental data and vice versa.

2.1 Scenarios

The modelling activities are divided into three main scenarios. First, a network with similar properties as the MUPBED network is modelled with pure IP routing. This will serve as a basic reference and the model is continuously updated with experimental information from the test bed to ensure valid model results. Secondly, manually installed routes are included in the model from end to end with given requirements to QoS. This indicates the potential benefit of traffic engineering across multiple administrative domains. Thirdly, the performance of the integration of the application layer with the resource reservation is evaluated.
For each performance study a topology conceptually similar to the one described in [1] is used. It should be clearly stated that it is not the intention to make a complete emulation of the network, as this will hardly add any useful information. The idea instead is to simulate how different parts of the network are influenced by background traffic, delay parameters etc.
OPNET provides a custom application model, where each task, phase and message of the application control can be modelled. Video conferencing and video transmission and storage applications are candidates for the modelling activities. For the video conferencing built-in models from OPNET can be modified and for storage applications new models should be developed based on the actual applications that are deployed in the MUPBED test bed.
In the second scenario traffic engineering (TE) is manually deployed using static label switched paths. This enables path differentiation between traffic flows, which is a key point in reducing the delay and delay jitter while increasing the guaranteed bandwidth for a certain connection. This is illustrated in Figure 1, where the dashed and the solid flows are specified to use different paths. Utilising GMPLS makes it possible to use dedicated links for the complete path or parts of it.

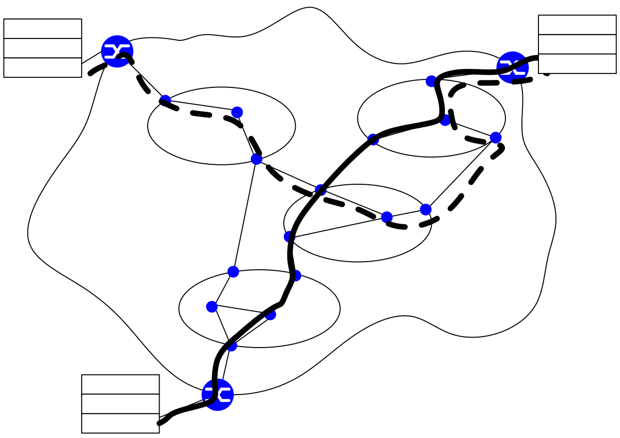

Figure 1: GMPLS network with static path setup. This setup is used to study the impact of traffic engineering for the selected applications.

The modelling work in this area has two focus points. First, the performance of the applications with manually deployed LSP is evaluated in terms of reduction in delay and delay jitters as previously discussed. Secondly, the horizontal integration of the different GMPLS administrative domains is currently one of the focus points in the CCAMP working group at IETF, and the performance studies will provide input to this standardisation work.
Another key point of the MUPBED project is the vertical integration of applications and network control layer. The applications communicate their QoS demands to the network control layer, which is then responsible for setting up path if the resource requirements are met.
In the performance studies this is modelled by including a network resource requester (NRR) and a network resource provider (NRP) layer in the cli-

ent and the network model, respectively. This is sketched in Figure 2, where the client side refers to the host and the network side refers to the edge router in the network

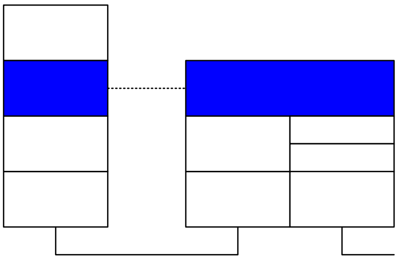

Figure 2: Adaptation layer added to the control plane of the applications in order to communicate basic resource requests to the edge router.

In the client model in OPNET, the NRR layer in the control plane is added as a new process model, why no modifications are required to the TCP, IP, RSVP and UDP models. Furthermore, in the edge router model the NRP layer is also included as a separate process model that polices the resource request before sending a specific path setup through the RSVP-TE layer.

The results are used to indicate whether the triggering levels for establishing and tearing down established paths should be adjusted. In parallel, it is expected that the studies in this scenario will indicate how the applications can be adjusted to benefit further from a dynamic infrastructure.

3 Implementation in OPNET

The networks today including the MUPBED test bed comprise interaction of several protocols at several different OSI layers. Such complexity disables the possibility of using fully analytical tools to model the influence of modifications and upgrades.

OPNET modeller offers complete models of routers, switches, clients, applications and protocols including IP, MPLS and a selection of standard applications like email, http and video conferencing. Each of the models can be adjusted as required and/or additional models or model components can be written as OPNET offers a complete platform for developing models. This enables the developer to make both initial studies based on standard components and to develop accurate and detailed models for specific evaluations.

3.1 Application model in an IP network

The first network topology modelled in OPNET is the pure IP network. This model is configured as a WAN network. OPNET provides preconfigured workstation, server, Label Edge Router (LER) and Label Switching Router (LSR).

Based on the created network topology, the next step is to add traffic to the network. OPNET provides two forms to generate background traffic:

Traffic Flows: The traffic flow is assigned manually on an end-to-end link and traverses the network from source to destination. In OPNET, the traffic profile is specified by an x-y graph, which is presented as the rates of traffic (in terms of packet per second or bit per second) versus time (in terms of second), as shown in Figure 3.

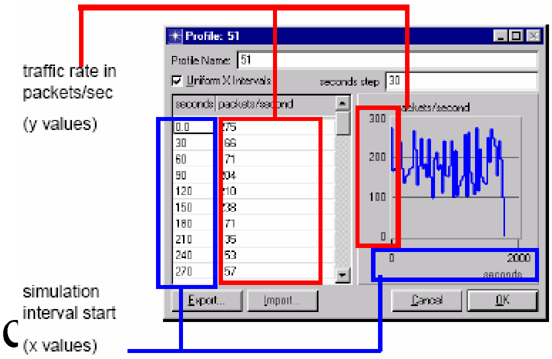

Figure 3: Specifying traffic flow profile attribute in OPNET.

Application Demands: The traffic flow is generated based on applications such as FTP, HTTP, e-mail, voice and video. The characteristics of the application traffic pattern are configured in a network application model. There are various standard application pre-defined by OPNET. For each application, e.g., the video conferencing application, its behaviour can be modified through parameters and represented corresponding to actual requirements. Figure 4 shows a sample Video Conferencing application configuration. The application traffic is specified as explicit traffic between each source (client) and destination (server) pair. Each application can be enabled or disabled on the client nodes, and each can be specified as a supported application service type on the server nodes. Besides the built-in application models, the tool provides an application framework for completely custom defined applications where each single control message can be implemented.

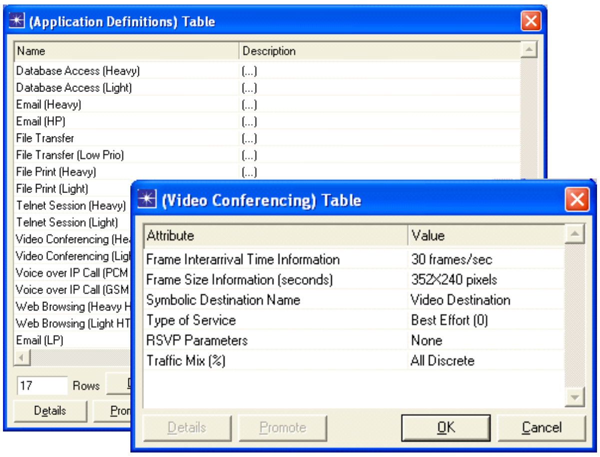

Figure 4: Sample Video Conferencing application configuration.

3.2 Description of the MPLS model

In Figure 5, traffic flows with source address from Ireland2 are specified into FEC_ireland2 and traffic flows with video conferencing application are specified into FEC_video. In order to direct an FEC into its routing path, LERs are configured to create traffic engineering binding that govern packets to be sent into LSP. In this scenario, FEC_ireland2 are directed to red (light grey) colour LSP and FEC_video are directed in blue (dark grey) colour LSP.

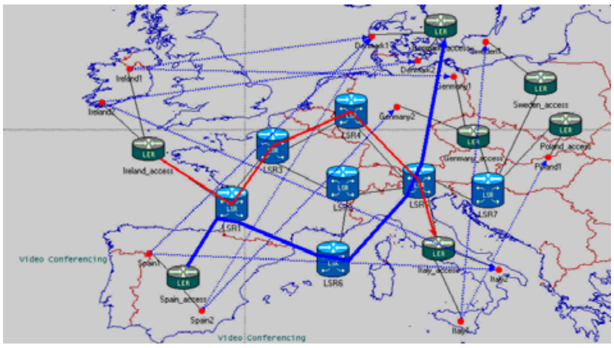

Figure 5: MPLS network in OPNET.

After configuring MPLS in network model, the traffic flows through network via different routes, resulting in balancing network bandwidth resource.

3.3 Description of RSVP signalling model

In the MPLS network model, the LSPs can be created either in static way or in dynamic way. With the static LSPs, one can specify the explicit route by clicking on every router in the OPNET workspace. To create the dynamic LSPs, you specify only two end points of LSP and the explicit route is determined by MPLS routing protocol (e.g. constrained OSPF). Once the explicit path is assigned, the Resource reservation Protocol (RSVP) is used to establish the forwarding state and reserve resources along the path.

RSVP mechanism

RSVP is a network layer signalling protocol that allows applications to reserve resources for data flows. RSVP treats traffic flows from a sender to a receiver and makes reservation for traffic on the outgoing links.

The reservation process is shown in Figure 6. RSVP carries its information in two types of messages: Path and Resv (reservation). The Path message is sent and creates a path state in each router that is traversed. When a receiver gets a Path message, it checks bandwidth resources and makes decision of resource reservation. Once the reservation is established, the receiver returns a Resv message along the path back to the sender. The reservation is made on a hop-by-hop basis, where all nodes along the path are required for resource reservation.

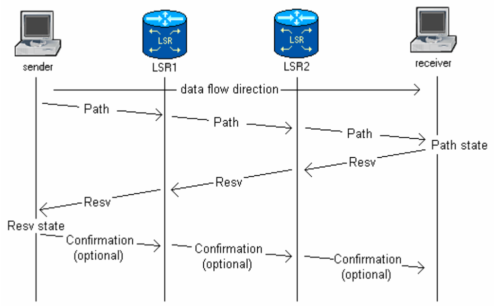

Figure 6: RSVP reservation process

RSVP model in MPLS network

In MPLS networks, RSVP is used as signalling protocol for creating, maintaining and withdrawing LSPs. A new object, the label object, is added to the Resv message. The reservation process is similar to the description above. However, in this case, a LSR allocates a label from its pool of free labels and adds this Label object into the Resv message.

Reconsider the previous example shown in Figure 5, which illustrates how the RSVP establishes MPLS forwarding state along an explicit route (the blue (dark) LSP). The LER Spain_access constructs an *Explicit Route Object* (ERO) that contains a sequence of four routers (LSR1, LSR6, LSR5, Denmark_access). The ERO object is included in the Path message. The Spain_access sends an RSVP Path message to LSR1 over link. LSR1 receives this Path message, checks next

nodes, LSR6, and finds the link that connects it to LSR6. LSR1 then modifies the ERO by removing LSR1 and sends out the Path message that contains the new ERO object. LSR6 and LSR5 handle the Path message in a similar process. The ERO that LSR5 sends to Denmark_access contains only Denmark_access.
When the Path message arrives at Denmark_access, it examines ERO and finds that it is the last node. After checking available resource, Denmark_access creates a Resv message and sends it to LSR5. The Resv message consists of a label object. LSR5, when it receives this message, uses this received label to populate its label forwarding table. Another label is assigned into Resv message and sent to LSR6. Continuing this process, the message is finally received by Spain_access. At this point, the LSP for the explicit route is established.

4 Integration of models

The success of the modelling in MUPBED is based on the interaction between the application model and the MPLS path establishment procedures. As discussed in the previous section, the signalling protocol for path setup in OPNET MPLS is RSVP. However, the way RSVP is used for resource reservation in connection with the upper layers is quite different from how the model is used as signalling protocol for LSP setups.
This separation of the functions very much reflects the common understanding of who is responsible for what. The LSP setup is solely done at the network level by the network operator independent of the applications to be deployed in the network. In this way MPLS enables traffic engineering and the service provider can use MPLS to ensure optimal utilisation of the network. The RSVP process used for resource provisioning is only intended (and documented) to be used on top of an IP network. It does not consider which kind of underlying topology the network is based on as long as the established path provides the requested QoS.
This section sketches how the OPNET MPLS model utilise RSVP for setting up LSPs between sources and clients. Secondly, the open issues and identified modifications to the models are reviewed.

4.1 LSP path setup in OPNET

There are two built-in approaches for setting up LSPs in OPNET models with the MPLS model. First, a GUI is provided for manually drawing static or dynamic LSPs between edge devices. Alternatively, a file with the pre-calculated traffic demands can be applied to setup the LSPs with different starting times as defined by the input file.
In the MUPBED concept, however, these offered interfaces for setting up label switched path are insufficient, as they do not allow the applications to directly request the path. Modifications to the models are required, why the rest of this section describes how an LSP is established in the standard models to indicate the necessary model modifications.
In Figure 7, a simple MPLS network is shown with a client, a server, two E-LSRs and three T-LSRs.

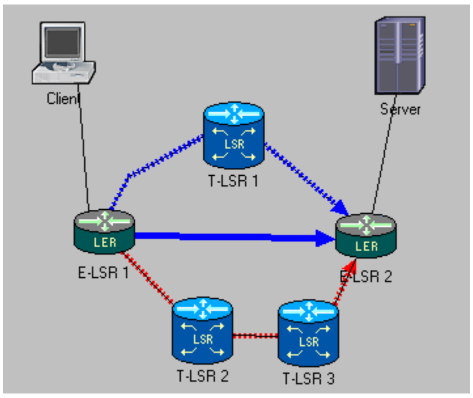

Figure 7: Simple example scenario for dynamic LSP setup.

In addition, a traffic demand from E-LSR1 to E-LSR2 is defined (solid line). To transport this traffic, an LSP is specified from E-LSR1 to E-LSR2. As the shortest path between these two edge-devices are through T-LSR 1, the OSPF routing protocol will inform the MPLS setup to use this path, which is shown with the upper dotted line. T-LSR1 is specified to break down at a given time, why the LSP is re-established through T-LSR2 and T-LSR3 as shown with the lower dotted line. The re-establishment is based on the new convergence of the OSPF routing protocol.
The RSVP process is triggered by three different kinds of interrupts:
Stream interrupt: Is initiated when an RSVP message is received at the node. The process determines if it is a PATH or a RESV message and processes it accordingly.
Self-interrupt: This is scheduled when the router is the egress router and it should respond to a PATH message. The interrupt then initiates transmission of a RESV message.
Remote interrupt: This is important for initialisation of the MPLS and RSVP integration.

When the simulation is initiated all the processes are initialised and the *MPLS manager* process sends a remote-interrupt to the RSVP processes (one for each LSR), which each calls a function that extracts a list of all LSPs originating from this node associated with the start time, end time and other parameters. A procedure interrupt is scheduled at the start time to initiate the LSP setup. This is done through *rsvp_send_label_request()* function, which extracts the explicit path to the destination from the routing protocol. The explicit path is stored in an ERO that is attached to the RSVP Path message. Finally, the PATH message is transmitted using the ERO information.
Incoming RSVP packets activates the RSVP process through a stream interrupt, and if the message is a PATH message the *rsvp_process_path_msg()* function is called which again calls *rsvp_route_for_unicast_find()* function. This function extracts the explicit route object. In contrast to [4] the explicit route stack is *not* popped, rather all the hops in the list are investigated and if a hop is the same as the current node, then the next hop is used for forwarding.
It is clear how the explicit route object is generated and how the explicit route list is extracted from the routing protocol. However, it is still unclear how the model works if no LSPs are predefined. In this case the procedure interrupt to the *rsvp_send_label_request()* procedure should be done manually through the application to rsvp interface process. Secondly, when the node T-LSR1 fails the *rsvp_send_label_request()* procedure is reinitiated, however, it is unclear what initiates it whether it is an interrupt or based on the rsvp notification messages.

4.2 Modifications to the processes

Some modifications to the processes are required if the applications should be able to directly request LSP setup. The RSVP label setup is initiated by a remote interrupt to the *rsvp_send_label_request()* function, which then asks for explicit route from the routing protocol and transmits the PATH message. The first modification is to schedule this interrupt correctly from the application side. Secondly, a mechanism should be found to clearly identify which data belongs to the LSP when it is eventually established.

5 Conclusion

The modelling activities within the MUPBED framework aims to support the result analysis of the experimental results obtained from the experimental test bed. The modelling can especially support the understanding of the application performance when the network size increases even beyond the size of the test bed.
Three main scenarios are suggested. First, the modelling of the applications in a conventional IP network is proposed, which acts as a reference scenario. Secondly, a traffic engineered path through the network is defined in order to indicate the potential benefit of traffic engineering seen from the application perspective. By adjusting the amount of delay and background traffic this can roughly be used to estimate the advantage of QoS supporting paths. The third scenario deals with the integration of the applications with the network control to help the definition of a generic interface.
It is examined and demonstrated for these scenarios how they can be implemented in the commercial modelling tool OPNET. For the first two scenarios the standard models can be used by adjusting the parameters to reflect and align the results with the MUPBED experimental application performance results. It is, however, not an objective to emulate the MUPBED test bed rather to simulate the performance when the network scales, when special situations occur etc. The latter allows for much more generic results than a complete emulation would do. The third scenario causes some troubles as the definition of LSPs are considered a “service provider-task” and the resource reservation in IP is considered a “client-task”. This very much reflects the common thinking that the operator takes care of the LSP setup and is only concerned in transporting bits, and the application just considers the network as “something” that transports the application data. The RSVP and MPLS model were initially studied to identify how they can be modified to facilitate the requirements from MUPBED.

References

[1] H.-M. Foisel et. al. “MUPBED deliverable D3.1: Test Bed Overview”, Project deliverable, European IST project MUPBED, Dec. 2004
[2] Y.1541 (2001) Network performance objectives for IP-based services.
[3] Dimitrios Miras (2002). A survey on Network QoS Needs of Advanced Internet Applications, University College London.
[4] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivsan, G. Swallow, “RFC 3209, RSVP-TE: Extensions to RSVP for LSP

Tunnels“, *Internet Engineering Task Force (IETF)*