Cooperation of Control and Management Plane for the Dynamic Provisioning of Connectivity Services on MPLS Networks

Eduardo Grampín Castro

Thesis Director - Dr. Joan Serrat Fernández

Departament de Teoria del Senyal i Comunicacions
Universitat Politècnica de Catalunya

A thesis submitted for the degree of
Doctor per la Universitat Politècnica de Catalunya

A Alicia y Mario...

A Daniel, Raquel y Federico...

A Laura y Mauro...

Abstract

Next Generation Networks multimedia services are creating a revolution in everyday communications. New applications are emerging continuously, demanding ever growing resources to transport networks.

Dynamic Provisioning is needed to support such demands.

MultiProtocol Label Switching has embedded intelligence on the network, through a Control Plane that enables the integration of multiple layers for timely and optimal mapping of resources to demands.

Nevertheless, the Path Computation problem demands stringent processing resources of network device, which must be devoted to the transfer of information at terabit speeds. Thus, provisioning tools are needed to assist the network in these challenging tasks.

This thesis proposes a synergistic, hybrid solution between the Control and Management Plane for dynamic provisioning of connectivity services in MultiProtocol Label Switching networks, and defines a practical operational Traffic Engineering strategy for Service Providers. The proposal introduces an architecture for connectivity provisioning, founded over a new network entity, the Routing and Management Agent, capable of providing Path Computation features to the evolved network, with peering communications in both the Management and Control Plane.

The proposed architecture has been extensively tested in a simulated environment, whereas real testbed prototyping is under development.

Contents

List of Figures

Chapter 1

General Introduction

1.1 Motivations

This doctoral thesis proposes a synergistic, hybrid solution between the Control and Management Plane for dynamic provisioning of connectivity services in MultiProtocol Label Switching (MPLS) networks, and defines a practical operational Traffic Engineering (TE) strategy for Service Providers (SPs). The technological context of this work are the Next Generation Networks (NGN), with focus in the broadband IP over Optical Transport Networks infrastructure, founded on the Generalized MPLS (GMPLS) Control Plane. MPLS augments classical connectionless IP networks with TE capabilities, and permits to handle traffic aggregates (flows) instead of individual data packets, which is advantageous for the routing processes, and more precisely for the allocation of network resource to traffic demands. IP traffic is grouped by some criteria in Forwarding Equivalence Classes (FECs) and assigned to Label Switched Paths (LSPs) which enable loop-free source routing with TE capabilities. In summary, IP is augmented with manageable connection-oriented capabilities by MPLS and its related protocols and mechanisms, as explained throughout this document (see Chapter 3).

Provisioning is a matter of allocating resources to traffic demands. It consists of a two phase process which involves:

- Path Computation
- Connection Establishment

Path Computation is a complex process which may include connection admission decision (Access Control), TE optimization and connection planning, at different timescales.

Connection Establishment implies configuration of network elements to actually transport a given flow end to end through the network, and may be achieved by the Management Plane (i.e. configuration of participating nodes

using SNMP or other management protocol) and/or by the Control Plane using signaling protocols, like the ReSerVation Protocol with Traffic Engineering extensions (RSVP-TE), which is considered in the present work.

Since resources are finite, network operators face the challenge to satisfy individual connectivity demands while maintaining global optimization; the dynamic allocation of resources shall avoid denial of service (i.e. every single request shall be satisfied). In other words, the Service Provider objective is to minimize blocking probability and optimize resource sharing while Quality of Service (QoS) assurance is preserved for every user. Meeting both *resource* and *traffic* (or *user*) oriented objectives is regarded as a *rational* TE objective[Awd99].

Transport networks are evolving from traditional overlay models into an integrated Optical Transport Network (OTN). This evolution is shown in Figure 1.1. The 90's model stacked IP over Asynchronous Transfer Mode (ATM) over Synchronous Digital Hierarchy (SDH/SONET) over Optical Fiber physical layer, as shown in (a); the model is evolving to IP/MPLS (which may involve several Layer 2 technologies as ATM, Frame Relay or Ethernet) over SDH/SONET over Dense Wavelength Division Multiplexed (DWDM) fiber, as shown in (b). DWDM permits to multiply fiber capacity, while SDH/SONET assures automatic protection capabilities to the client layers in the hierarchy. IP/MPLS integration, as will be explained in detail throughout this document, enables multi-technology integration in a common Control Plane, among other advantages. The challenge is the evolution towards an integrated IP/Generalized MPLS (GMPLS) model over an All-Optical Network, as shown in (c). The GMPLS common Control Plane is the foundation for layer peering, which provide visibility between layers, enabling better resolution of the dynamic allocation of resources to traffic demand. Nevertheless, due to business specifics, sometimes it is not possible, even desirable, to establish peering relationships. The different emerging models for layering interactions in the Optical Transport Networks are discussed in Chapter 2.

1.2 The Operational Challenge

Research work devoted to the problem of Traffic Engineering and QoS assurance in IP networks, and specifically in MPLS-enabled networks, often formulates the allocation of network resources to traffic demands as an optimization problem. There are different approaches in the election of the objective functions, constraints, exact algorithms and/or heuristics, but basically the problem of network optimization is solved under certain hypothesis and for chosen topologies, assuming the existence of a management system capable of translating the mathematical solution (i.e. LSPs that satisfy a system of traffic demands) into actual network configuration. Besides, a metering

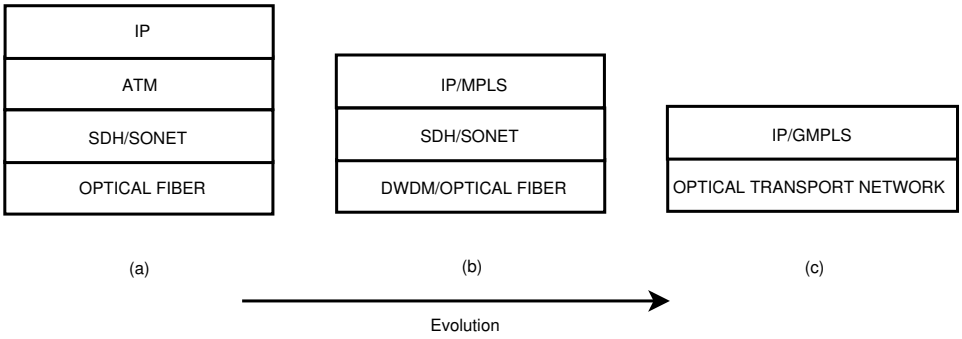

Figure 1.1: Transport Networks Evolution

artifact is also supposed to be in place in order to provide the necessary feedback to the TE system, which can be considered an adaptive control loop.

This thesis concentrate in the operational aspects of the problem, and one of its major objectives is to build practical procedures for the provision of network paths (LSPs) under operational constraints in near real time.

Service Providers willing to deploy multiservice networks need an operational framework to support Service Level Agreement (SLA) guarantees, QoS assurance and overall interworking service management. Among the objectives of TE, is worthy to mention the control and optimization of the routing function, which should satisfy the QoS requirements for every admitted connection, achieving global efficiency in sharing resources [AMA+99]. Constraint-Based Routing (CBR) is the TE mechanism for computing a feasible network path based on a traffic description and a set of constraints. CBR, as a generalization of QoS routing, evolves from current topology driven hop-by-hop Internet Interior Gateway Protocols (IGPs). Since CBR with more than one restriction is known to be a NP-complete problem, the needed computation power to solve the general problem is unbounded. This prevents the introduction of full CBR capabilities into network devices, since they have scarce computational resources mainly devoted to packet forwarding. Moreover, different implementations of CBR algorithms lead to the impossibility of fulfilling network-wide TE objectives.

Commercial Label Switched Routers (LSRs) usually perform Control Plane LSP setup using IGP next-hop routing, which cannot ensure the fulfilment of QoS and policy/administrative constraints. This behaviour can be slightly improved by Constraint Shortest Path First (CSPF), which provides some basic online CBR, but, where possible, only performs local optimization. The network, considered as a whole, tends to operate far from its optimal state, and therefore waste resources, that could lead to revenue losing by Service Providers.

On the other hand, solutions for connectivity setup by the Management Plane usually perform off-line route computation on a Path Computation Server (PCS). This solution, while logically correct and robust for near static networks, cannot meet the timing requirements for dynamic provisioning. The update of network information often requires several interactions between the management application and network devices, and inaccuracies are very likely to happen. Delay is also relevant in provisioning time, when the management application must communicate with every LSR along the LSP path being configured. Another problem is caused by equipment vendors, which usually provide non-standard management applications, as a market differentiator from competitors. It is very unlikely to manage certain equipment with other vendor's management software.

Therefore, providing an operational context that enables the introduction of enhanced routing functionality to MPLS networks is another major objective of the thesis. The Constraint-Based Routing problem may be solved in near real time using well-known algorithmic and heuristics; this aspect of the provisioning process is out of the scope of the thesis, but approaches that provide access to the functionality and possible hardware/software architectures that support this CPU intensive task are proposed in this work.

The general thesis objective is, in other words, to find provisioning mechanisms with setup times as shorter as those realizable by Control Plane signalling, and Traffic Engineering capabilities as powerful as in management systems.

There are some solutions for some of these operational challenges in the intra-area, intra-domain scope, but inter-area and, more challenging, inter-domain solutions are yet to be built. Traffic Engineering across Service Providers boundaries is a major burden, given the fact that the exchange of topology information is usually banned by confidentiality reasons. This thesis explore solutions on this area, trying to evolve from traditional inter-domain routing solutions based on Border Gateway Protocol (BGP) to a practical inter-domain TE framework.

1.3 Thesis Contributions

The proposed approach to achieve Control and Management Plane synergy is the insertion in the network of an entity called the *Routing and Management Agent* (RMA), which performs online CBR and acts as a peer network node for signalling purposes but excluding packet forwarding. Being a logically centralized platform, the computational resources of the RMA can be considered unbounded in practical terms. Different RMAs can cooperate together at the Management Plane for long term global optimization and/or inter-provider connectivity setup, applying appropriate routing policies.

As mentioned before, this work assumes that a common MPLS Control Plane is in place in the considered network. This thesis proposes contributions in the intra-area and inter-area (intra-domain) provisioning, and explores the inter-domain provisioning case.

The proposed *intra-domain* provisioning techniques rely on a complete knowledge of network topology, acquired by peering with the IGP processes. The *inter-domain* case has to deal with partial information and relies on peering at the management plane with traffic guarantees built with MPLS tunnels across the inter-domain network infrastructure.

The RMA solution achieves provisioning times closer to Control Plane signalling, and enhances the routing function of the network providing access to a powerful CBR engine. Moreover, the RMA can be used as a Traffic Engineering optimization tool by management applications, integrating many Control and management functionalities in a single platform.

1.4 Document Organization

This thesis is structured as follows: Chapters 2 describe the technological context of this thesis work, centered in convergent multiservice Next Generation Networks, while Chapter 3 addresses Traffic Engineering in MPLS network, and more specifically the main issue of LSP provisioning with Constraint Based Routing. Chapter 4 reviews the state of the art in MPLS Management Plane provisioning, and Chapters 5 and 6 are devoted to propose and evaluated the thesis contributions. General concluding remarks are given in the final Chapter 7.

Chapter 2

Technological Context: Next Generation Networks

2.1 Introduction

Telecom operators have long been seeking service integration in a single infrastructure. The appearance of the ATM technology in network backbones has fuelled this objective in the past, but since this approach could not cope with the challenge, new alternatives are under development. Convergence is driven by the foreseen capital and operational (CAPEX and OPEX) cost reductions with new expected revenues for the operator. The Internet success has imposed IP as the natural choice for integration. The evolution of nowadays IP networks towards an integrated multiservice infrastructure requires the adaptation of IP and its related protocols to provide a service transport capable of offering service differentiation and a flexible adaptation to new services through signaling and control functions. The architecture appearing as the natural evolution towards this integrated services infrastructure is know as Next Generation Networks (NGN).

NGN definition and main characteristics

The term Next Generation Networks is used by the telecommunications industry to describe the developing set of standards for future networks that will be able to carry a wide range of services, including voice, data and multimedia, over packet transportation. The European Telecommunications Standardisation Institute (ETSI) defines NGN as:

"A concept for defining and deploying networks, which, due to their formal separation into different layers and planes and use of open interfaces, offers service providers and operators a platform which can evolve in a step-by-step manner to create, deploy and manage innovative services."

The ITU-T definitions of the NGN concept include a number of fundamental characteristics such as the use of packet-based transfer mechanisms, de-

coupling of service provisioning and network access, increasingly separated control functions for bearer resources, calls or sessions and services or applications, and support of generalised mobility, while supporting interworking with existing networks and assuring end-to-end QoS.

The defining characteristics of NGNs can be summarised as:

- decoupling of transport network, service control and application layers;
- use of packet-based transfer mechanisms, rather than circuit-switched technology; and
- a "smart terminal, smart network" model.

A non exhaustive list of applications of NGNs are likely to include:

- voice telephony (classical audio only phone calls);
- video telephony (simultaneous audio and video);
- video streaming;
- interactive multimedia (i.e. online games, interactive TV);
- broadband Internet access.

The NGN seamlessly blends the public switched telephone network (PSTN) and the public switched data network (PSDN), creating a single multiservice network. Rather than large, centralized, proprietary switch infrastructures, this next-generation architecture pushes central-office (CO) functionality to the edge of the network, with the introduction of the Soft-Switch, a network device capable of delivering robust switching functionality, supporting all current analog and digital network standards, interfaces, media, and service elements, which provides call agent, signaling gateway and media gateway functionality.

- *Media Gateway* provides inter-operation with legacy networks, with the following basic functions:
  - The coding and packetization of media streams coming from a non IP network, and the conversion of IP packets to the media stream in the destination network.
  - The relay of the media streams as signalled by the media gateway controller.
- *Call Control/Media Gateway Controller* enable the signalling of media calls (real-time services like voice calls and streaming applications) using appropriate signalling protocols.

- *Signalling Gateway* provide inter-operation at the control plane.

The described functionality can be integrated into a single entity, or distributed among network elements, as depicted in Figure 2.1.

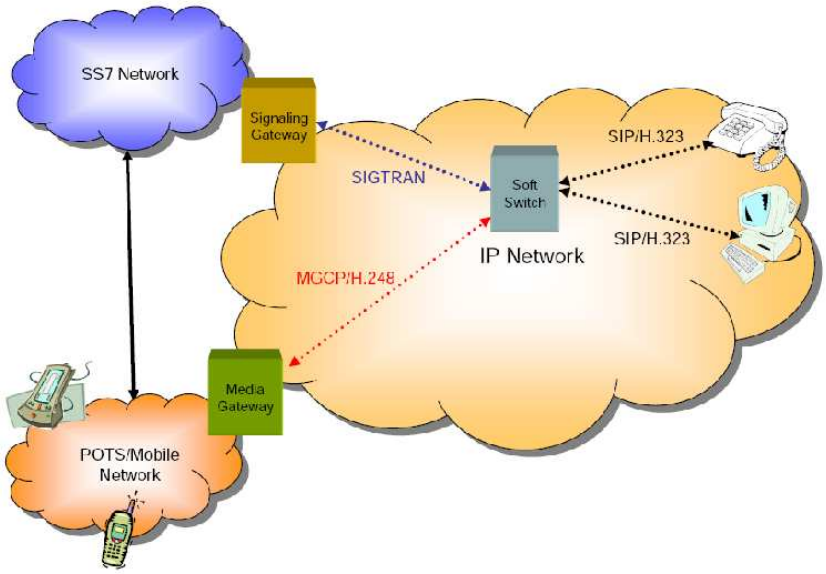

Figure 2.1: NGN Control Architecture (source: Arcome)

The Signalling Protocols in the NGN can be grouped by functionality:

- *Call Control Protocols:* enable the establishment of media communication from a terminal to another terminal or to a server. Candidates protocols are H.323 (ITU-T) and SIP (IETF).
- *Media Gateway Command Protocols* corresponding to the separation between transport and control layers in the NGN architecture. It allows the media gateway controller to control the media gateways. Candidate protocols are H.248/MEGACO (ITU-T/IETF) and MGCP (IETF).
- *Inter-Media Gateway Controller Signaling Protocols* for the management of the control plane. In the backbone the candidate protocols are Bearer Independent Call Control (BICC) and H.323 both from ITU-T, and SIP-T (IETF). Regarding the interconnection with legacy networks (in particular with SS7 networks), the corresponding signaling gateways implement protocols like SIGTRAN (IETF).

The provision of services in the convergent networks evolves each technological context: the Intelligent Network services (IN) for the telephone terminals

(fixed or mobile), and traditional Internet services such as Web, Mail, News, etc. for the IP networks. The evolution of access technologies, which multiply the available bandwidth to the broadband user, and the evolution in terminals capabilities push a transformation of the service platform. This new platform must allow a broad range of users to access services no matter the terminal and protocols used, and enable the user to recover its personal environment no matter what particular terminal he/she is using. Two different service offering models, based on adaptability and portability, are emerging:

- *Soft-switch Centered:* service architecture based on the OSA/PARLAY normalized service interface. This model is well adapted to telecom type services, requiring a strong participation of the call control entities.
- *Web Services Centered:* service architecture based on the protocols and technologies used in the Internet world (XML, SOAP), providing distributed services transparently transported on IP and with strong participation of the terminals.

NGN Transport Network

The most important component of the NGN in the context of this thesis is the Transport Network, which support the aforementioned services. The historical packet switching paradigm B-ISDN with ATM as the transport technology has established the foundations of nowadays broadband multiservice infrastructures. Despite its potential, ATM did not succeed in becoming the chosen technology for service integration. In particular, the lack of native ATM services (i.e. designed natively to use ATM as transport technology) is one of the main reasons why it was not widely adopted by operators as a single transport technology capable of integrating voice and data traffic. ATM is today being used mostly in the access to IP broadband networks (e.g. ADSL and cable operators). Most deployed infrastructures use IP transport technology and related protocols, making unavoidable the use of IP as agent for transport unified infrastructure in NGNs. The establishment of virtual circuits to assign resources to traffic demands, QoS control and policing functions, evolved link-state routing, among other important traffic engineering tools have been inherited by MPLS-based networks, which incorporate TE capabilities to native IP, enabling evolved network engineering such as optimal dimensioning in both static and dynamic environments.

2.2 Multiservice Broadband Transport Networks

Operational broadband networks are segmented into *Access*, *Aggregation* and *Core* networks. Several wired and wireless technologies are used in each seg-

ment. DSL, Coax and wireless broadband "first-mile" technologies are used in the *Access*, with network terminating devices such as DSLAMs or wireless Access Points. Traffic control can be performed using ATM techniques in the first-mile; for example per-service PVCs with different QoS parameters can be configured in the link to the broadband xDSL home user.

The *Aggregation* can be based on ATM, which supports both DSLAMs and corporate connectivity (legacy) services in the same infrastructure. The ever growing demand of new bandwidth-consuming broadband services and the business challenge for SPs to reduce their expenses by simplifying the aggregation architecture have paved the way to Ethernet as the aggregation technology of choice. Ethernet is a simple, ubiquitous and cost-efficient technology, and delivers more bandwidth than ATM. Building Metro Ethernet aggregation networks raise many operational issues, like traffic differentiation and QoS assurance, multicast support, fast restoration among others. There is an ongoing discussion between standardization fora, operators and vendors about this issues. This Metro Ethernet aggregation network deliver traffic between end users, content networks and the Internet, as depicted in Figure 2.2.

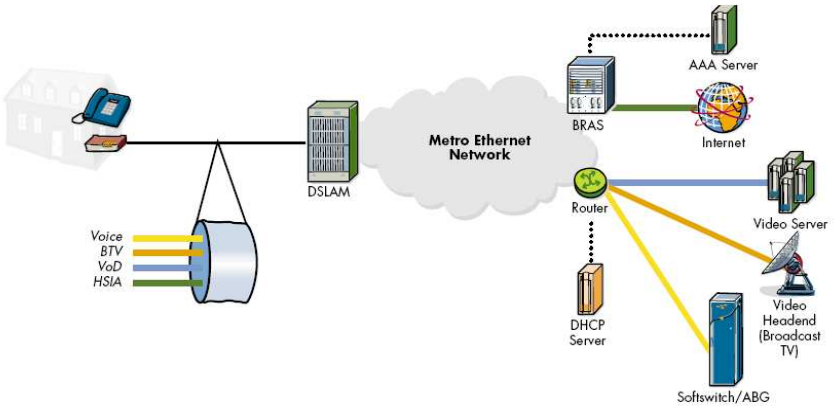

Figure 2.2: Metro Ethernet Aggregation (source: Alcatel)

NGN *Core* transport network is moving towards a combination of packet and optical technologies, as depicted in the Introduction, Figure 1.1(c). There exist two basic network models of the integration of IP and Optical Transport Networks: the Automatic Switched Transport Network/Automatic Switched Optical Network (ASTN/ASON) network operation concept developed by ITU-T [ITU01a], [ITU01b], and the Generalized Multi Protocol Label Switching (GMPLS) concept of IETF [Man04]. A third standardization fora, the OIF, is working in the definition of interfaces between Client and Transport Layers, as defined in section 2.3.

A comprehensive view of the contemporary network layers is shown in Figure 2.3.

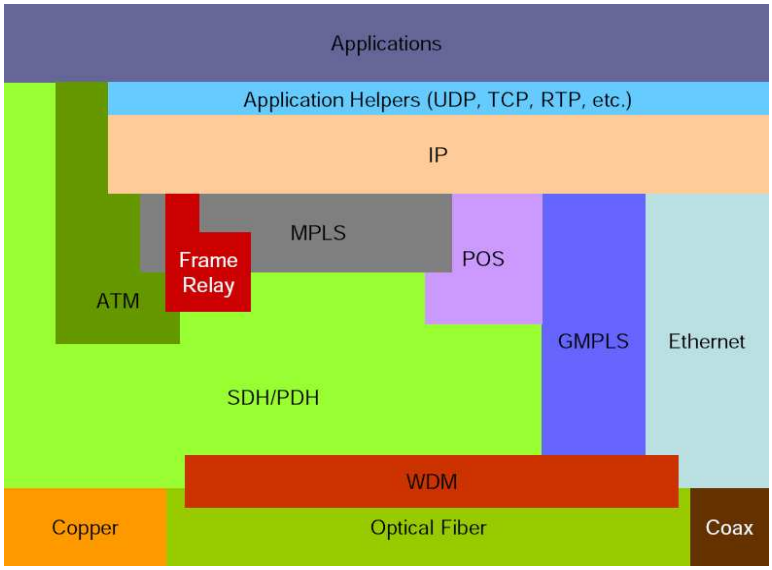

Figure 2.3: NGN Transport Layers (source: ITU-T/Arcome)

It is important to distinguish between the data and control planes when analyzing the architectural alternatives for IP over optical (IPO) networks. The optical network provides fixed bandwidth transport pipes (optical paths) to client layers. IP routers at the edge of optical networks must establish such optical paths before starting transfer IP packets. Thus, the IP data plane over optical networks is realized as a virtual topology of optical paths. In other words, even though there is ample attention to the subject of optical packet transport and switching, the present work assumes that the optical core is unable of processing individual IP packets in the data plane. On the other hand, there is a broad industry agreement on utilizing IP-based protocols for the Control Plane; therefore, IP routers and Optical Cross-Connects (OXCs) can have a peer relationship at the Control Plane, particularly for routing protocols that permit the dynamic discovery of IP endpoints attached to the optical network [RLA04].

A general network model consists of IP routers attached to an OTN, and connected to their peers over (dynamically established) switched optical channels. The optical internetwork consist of multiple optical networks, each of which may be administered by a different entity. Each optical network consists of sub-networks interconnected by optical fiber links in a general topology (referred to as an optical mesh network, not only interconnected optical rings). This network model is shown in Figure 2.4. As shown in the

Figure, other client networks than IP (e.g., ATM) may also connect to the OTN.

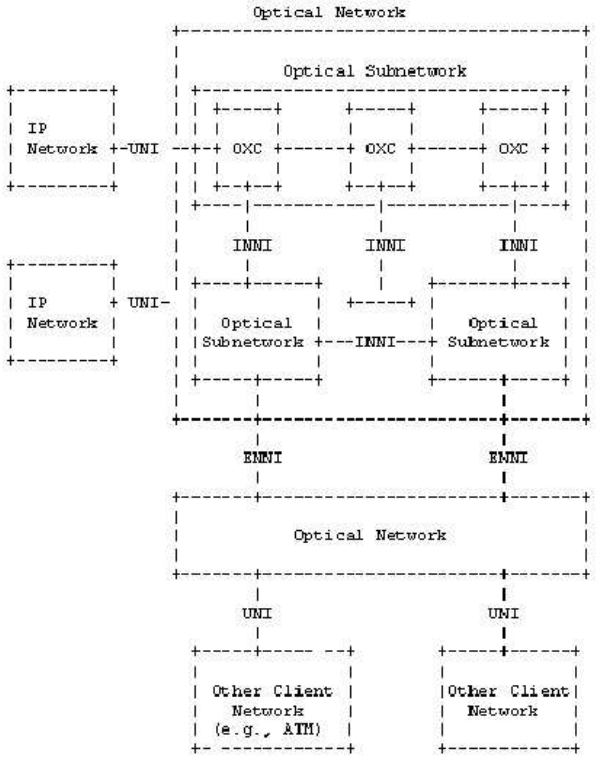

Figure 2.4: Optical Internetwork Model (source: IETF)

The logical control interfaces, as defined by the Optical Internetworking Forum (OIF) are:

- Client-optical internetwork interface or User-Network Interface (UNI) [The01].
- Internal node-to-node interface within an optical network (INNI).
- External node-to-node interface between nodes in different optical networks (ENNI).

These interfaces are capable of certain functionalities, that permit to define different service and interconnection models, which will be described in next section.

2.3 IP over Optical Service and Interconnection Models

Two general models have emerged for the services at the UNI, the *Domain Services Model* and the *Unified Service Model*. These models are described as follows.

Domain Services Model

Under this model, the optical network primarily offers high bandwidth connectivity in the form of light-paths. Standardized signaling across the UNI is used to invoke the following services:

1. Light-path creation: This service allows a light-path with the specified attributes to be created between a pair of termination points in the optical network. Light-path creation may be subject to network-defined policies (e.g., connectivity restrictions) and security procedures.

2. Light-path deletion: This service allows an existing light-path to be deleted.

3. Light-path modification: This service allows certain parameters of the light-path to be modified.

4. Light-path status enquiry: This service allows the status of certain parameters of the light-path (referenced by its ID) to be queried by the router that created the light-path.

Discovery procedures may be used over the UNI to verify local port connectivity and to obtain UNI services. The protocols for neighbor and service discovery are different from the UNI signaling protocol itself (for example, see LMP [Lan04]). Because a small set of well-defined services is offered across the UNI, the signaling protocol requirements are minimal. Specifically, the signaling protocol is required to convey a few messages with certain attributes in a point-to-point manner between the router and the optical network, e.g., such a protocol may be based on RSVP-TE or CR-LDP.

The optical domain services model does not deal with the type and nature of routing protocols within and across optical networks. The resulting overlay model for IP over optical networks is discussed in section 2.3.2.

Unified Service Model

Under this model, the IP and the OTN are a single integrated network from a Control Plane point of view, meaning that the the OXCs are Control Plane peers of IP routers. Thus, in principle, there is no distinction between the UNI, NNIs and any other router-to-router interface from a routing and signaling point of view. It is assumed that this control plane is IP-based, for

example leveraging the traffic engineering extensions for MPLS or GMPLS, as described in section 2.5.

Under the unified service model and within the context of a GMPLS network, optical network services are obtained implicitly during end-to-end GMPLS signaling. Specifically, an edge router can create a light-path with specified attributes, or delete and modify light-paths as it creates GMPLS label-switched paths (LSPs). In this regard, the services obtained from the optical network are similar to the domain services model, but may be invoked in a more seamless manner. For instance, when routers are attached to a single optical network (i.e., there are no ENNIs), a remote router could compute and establish an end-to-end path across the optical internetwork (a light-path).

The most significant difference between the service models is regarding routing protocols, as discussed in next sections.

As mentioned above, the IP over optical network architecture is defined essentially by the organization of the control plane. Depending on the service model, the control planes in the IP and optical networks can be loosely or tightly coupled. This coupling determines the following characteristics:

- The details of the topology and routing information advertised by the optical network across the client interface;
- The level of control that IP routers can exercise in selecting explicit paths for connections across the optical network;
- Policies regarding the dynamic provisioning of optical paths between routers. These include access control, accounting, and security issues.

The possible interconnection models between the optical and client layers are the Peer, Overlay and Augmented Models.

2.3.1 The Peer Model

Under the peer model, the IP control plane acts as a peer of the OTN control plane. This implies that a single instance of the control plane is deployed over the IP and optical domains. When there is a single optical network involved and the IP and optical domains belong to the same entity, then a common IGP with appropriate extensions can be used to distribute topology information over the integrated network. These extensions are defined within the context of GMPLS.

In the *Integrated Routing* approach, the IP and optical networks run the same instance of an IGP, e.g., OSPF-TE with suitable "optical" extensions. These extensions must capture optical link parameters, and any constraints that are specific to optical networks. Thus, a router can seamlessly compute and establish an end-to-end light-path to another router across the optical

network, using GMPLS signaling. This light-path is a tunnel across the optical network, and can be defined as a "forwarding adjacency" (FA) and advertised in the link state protocol (see section 2.5).

When an optical internetwork with multiple optical networks is considered (e.g., spanning different administrative domains), a single instance of an intra-domain routing protocol is not attractive or even realistic. In this case, inter-domain routing and signaling protocols are needed. In either case, a common IP addressing scheme is used for the optical and IP networks.

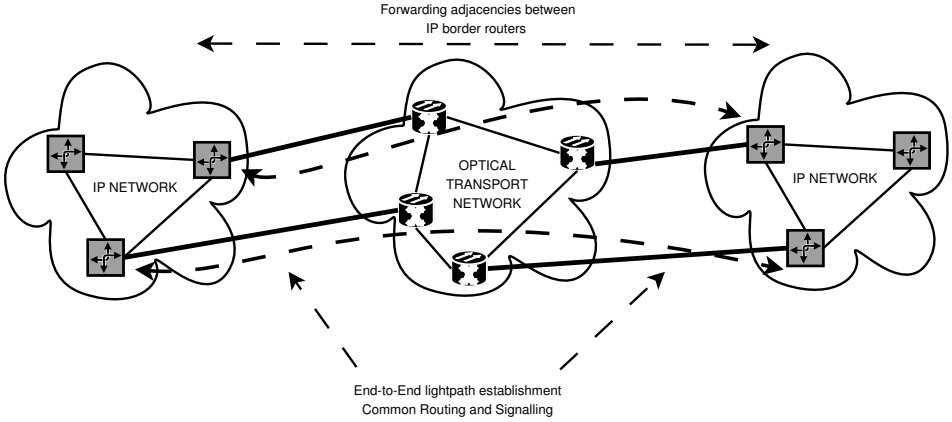

Figure 2.5: Peer Model

2.3.2 The Overlay Model

Under the overlay model, IP and optical domain routing, topology distribution, and signaling protocols are independent, as depicted in Figure 2.6. This model is conceptually similar to the classical IP over ATM model. In the overlay model, a separate instance of the control plane (especially the routing and signaling protocols) is deployed in the optical domain, independent of what exists in the IP domain. In certain circumstances, it may also be feasible to statically configure the optical channels that provide connectivity for the IP domain in the overlay model. Static configuration can be carried out through network management functions, but it is unlikely to scale in very large networks, and may not support the rapid connection provisioning requirements of future highly competitive networking environments.

The overlay routing approach supports the overlay interconnection model. Under this approach, an overlay mechanism that allows edge routers to register and query for external addresses is implemented. This is conceptually similar to the address resolution mechanism used for IP over ATM. Because IP-optical interface connectivity is limited, the determination of how many light-paths must be established and to what endpoints are traffic engineering

decisions.

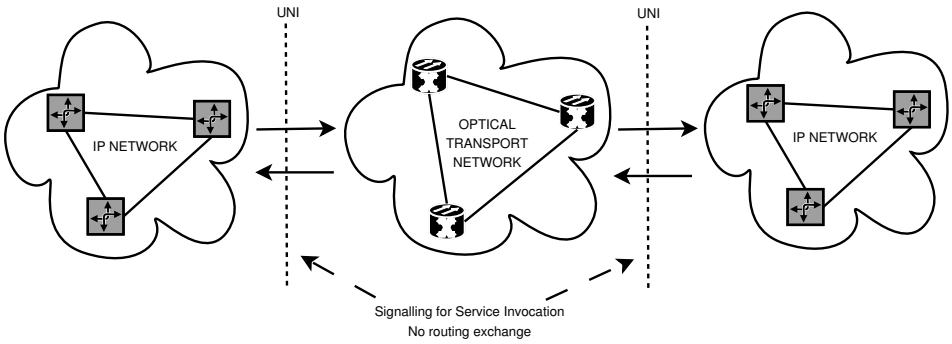

Figure 2.6: Overlay Model

2.3.3 The Augmented Model

Under the augmented model, there are separate routing instances in the IP and optical domains, but certain types of information from one routing instance can be passed through to the other routing instance. For example, external IP addresses could be carried within the optical routing protocols to allow reachability information to be passed to IP clients.

The *Domain-specific Routing* supports the augmented interconnection model. Under this approach, routing within the optical and IP domains are separated, with a standard routing protocol running between domains. A natural candidate for exchanging routing information between IP and optical domains is BGP.

This is a well-known routing framework from IP networks; there are some potential negative effects that could result from domain-specific routing using BGP in an IP over Optical (IPO) environment:

- The amount of information that optical nodes will have to maintain will not be bound by the size of the optical network, but will have to include external routes as well.
- The stability of the optical network control plane will no longer be dictated solely by the dynamics emanating within the optical network, but may be affected by the dynamics originating from external routing domains from which external reachability information is received.

Note that there is a great similarity between integrated routing and domain-specific routing, because both ultimately deal with the creation of a virtual light-path topology (which is overlaid over the optical network) to meet certain traffic engineering objectives.

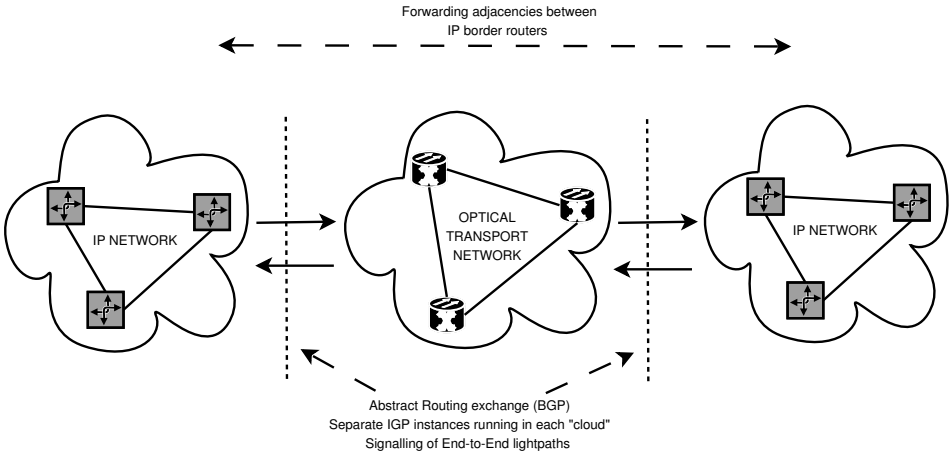

Figure 2.7: Augmented Model

These service and interconnection models must be instantiated under the aforementioned reference IP over Optical Transport Networks models, namely ASTN/ASON and GMPLS, that will be presented in the following sections.

2.4 ITU-T ASTN/ASON model

ASTN/ASON defines an Optical Network, which extends the Optical Transport Network (OTN) concept defined in [ITU01c] and [ITU03a], with the capability of fast and automatically provisioning end-to-end Optical Channel (OCh) connections via the control plane as the outcome of a request of any client layer such as IP/MPLS, ATM, SDH etc, through a UNI-type interface or by the management system of the Och layer and the use of a NNI-type interface. While ASTN [ITU01a] defines types of connection establishment and several control plane agents supporting dynamic call connection, ASON [ITU01b] describes the set of control plane components or abstract entities that are used to manipulate transport network resources in order to provide the functionality of setting up, maintaining and releasing connections, following the requirements of the ASTN. Simply speaking the ASTN/ASON model is basically an OTN equipped with a Control Plane supporting switched and soft-permanent connections.

The concept of switched connectivity is a newcomer in the ITU-T environment, used to networks of circuits provisioned by the management plane. Basically there is an evolution from the Provisioned to the Switched Connection as depicted in figures 2.8 and 2.9.

Three types of connections are defined:

- *Provisioned*: established by configuring every network element along the path with the required information to establish an end-to-end connection. Provisioning is provided either by means of management systems or by manual intervention. This type of connection is referred to as a *hard permanent* connection.

- *Signalled*: established on demand by the communicating end points within the control plane using a dynamic protocol message exchange in the form of signalling messages. These messages flow across either the I-NNI or E-NNI and UNI within the control plane. This type of connection is referred to as a switched connection. Such connections require network naming and addressing schemes and control plane protocols. In this respect IP addressing and GMPLS signalling is being adapted [ITU03b], [ITU03c].

- *Hybrid*: establishment exists whereby a network provides a permanent connection at the edge of the network and utilises a switched connection within the network to provide end-to-end connections between the permanent connections at the network edges (no UNI defined). Connections are established via network generated signalling and routing protocols. The establishment of such connections is dependent upon the definition of an NNI. Provisioning is therefore only required on the edge connections. This type of network connection is known as a *soft permanent* connection (SPC). From the perspective of the end points a soft permanent connection appears no different than a provisioned, management controlled, permanent connection.

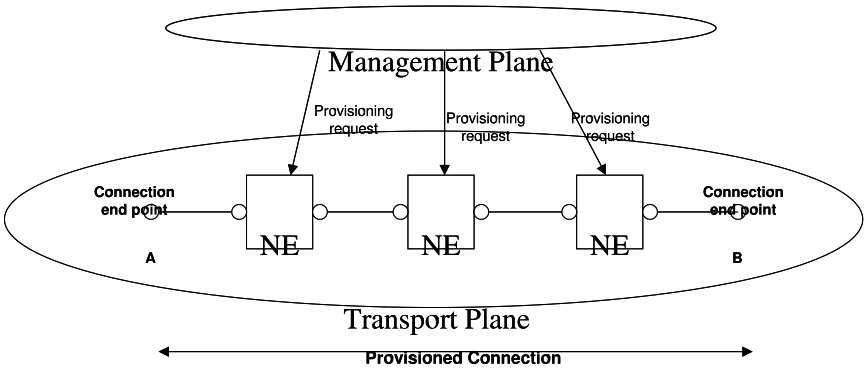

Figure 2.8: Provisioned Connection in the OTN (source: ITU-T)

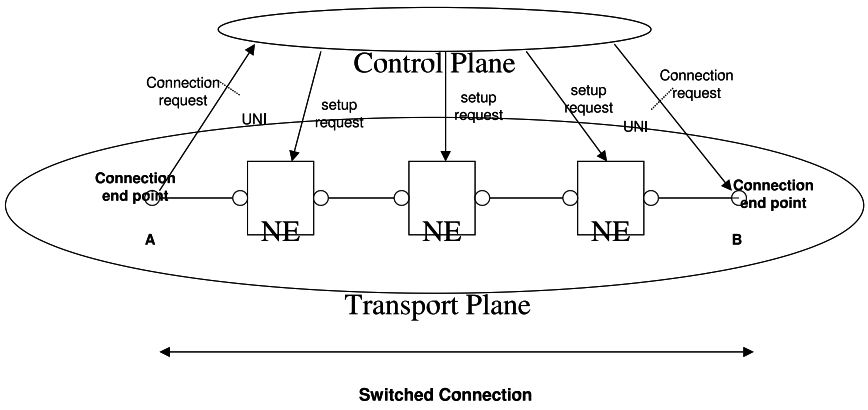

Figure 2.9: Switched Connection in the OTN (source: ITU-T)

The logical ASTN/ASON architecture is shown in Figure 2.10. Similarly with the B-ISDN ATM reference model it comprises from three main planes:

- The *Transport Plane*, which represents the OTN comprising the OXCs, and connectivity functions such as the sub-network connection function as defined in [ITU00].

- The *Control Plane*, which is comprised by a series of logical functional entities and interfaces that are needed for the processing and establishment of the Och connections. The main functional entity is the Optical Connection Controller- OCC which is responsible for the processing of the connection request. The OCC is the functional entity responsible for the interaction with the Transport Resource Agent (TPA) of the transport plane via the Connection Control Interface.

- The *Management Plane* framework is detailed in Recommendation [ITU05]. It places ASON management within the TMN context and specifies how the TMN principles may be applied. A management view of the ASON control plane is developed, which provides the bases for the ASON management requirements specified in this Recommendation.

  The NMI-A interface facilitates the interaction between the management plane with the control plane mainly to initiate service requests and for supervision/maintenance of the control plane. NMI-T is the interface between the Management plane and the Transport plane.

It is important to recognize that there are two separate planes involved in the network. The Optical Transport Plane contains the transport network elements that carry the client signal. In general, these provide the capability to cross-connect optical channels. End-to-end connections of client

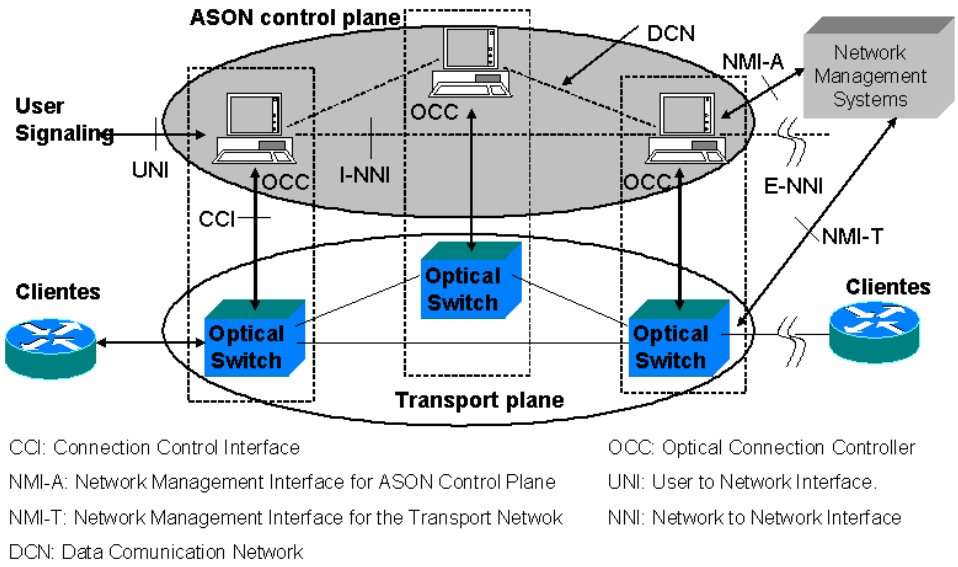

Figure 2.10: ASTN/ASON Architecture

signals are setup within the optical transport plane under the control of the ASTN/ASON Control Plane. The purpose of the optical control plane is to facilitate fast provisioning of connections within the transport network, to re-provision or modify connections that have previously been set up and to perform a restoration function.

Note that the above figure is a logical one; a physical realisation could probably encapsulate the control plane functionality (the OCC) inside the elements, in other words integrate the control plane with the transport plane. Early testbed realisations of the reference architecture [MPM+05] use "dumb" OXCs controlled via the CCI by the OCCs, which are implemented on router-PCs programmed with the needed functionality.

A comprehensive list of the the components of the control plane architecture are:

1. Connection Controller Function (CC): The connection controller is responsible for coordination among the Link Resource Manager, Routing Controller for the purpose of the management and supervision of connection setup, release and modification.

2. Routing Controller (RC): The role of the RC is to respond to requests from CC for route information needed to setup a connection, and to respond to requests for topology information for network management purposes.

3. Link Resource Management (LRM): The LRM component are responsible for the management of subnetwork links including the allocation and de-allocation of resources, providing topology and status information.

4. Traffic Policing (TP): The role of the TP is to check that the incoming user connection is sending traffic according to the parameters agreed upon.

5. Call Controller: There are two types of call controller, a calling/called party call controller and a network call controller. The role of the call control is the generation and processing of call requests.

6. Protocol Controller (PC): The PC provides the function mapping of the parameters of the abstract interfaces of the control components into messages that are carried by a protocol to support interconnection via an interface.

This brief description illustrate some relevant ideas of the ITU-T standardization work in respect to the OTN. The need of a Control Plane has been recognized, in order to achieve dynamic connectivity in the OTN. Different entities, interfaces and reference points are defined in the normative documentation; some of the signalling, addressing and routing ([ITU02a], [ITU04]) requirements of the ASON/ASTN can be fulfilled by IP-centric protocols, as being defined by the IETF GMPLS body of standards, which will be discussed next. The interested reader may refer to the cited ITU-T standard documents.

2.5 IETF Generalized MPLS model

The IETF conceive that future IP over optical networks, consisting of elements such as routers, switches, Dense Wavelength Division Multiplexing (DWDM) systems, Add-Drop Multiplexers (ADMs), optical cross-connects (OXCs), among others, will use Generalized Multi-Protocol Label Switching (GMPLS) to dynamically provision resources and to provide network survivability using protection and restoration techniques.

GMPLS extends the reference Multi-Protocol Label Switching (MPLS) architecture [RVC01], considering the peculiarities of non packet-based forwarding planes being considered: time-division (e.g., SONET/SDH, PDH, G.709), wavelength (lambdas), and spatial switching (e.g., incoming port or fiber to outgoing port or fiber). The focus of GMPLS is on the control plane of these various layers since each of them can use physically diverse data or forwarding planes, covering both the signaling and the routing part of that control plane. Some of the concepts introduced by GMPLS (i.e., link

bundling, unnumbered links, and LSP hierarchy) are backward applicable to the original MPLS architecture.

The original MPLS architecture is extended to include LSRs whose forwarding plane recognizes neither packet, nor cell boundaries, and therefore, they cannot forward data based on the information carried in either packet or cell headers. Taking into account these different types of interfaces, LSRs can be subdivided into the following classes:

1. Packet Switch Capable (PSC) interfaces:

   Interfaces that recognize packet boundaries and can forward data based on the content of the packet header. Examples include interfaces on routers that forward data based on the content of the IP header and interfaces on routers that switch data based on the content of the MPLS "shim" header.

2. Layer-2 Switch Capable (L2SC) interfaces:

   Interfaces that recognize frame/cell boundaries and can switch data based on the content of the frame/cell header. Examples include interfaces on Ethernet bridges that switch data based on the content of the MAC header and interfaces on ATM-LSRs that forward data based on the ATM VPI/VCI.

3. Time-Division Multiplex Capable (TDM) interfaces:

   Interfaces that switch data based on the data's time slot in a repeating cycle. An example of such an interface is that of a SONET/SDH Cross-Connect (XC) or Add- Drop Multiplexer (ADM).

4. Lambda Switch Capable (LSC) interfaces:

   Interfaces that switch data based on the wavelength on which the data is received. An example of such an interface is that of a Photonic Cross-Connect (PXC) or Optical Cross-Connect (OXC) that can operate at the level of an individual wavelength. Additional examples include PXC interfaces that can operate at the level of a group of wavelengths, i.e., a waveband and G.709 interfaces providing optical capabilities.

5. Fiber-Switch Capable (FSC) interfaces:

   Interfaces that switch data based on a position of the data in the (real world) physical spaces. An example of such an interface is that of a PXC or OXC that can operate at the level of a single or multiple fibers.

A circuit can be established only between, or through, interfaces of the same type. Depending on the particular technology being used for each interface, different circuit names can be used, e.g., SDH circuit, optical trail, lightpath, etc. In the context of GMPLS, all these circuits are referenced by a

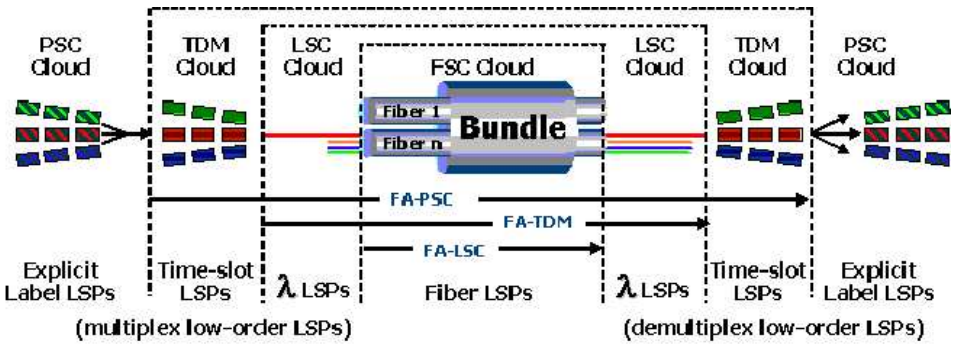

Figure 2.11: LSP hierarchy in GMPLS

common name: Label Switched Path (LSP). The concept of nested LSP (LSP within LSP), already available in the traditional MPLS, facilitates building a forwarding hierarchy, for example, a lower order SONET/SDH LSP (e.g., VT2/VC-12) nested in a higher order SONET/SDH LSP (e.g., STS-3c/VC-4) (or any of the several levels defined in the SONET/SDH multiplexing hierarchy).

Nesting can also occur between interface types. At the top of the hierarchy are FSC interfaces, followed by LSC interfaces, followed by TDM interfaces, followed by L2SC, and followed by PSC interfaces. This way, an LSP that starts and ends on a PSC interface can be nested in the hierarchy as shown in Figure 2.11.

Extensions to routing protocols and algorithms are needed to encode and carry TE link information, and explicit routes (e.g., source routes) are required in the signaling. In addition, the signaling must now be capable of transporting the required circuit (LSP) parameters such as the bandwidth, the type of signal, the desired protection and/or restoration, the position in a particular multiplex, etc. Most of these extensions have already been defined for PSC and L2SC traffic engineering with MPLS. GMPLS primarily defines additional extensions for TDM, LSC, and FSC traffic engineering. Thus, GMPLS extends the two signaling protocols defined for MPLS-TE signaling: RSVP-TE [ABG+01] and CR-LDP [JAC+02], and the intra-domain link-state routing protocols already extended for TE purposes: OSPF-TE [KKY03] and IS-IS-TE [SL04]. Extensions for inter-domain routing (e.g., BGP) are for further study.

Several GMPLS key extensions to "traditional" MPLS can be recognized:

- Link Bundling: the use of technologies like DWDM (Dense Wavelength Division Multiplexing) implies that we can now have a very large number of parallel links between two directly adjacent nodes (hundreds of

wavelengths, or even thousands of wavelengths if multiple fibers are used). The traditional IP routing model assumes the establishment of a routing adjacency over each link connecting two adjacent nodes. Having such a large number of adjacencies does not scale well. Each node needs to maintain each of its adjacencies one by one, and link state routing information must be flooded throughout the network. To solve this issue the concept of *link bundling* was introduced. Moreover, the manual configuration and control of these links, even if they are unnumbered, becomes impractical. The Link Management Protocol (LMP) was specified to solve these issues [Lan04].

- GMPLS extend the concept of heterogeneous label encoding by including links where the label is encoded as a time slot, or a wavelength, or a position in the physical space.
- In MPLS, an LSP that carries IP packets has to start and end on a router. GMPLS extends this by requiring an LSP to start and end on similar type of interfaces.
- The type of a payload that can be carried in GMPLS by an LSP is extended to allow such payloads as SONET/SDH, G.709, 1Gb or 10Gb Ethernet, etc.
- The use of Forwarding Adjacencies (FA) provides a mechanism that can improve bandwidth utilization. To improve scalability, it may be useful to aggregate multiple TE-LSPs inside a bigger TE-LSP. Intermediate nodes see the external LSP only, and, as they do not have to maintain forwarding states for each internal LSP, less signaling messages need to be exchanged. Protection is provided for the external LSP instead (or in addition) to the internal LSPs. This can considerably increase the scalability of the signaling. The aggregation is accomplished by: (a) an LSR creating a TE-LSP, (b) the LSR advertising this LSP as a Traffic Engineering (TE) link into IS-IS/OSPF, (c) allowing other LSRs to use forwarding adjacencies for their path computation, and (d) nesting of LSPs originated by other LSRs into that LSP (e.g., by using the label stack construct in the case of IP).

  ISIS/OSPF floods the information about "Forwarding Adjacencies" FAs just as it floods the information about other links. Consequently, an LSR has in its TE link state database the information about not just conventional links, but FAs as well, which can be used for path computation. FAs need simple extensions to signaling and routing protocols, and are advertised as GMPLS TE links such as defined in [KR02].
- GMPLS allows suggesting a label by an upstream node to reduce the setup latency. This suggestion may be overridden by a downstream

node but in some cases, at the cost of higher LSP setup time. GMPLS also extends on the notion of restricting the range of labels that may be selected by a downstream node. In GMPLS, an upstream node may restrict the labels for an LSP along either a single hop or the entire LSP path. This feature is useful in photonic networks where wavelength conversion may not be available.

- While traditional TE-LSPs are unidirectional, GMPLS supports the establishment of bi-directional LSPs.
- GMPLS supports the termination of an LSP on a specific egress port, i.e., the port selection at the destination side.
- GMPLS with RSVP-TE supports an RSVP specific mechanism for rapid failure notification.

Note also key differences between MPLS and GMPLS; for TDM, LSC and FSC interfaces, bandwidth allocation for an LSP can be performed *only in discrete units*, which have great impact when solving the allocation of resources for traffic demands (see [PM04] for a complete reference).

Back to the concept of link bundling, a typical example is an optical meshed network where adjacent optical cross-connects (LSRs) are connected by several hundreds of parallel wavelengths (DWDM). In this network, consider the application of link state routing protocols, like OSPF or IS-IS, with suitable extensions for resource discovery and dynamic route computation. Each wavelength must be advertised separately to be used, except if link bundling is used. It is possible to advertise several (or all) of these links as a single link into OSPF-TE and/or IS-IS-TE, in order to improve routing scalability by reducing the amount of information that has to be handled by these IGPs, accomplished by performing information aggregation/abstraction. Note that some information, i.e., it could be not possible for a CBR process to assign a particular wavelength to a light-path, due to the lack of information of the available lambdas in a given bundle. The choice of the component link to use (i.e., the wavelength) is always made by an upstream node. If the LSP is bi-directional, the upstream node chooses a component link in each direction.

Regarding the IP over Optical Service and Interconnection Models described in Section 2.3, in principle, GMPLS supports the three types of relationships, because it does not specify separately a UNI and a NNI. Edge nodes are connected to LSRs on the network side, which are connected between them. Nevertheless, GMPLS has been enhanced to support particularities at the UNI defined by the OIF [SDIR05].

2.5.1 Signalling Protocols

The GMPLS signaling extends and adds functionality to certain basic functions of the RSVP-TE and CR-LDP signaling protocols. These changes and additions impact basic LSP properties: how labels are requested and communicated, the unidirectional nature of LSPs, how errors are propagated, and information provided for synchronizing the ingress and egress nodes. The core GMPLS signaling specification comprises a signaling functional description [Ber03a], RSVP-TE extensions [Ber03b], and CR-LDP extensions [ASB03]. In addition, there are independent specifications per technology.

The following MPLS profile expressed in terms of MPLS features [RVC01] applies to GMPLS:

- Downstream-on-demand label allocation and distribution.
- Ingress initiated ordered control.
- Liberal (typical), or conservative (could) label retention mode.
- Request, traffic/data, or topology driven label allocation strategy.
- Explicit routing (typical), or hop-by-hop routing.

The GMPLS signaling defines the following new building blocks on the top of MPLS-TE:

1. A new generic label request format.
2. Labels for TDM, LSC and FSC interfaces, generically known as Generalized Label.
3. Waveband switching support (switching of contiguous groups of wavelengths).
4. Label suggestion by the upstream for optimization purposes (e.g., latency).
5. Label restriction by the upstream to support some optical constraints.
6. Bi-directional LSP establishment with contention resolution.
7. Rapid failure notification extensions.
8. Protection information currently focusing on link protection, plus primary and secondary LSP indication.
9. Explicit routing with explicit label control for a fine degree of control.
10. Specific traffic parameters per technology.

11. LSP administrative status handling.
12. Control channel separation.

Apart from the technological updates (i.e., the existence of new types of labels), the most interesting aspects of these building blocks are

- 4, 5: Label Suggestion and Label Restriction by the Upstream

  GMPLS allows for a label to be optionally suggested by an upstream node (which may be overridden by a downstream node). The suggested label is valuable when establishing LSPs through certain kinds of optical equipment where there may be a large (in electrical terms) delay in configuring the switching fabric.

  An upstream node can optionally restrict (limit) the choice of label of a downstream node to a set of acceptable labels. Giving lists and/or ranges of inclusive (acceptable) or exclusive (unacceptable) labels in a Label Set provides this restriction. If not applied, all labels from the valid label range may be used. There are at least four cases where a label restriction is useful in the "optical" domain.

  - Case 1: the end equipment is only capable of transmitting and receiving on a small specific set of wavelengths/wavebands.
  - Case 2: there is a sequence of interfaces, which cannot support wavelength conversion and require the same wavelength be used end-to-end over a sequence of hops, or even an entire path.
  - Case 3: it is desirable to limit the amount of wavelength conversion being performed to reduce the distortion on the optical signals.
  - Case 4: two ends of a link support different sets of wavelengths.

  The receiver of a Label Set must restrict its choice of labels to one that is in the Label Set. A Label Set may be present across multiple hops. In this case, each node generates its own outgoing Label Set, possibly based on the incoming Label Set and the node's hardware capabilities. This case is expected to be the norm for nodes with conversion incapable interfaces.

- 6: Bi-directional LSP

  GMPLS allows establishment of bi-directional symmetric LSPs. A symmetric bi-directional LSP has the same traffic engineering requirements including fate sharing, protection and restoration, LSRs, and resource requirements (e.g., latency and jitter) in each direction. Normally to establish a bi-directional LSP in traditional MPLS, two unidirectional paths must be independently established. This approach has the following disadvantages:

1. The latency to establish the bi-directional LSP is equal to one round trip signaling time plus one initiator-terminator signaling transit delay. This not only extends the setup latency for successful LSP establishment, but it extends the worst-case latency for discovering an unsuccessful LSP to as much as two times the initiator-terminator transit delay. These delays are particularly significant for LSPs that are established for restoration purposes.
2. The control overhead is twice that of a unidirectional LSP. This is because separate control messages (e.g., Path and Resv) must be generated for both segments of the bi-directional LSP.
3. Because the resources are established in separate segments, route selection is complicated. There is also additional potential race for conditions in assignment of resources, which decreases the overall probability of successfully establishing the bi- directional connection.
4. It is more difficult to provide a clean interface for SONET/SDH equipment that may rely on bi-directional hop-by-hop paths for protection switching. Note that existing SONET/SDH equipment transmits the control information in-band with the data.
5. Bi-directional optical LSPs (or light-paths) are seen as a requirement for many optical networking service providers.

With bi-directional LSPs both the downstream and upstream data paths are established using a single set of signaling messages. This reduces the setup latency to essentially one initiator-terminator round trip time plus processing time, and limits the control overhead to the same number of messages as a unidirectional LSP.

- 7: Rapid Notification of Failure

  GMPLS defines several signaling extensions for this purpose; in particular, extensions to RSVP-TE permit expedited notification of failures and other events to determined nodes. This first extension identifies where event notifications are to be sent, and the second provides for general expedited event notification with a Notify message. Such extensions can be used by fast restoration mechanisms.

  The Notify message is a generalized notification mechanism that differs from the currently defined error messages in that it can be "targeted" to a node other than the immediate upstream or downstream neighbor. The Notify message does not replace existing error messages.

- 12: Control Channel Separation

  In GMPLS, a control channel may be separated from the data channel. Indeed, the control channel can be implemented completely out-of-

band for various reason, e.g., when the data channel cannot carry in-band control information.

In traditional MPLS, there is an implicit one-to-one association of a control channel to a data channel. When such an association is present, no additional or special information is required to associate a particular LSP setup transaction with a particular data channel.

Otherwise, it is necessary to convey additional information in signaling to identify the particular data channel being controlled. GMPLS supports explicit data channel identification by providing interface identification information.

2.6 Conclusions

Next Generation Networks define a convergent universe of services supported by multiple integrated networks where the concepts of "Data Network" and "Telephone Network" are no longer valid. Distributed switched intelligence (i.e., the soft-switch/media gateway) is being incorporated in conjunction with the "media call" concept, supported in IP applications (payload) and IP-centric signalling and transport. Smart end-user devices such as 3G mobile phones, IP Set-top boxes, SIP-controlled home appliances and the like, together with ever increasing broadband "first-mile" to the home by means of xDSL, FTTH and wireless technologies demand new and smart/interactive services from the network. This challenge is being faced by traditional POTS/PSDN operators and Content Providers, with new revenue opportunities but increased operational complexities.

New Ethernet-based networks with multicast capabilities are being deployed in the aggregation layer in order to cope with the bandwidth demand to the residential users. New demands from the Access and Aggregation networks must be supported by new, enhanced Core networks. In this context, the IP over Optical concept is being deployed, based upon the two developing set of standards analyzed in previous sections: the ITU-T ASTN/ASON model and the IETF GMPLS model.

Both models share the same building blocks: giga/terabit routers with optical interfaces in the edge connected over all-Optical Transport Networks with a smart IP-centric Control Plane. While ASTN/ASON is basically defined to operate under the Overlay Model, with limited visibility inside the OTN by the client layers (such as IP/MPLS), across a well-defined UNI interface, GMPLS assumes a Peer Model with the corresponding visibility that enable client layers to discover the underlying OTN topology, with additional functionality of call setup and restoration. Nevertheless, lately GMPLS developments show a clear approach to the ASTN/ASON requirements. In this

	ASTN/ASON	GMPLS
Client/Server relationship	Multiple client/server relationship	Originally IP-centric, extended to multiple client/server (SDH/G.709)
Control Architecture	Overlay Model	Originally Peer Model, support for Overlay and Augmented models
Connection Management	Partly centralised	Fully distributed management
Connection type	Soft permanent connection / Switched connection	Switched connection

Table 2.1: ASTN/ASON - GMPLS comparison

regard, it is possible to envision an ASTN/ASON architecture implemented using GMPLS signalling and routing protocols. Since transport networks are built by ITU-T affiliated Telecom Operators and vendors, a first implementation of the IPO concept will be necessarily based in the Overlay Model, with a slow evolution towards the Augmented model as the BGP protocol evolve with TE capabilities, and the carriers get experience with the new IP-centric control plane. Table 2.1 summarizes ASTN/ASON-GMPLS relationship.

Disregarding the IPO model adopted, in terms of the path computation and connection establishment problems, two variants shall be considered:

- The overlay CBR problem, i.e., separate resolution of the allocation problem for each technological network (corresponding to the Overlay Model with Forwarding Adjacencies).
- The multilayer CBR problem, i.e., simultaneous resolution of the overall network (correspondent to the Peer Model).

The former is a well-known network design problem, and several solving techniques exist. The latter is a new problem that needs to be considered (see [PM04]) . Moreover, the allocation of bandwidth in discrete units in the transport technologies (notably the SDH hierarchy), adds complexity to the problem. The RMA proposal presented in Chapter 5 deploys solutions for the single-layer connectivity problem, thus regarding the multilayer, and specifically the provisioning in the OTN as future work.

Chapter 3

Traffic Engineering

3.1 Introduction

Challenging demands to the IP transport services in convergent NGNs include service differentiation for different applications, flexibility to easily adapt to new services, and tools to control the behavior of the network and make an efficient use of network resources. Evolved Traffic Engineering mechanisms are needed to complement the current IP transport functionalities in order to achieve those objectives. Enhancing the performance of an operational network, at both the traffic and the resource levels, are major objectives of TE, which is defined in [AMA+99] as *"... that aspect of Internet network engineering dealing with the issue of performance evaluation and performance optimization of operational IP networks..."*. These goals are accomplished by means of advanced routing strategies that enable the utilization of network resources efficiently and reliably.

Internet routing protocols are generally not well suited for traffic engineering of a network; shortest path route selection tends to provoke congestion on "shortest" network paths and under utilize other suitable paths, leading to a far less than optimal resource utilization. Several techniques like Equal Cost Multi Path (ECMP) and/or weight settings (in the OSPF routing protocol) have been devised to overcome these issues using traditional IGPs (a comprehensive proposal in this regard is presented in[SGD05]). As such, network engineers have relied on lower layer transport, particularly ATM networks, which builds virtual topologies that appear as physical links to the IP layer. ATM networks in particular provide for a rich set of constraint based routing, call admission control, traffic shaping and policing resources.

The problem of such overlay approach is that two different networks have to be built and managed, which increases complexity in network architecture and design. Reliability and scalability problems arise due to the amount and diversity of network elements that must be managed. The virtual topology is usually a near full-mesh of IP routers, leading to quadratic order of

adjacencies ($O(n^2)$) for the routing protocols, which not only causes overhead of routing messaging and processing CPU, but also may lead to serious scalability problems when a link goes down in the lower level network.

The evolution towards IP over Optical networks with integrated routing permits to envision a single traffic engineering framework based in MPLS. In this chapter general traffic engineering techniques will be reviewed.

3.2 Traffic Engineering Functions and Objectives

A practical function of the traffic engineering is the mapping of traffic onto the network infrastructure to achieve specific performance objectives. Network performance objectives such as QoS, efficiency, survivability are of major concern for Service Providers (which of course are interested in the economical impact of performance optimization). The network operator aims at providing QoS guarantees to its customers while making an efficient resource usage. These objectives as seen by the operator can then be summarized as follows:

- Traffic Oriented

  Related to the control of the QoS provided to the customer traffic. The user perceives relative (e.g. service differentiation, relative priorities and drop probabilities at packet or flow level) and absolute objectives (e.g. guaranteed throughput, end-to-end delay). These objectives are measured using traffic parameters that include packet loss, end-to-end packet delay, delay variation and throughput among others. The operator is mainly concerned with the fulfillment of Service Level Agreements (SLAs), with a global perspective of network performance.

- Resource Oriented

  Related to the efficient usage of network resources. These objectives are only visible to the operator, and determined mainly by the business model adopted. Efficient management of network resources is the vehicle for the attainment of resource oriented performance objectives. In particular, it is generally desirable to ensure that subsets of network resources do not become over utilized and congested while other subsets along alternate feasible paths remain underutilized. Nevertheless, is possible to meet traffic oriented objectives, even with poor efficiency on resource utilization; this is the usual policy in large backbones which tend to minimize traffic engineering efforts by over-provisioning the network.

A traffic engineering system is called *rational* if it is designed to attain the traffic oriented objectives, while optimizing the resource oriented objectives [Awd99].

The task of the traffic engineering system is to drive the network from sub-optimal states towards optimal states (in terms of both traffic engineering objectives) reflecting the business model and SLAs. In practice, sub-optimal network states are related to congestion situations produced whether by insufficient network resources for a given demand at a particular instant, or by inefficient mapping of the transported traffic over those resources. The TE system can deal with congestion situations produced by insufficient network resources by increasing the available capacity (planning and dimensioning) when the traffic injected to the network conforms the traffic contracts, or by policing the traffic at the ingress if that condition is not met.

Traffic policing enforces the ingress traffic to conform to the traffic contract, and is applied on a customer by customer basis on the ingress interface (e.g. traffic shaping, flow control, packet marking, etc.). Different TE mechanisms shall be applied according to the problem and the timescale. For instance, a congestion state produced by a systematic increase in traffic demands cannot be addressed using traffic policing, requiring a network re-dimensioning (or even a capacity reallocation process) applied in a longer term. On the other hand, a congestion state produced by a temporary increase in the traffic demands for a particular group of customers (e.g. non-conforming traffic) shall be addressed by means of short-term related TE mechanisms (e.g. traffic shaping) applied to the particular interfaces associated with those customers. Online traffic engineering tools based on advanced routing paradigms (i.e., Constraint-Based Routing) can adapt traffic mappings onto network resources to face varying traffic conditions. On a longer timescale, the base layout representing the actual traffic mapping to the network resources can be recalculated, resulting in a more efficient resource usage.

3.3 Traffic Engineering Process

Conceptually, the control of the network dynamics can be thought as a feedback control system, including a demand system (i.e. the traffic matrix), a constraints system (i.e. the interconnected network elements), and a response system (i.e. the network protocols and control mechanisms) [Awd99]. TE defines the parameters and points of operation for the network, as well as the mechanisms that control the deviations that occur when the demand system and/or the constraint system vary.

The TE process model, depicted in Figure 3.1, includes several stages. The first stage is the formulation of a Control Policy, or, in other words, to clearly state the business model and operational objectives. Next, a feedback mechanism is needed: the network relevant parameters shall be observed using a suitable set of monitoring functions, in order to characterize and analyze

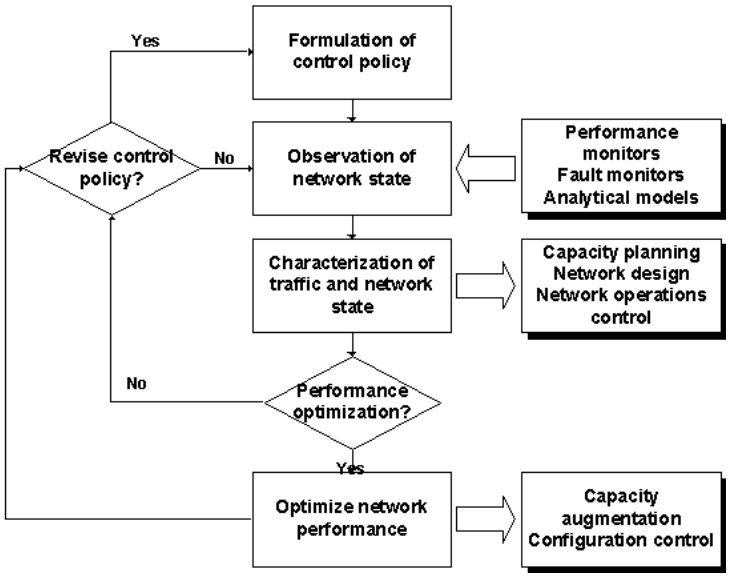

Figure 3.1: Traffic Engineering Process Model

the network state. This will make possible to identify the causes of network performance degradation, which is a basic input to network performance optimization, operations control, design and planning. The optimization of network performance may be accomplished by long term actions such as augmenting network capacity, and/or short term configuration actions (i.e. modify traffic control parameters).

This is an iterative control process, and it is desirable to automate most of the involved tasks in an operational environment. Next section analyzes the span of traffic engineering actions over different timescales.

3.3.1 Control Loops and Timescales

The control function of traffic engineering responds at multiple levels of temporal resolution to network events. Certain aspects of capacity management, such as capacity planning, respond at very coarse temporal levels, ranging from days to weeks or months. The introduction of optical transport networks with switching capabilities could significantly reduce the life-cycle for capacity planning by expediting provisioning of optical bandwidth. Routing control and packet level processing functions operate at finer levels of temporal resolution.

Long-term TE processes are often referred as network planing, meaning the dimensioning and building the physical network for long-term traffic growth. Corrective actions at this level comprises layout optimization and reconfiguration of the transport network, which may be dynamically achieved

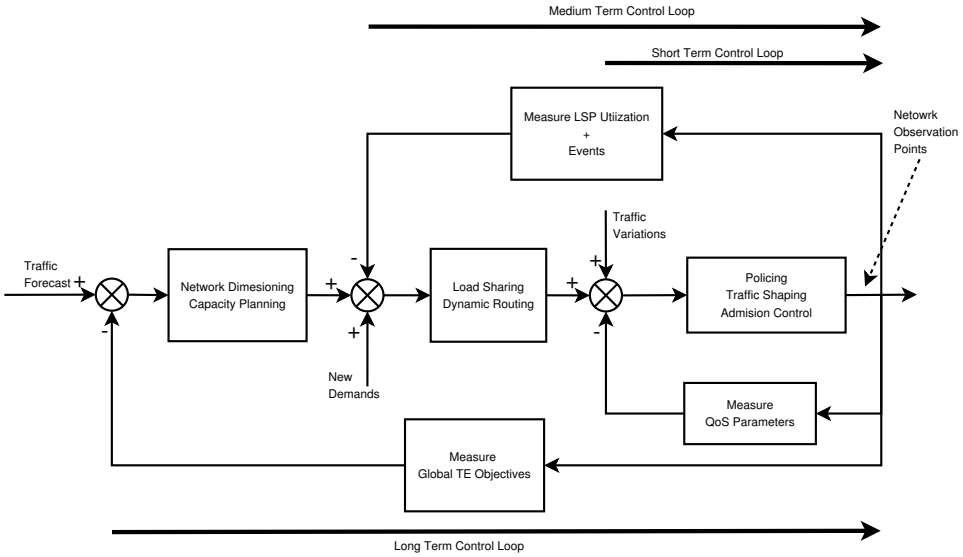

Figure 3.2: Traffic Engineering loops and timescales

in optical transport networks with switching capabilities, as mentioned before.

Medium-term TE processes perform the mapping of traffic demands onto network resources. In an MPLS network this means the computation and setup of LSPs to satisfy a given traffic matrix. Corrective actions at this level may include rerouting and load-sharing techniques. While the latter process aims to map traffic to available capacity (on a relatively static topology), the former aims to establish capacity where the traffic needs it, meaning a relaxation in bandwidth constraints.

Short-term TE involve actions at the flow and packet level, basically configuration of access control policies and shaping in routers interfaces.

Note that shorter term processes deal with local scope actions, while longer term processes scope is bigger, from LSP configuration to global network design. The aforementioned interdependencies among TE mechanisms and their correspondent timescales is modeled in [Bek04] as nested control loops. Nesting the control loops, actions taken in an outer loop generate a network state that is presented to inner loops as the observed state, so interdependencies are solved in the model. Figure 3.2 shows the different control loops and some of the actions identified with each one of them.

Once the network positioned in an optimal operational state, significant load variations will require a new network dimensioning (and eventually a new capacity allocation scheme), resulting in a network reconfiguration. Smaller load variations can be dealt with in an inner control loop correspond-

ing to a medium-term timescale. Techniques such as optimal load sharing in the case of MPLS as transport technology, or changing the metrics of the IGP routing protocols in a pure IP environment, can help to balance the load dynamically and allow for a better efficiency of the network resources. In a shorter timescale, and for traffic variations on a user-by-user flow level, the aforementioned admission control or traffic shaping, among others, can be used. The three interrelated control loops take actions based on measurement and system policies. Policy Based Network Management is a wide field of research, and will be considered in Chapter 4.

This thesis is related with the problems associated with dynamic routing and load sharing that arise with the arrival of new traffic demands (connectivity requests) on a medium-term timescale. Contributions on this field are detailed in Chapters 5 and 6.

3.3.2 Input Parameters for Traffic Engineering

Dynamic traffic engineering processes, as suggested by the nested control loops model, take as input a given topology and traffic demands and computes an optimal (or at least feasible) mapping of demands to resources. The process is frequently updated by the arrival of new demands, as a consequence of the feedback mechanisms, which involve network parameters measurement as a major component, and/or network policies.

3.3.2.1 Topology

Routing processes operate over a representation of the state of network resources, named topology database. This information is usually gathered from the link state advertisements (i.e. updates) of the routing protocol. Monitoring of relevant MIB variables and/or data from netflow applications may complement the information used as a base for path computation.

State information is inherently imprecise in a distributed network environment, and this directly affects the routing performance. Therefore, the design of routing algorithms for large networks should take the information imprecision into consideration. There are two main factors to consider: the number of entities generating updates, and the frequency at which each entity generates these updates.

Limiting the number of entities (nodes and links) that generate updates is a scalability issue, leading to the usage of hierarchical schemes by network protocols. OSPF supports a two-level hierarchy, with different granularity in terms of topology and routing information for each level. In particular, detailed information about topology and routes is available within an area, while only much coarser information is available about the backbone and remote areas. Other protocols, for instance ATM Private Network to Node

Interface (PNNI) [ATM02], generalizes this concept to an arbitrary number of levels, and defines a progressive aggregation procedure that combines multiple networks into "abstract nodes" representation. As a result, information about the state of individual nodes and links is often lost. This loss of accuracy in state information can have a substantial impact on the path-selection process. For example, the knowledge that a remote network, or set of networks, has an amount of available bandwidth, must be interpreted only as an indication that if a flow requests that amount, it is likely to be accepted. This is because the quantity is now only a summary of the amount of bandwidth actually available on the many different paths across this remote network.

Advertisement of a state change consumes both network bandwidth and node CPU cycles; therefore is desirable to keep this overhead to a minimum. The different methods that can be used to achieve such a goal typically involve waiting for either a change threshold or until a timeout has expired. As a result, the actual state of a remote node or link can drift away from the value known to other nodes without these being aware of it. The size of the gap between the actual state and its last advertised value clearly depends on the specifics of the mechanisms used to control distribution of state information.

In [GO99] it is shown that the impact of inaccuracies is relatively minimal in the case of flows with only bandwidth requirements, but had a major impact on the complexity of the path-selection process for flows with end-to-end delay guarantees, which typically became intractable. Further considerations and proposed solutions have been explored in [AGKT99] (e.g., based in triggering policies), [MB03], among others.

3.3.2.2 Traffic Demands

Traffic engineering processes such as network planning and routing need a network-wide view of traffic. Three possible models to represent the traffic volumes that travel across the network are depicted in Figure 3.3. The *path matrix* shown in (a) defines the data volumes for every path between every source and destination node. Typically, the path matrix is a result of the the demand allocation process. In an MPLS network, the paths between node pairs can be defined explicitly and the path matrix can be derived by measurements. This method is suggested in [XHBN00] to determine the actual bandwidth for each LSP. The path matrix is a fine-grain approach that gathers current network status, and is used in real-time operational situations. The *traffic matrix* shown in (b) represents offered load between each ingress and egress pair. This model is widely used in academic and operational contexts, targeted to intra-domain traffic engineering. Finally, the third model shown in (c) is the *demand matrix*, that describes the traffic volumes between an ingress node and the set of egress nodes. This approach

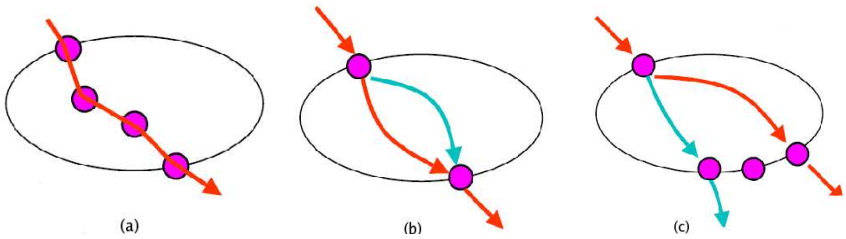

Figure 3.3: Traffic models

is used for inter-domain routing, more precisely, to choose the set of egress nodes for a given demand.

The traffic matrix has operational applications both in the medium and long term traffic engineering processes. When facing congestion and performance problems, it may be used to predict link loads after a routing change, and may also be used in long term processes, e.g., predicting link loads after a change in network topology. A basic issue is how the traffic matrix is constructed in operational IP networks. The fact is that information is not directly available, leading to a number of estimation methods based on monitoring.

SNMP monitoring of network nodes give the link loads, but this represent the aggregate of several network flows, which is different from the origin-destination (OD) traffic matrix. Each link transport a fraction of the load of different OD pairs; a possible way to discern each OD pair flow in a link could be to measure individual flows (e.g., using netflow techniques), which is impracticable in operational networks.

The problem of estimating the OD byte counts from aggregated measurements on network links is called *network tomography* [Var96]. The similarity to conventional tomography lies in the fact that the observed link counts are linear transforms of unobserved OD counts with a known transform matrix determined by the routing scheme.

In path-level traffic intensity estimation, the measurements consist of counts of packets that pass through nodes in the network. Based on these measurements, the goal is to estimate how much traffic originated from a specified node and was destined for a specified receiver. The combination of the traffic intensities of all these origin-destination pairs forms the origin-destination traffic matrix.

Let $y = (y_1, ..., y_J)'$ denote the *observed* column vector of incoming and outgoing byte counts measured on each router link interface during a given interval, and let $x = (x_1, ..., x_I)'$ denote the *unobserved* vector of corresponding byte counts for all OD pairs in the network. One element of x, for example, corresponds to the number of bytes originating from a

specified origin node to a specified destination node, whereas one element of y corresponds to bytes originating from the origin node regardless of their destination. Thus, each element of y is a sum of selected elements of x, so

$$y = Ax$$

where A is a $J \times I$ routing matrix of 0's and 1's that is determined by the routing scheme of the network (fixed routing is considered). Since for n nodes the number of links J is $O(n)$, and the number of I OD pairs is $O(n^2)$, then $I \gg J$, and the dimension of the solution sub-space is at least $I - J$, which means that the linear system is under determined.

There are at least three techniques to cope with this problem: Linear Programming(LP), Expectation Maximization (EM) and Bayesian estimation. A complete reference on these methods and in network tomography in general is given in [CHINB02].

Note that any traffic matrix is based on measurements performed during a specific time interval, whose granularity should be related to the purpose for which the traffic matrix is to be used. For planning purposes a time series of the traffic of interest is typically built with each point on the curve being representative of the traffic over an extended period (from a year in a stable scenarios to a week in a very rapidly changing scenario). If we want to examine the traffic behaviour during an extended time span (e.g over several days or weeks, or even months) we will typically have several traffic matrices pertaining to short time intervals collected within the extended time period of interest, i.e. a set of snapshots, none of which is by itself a proper characterization of the traffic during the whole period. An unavoidable task is therefore the establishment of criteria for deriving a single traffic matrix, representative of the traffic during the extended time period, from the collected snapshots, or, in other terms, the aggregation of the single traffic matrices over time. This problem has been dealt by the ITU for the telephone network; however, it has been approached again in the scenario of data networks.

3.3.3 Traffic Characteristics and Measurement

The operational state of a network can be conclusively determined only through measurement, and is also a critical input for the optimization function because it provides feedback data that is used by traffic engineering control subsystems. This data is used to adaptively optimize network performance in response to events originating within and outside the network. Measurement is also needed to determine the quality of network services and to evaluate the effectiveness of traffic engineering policies as perceived by users, i.e., *emergent properties* of the network.

The problem of capacity allocation in traditional telecommunications networks is characterized by some key features: homogeneous users with inelastic demands, given that voice traffic aggregates in 64 kbps steps with each phone call, and once a link is saturated no more callers are admitted to the network. Moreover, a stationary demand of calls is verified (a Poisson process). The natural allocation policy for telephony is to assign resources at call admission time; the intelligence is in the network, and end-systems are dumb. Network is provisioned to accommodate the known demand, allowing a moderate pace evolution. Therefore, the analytical framework to study these problems is queueing theory.

On the other hand, the Internet is a user-driven network with dramatic growth rate, and carries two great categories of flow types: elastic and stream flows. Stream flows are representative of real-time applications (e.g. audio or video UDP/RTP flows); they have intrinsic temporal properties that the network must preserve. This flows do not respond to congestion, or recover from packet loss. The elastic flows (TCP flows) are representative of file transfer-like applications. For elastic flows the transfer throughput depends on the available bandwidth: the transfer duration is a function of the file size and traffic characteristics of the transport network during the transfer. Elastic traffic is the most important of the Internet in terms of its proportion to stream traffic (approximately 80% of the flows), and can be classified in Short TCP flows ("mice"), which retransmit lost packets, but are too short to adapt their rate to the network capacity, and long TCP flows ("elephants"), which retransmit lost packets, and carry the burden of controlling bandwidth, avoiding congestion. In a best-effort traffic model, network can survive as long as elephants predominate (and behave well). It worth noting that the growth on available bandwidth (both in the core and residential networks), improvement on users' hardware, and the rapid growth in usage of peer-to-peer file sharing systems, have significantly changed the traffic mix, with ever increasing size of transmitted files.

The Internet has evolved in the understanding of traffic characterizations, thanks in part to the developments at CAIDA [CAI05] and the IETF Internet Protocol Performance Metrics Working Group [IET]. These measurement frameworks have been used by several studies that conclude that Internet traffic is fractal or self-similar in nature. Self similar means that the traffic exhibits the same characteristics regardless of the number of simultaneous sessions on a given physical link, i.e., traffic variability is invariant to the observed time scale, and do not become smooth with aggregation as fast as the Poisson traffic model would indicate (see Figure 3.4). Self-similarity in Internet traffic is mainly attributed to heavy-tailed distributions of file sizes. In addition, large scale correlations characterize wide-area traffic traces, concluding that the Poisson model should be abandoned for all but user session arrivals, because it underestimate both burstiness and variability. The na-

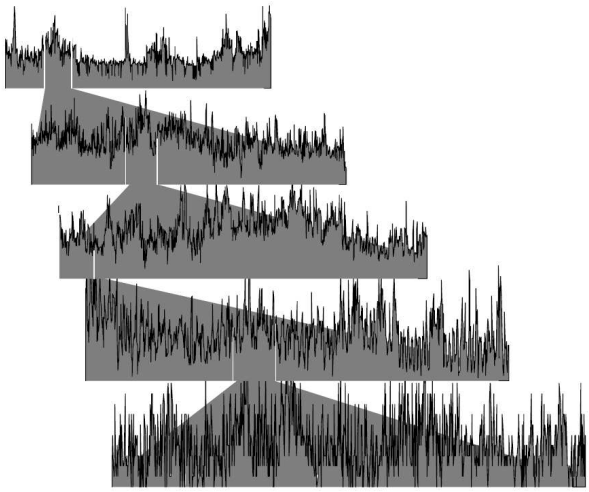

Figure 3.4: Self-similar Internet traffic

ture of the congestion produced from self-similar network traffic models had a considerable impact on queuing performance, due in large part to variability across various time scales. Further studies proved that Poisson-based models significantly underestimated performance measures, showing that self-similarity resulted in performance degradation by drastically increasing queuing delay and packet loss. Despite the overwhelming evidence of Long Range Dependence (LRD) presence in Internet traffic, a few findings indicate that Poisson models and independence could still be applicable as the number of sources increases in fast backbone links that carry vast numbers of distinct flows, leading to large volumes of traffic multiplexing [KMF04]. To minimize congestion IP networks must operate at a higher average peak to average load than in a traditional telecommunications network. A possible solution for network operators is to increase the buffer size at admission points into the network to smooth out the bumps and valleys. However, large buffering results in throughput delays of the data.

The adopted traffic model has impact in network dimensioning, when it is considered to model certain objective function or constraint (e.g., if end-to-end delay is bounded, a mathematical expression using the corresponding statistical assumption shall be considered).

3.3.4 Network Dimensioning

Classical allocation problems can be classified in *uncapacitated*, where both path and link capacities are computed to satisfy a given demand (also called a *dimensioning* problem since link capacity is to be fixed), and *capacitated*, where the networks path are computed for a given network (i.e., link capac-

ities are known in advanced). A mixed capacitated/uncapacitated problem arises when the objective is to dimension the link capacities within certain upper bounds. The problem of allocating network resources to traffic demands is a multi-commodity allocation problem, because multiple demands (commodities) must be fitted into network pipes (links).

This set of problems are usually solved using a load balancing techniques (i.e. the traffic demand is splitted in more than one network path). The offered traffic of each ingress-egress pair is carried in the network in such a way that some objective function is minimized. Depending on the application, the paths used by each OD-pair can be arbitrary or selected beforehand. As an example the optimization problem with unconstrained paths is presented next.

Let us consider a single domain network, which consists of nodes and links connecting them. Let $\aleph$ denote the set of nodes n and $\mathcal{L}$ the set of links l of the network. Alternatively we use notation (i, j) for a link from node i to node j. The capacity of link l is denoted by b_l. The set of origin-destination (OD) pairs $k = (s_k, t_k)$ is denoted by K with s_k referring to the ingress node and t_k referring to the egress node of OD-pair k. The traffic demand, that is, the mean rate of offered traffic between nodes s_k and t_k, is denoted by d_k.

Furthermore, $A \in \mathbb{R}^{\mathbb{N}\times\mathbb{L}}$, where $N = \mid \aleph \mid$and $L = \mid \mathcal{L} \mid$, denotes the link-node incidence matrix for which $A_{nl} = -1$ if link l directs to node n , $A_{nl} = 1$ if link l leaves from node n, and $A_{nl} = 0$ otherwise; $x^k \in \mathbb{R}^{\mathrm{Lx1}}$, $k \in K$, refers to the link load vector with elements x_l^k; and $R^k \in \mathbb{R}^{\mathrm{Nx1}}$, $k \in K$, denotes the vector for which $R_{s_k}^k = d_k$, $R_{t_k}^k = -d_k$, and $R_n^k = 0$ otherwise.

The rate of traffic allocated by OD-pair k on link l is denoted by x_l^k. These rates are the control variables in the static load balancing problem. Given the x_l^k, the induced load y_l on link l is

$$y_l = \Sigma_{k\in K} x_l^k; \text{ for all } l \in \mathcal{L}.$$

It can be seen that, in general, the mapping between the traffic demands d_k and the link loads y_l is not one-to-one, i.e. while d_k's determine y_l's uniquely, the opposite is not true. Load y_l on link l incurs cost $C_l(y_l)$, which is assumed to be an increasing and convex function of the load y_l. The objective in the *unconstrained static load balancing* problem is to minimize the total cost by choosing an optimal traffic allocation.
The problem is formulated as follows:

Minimize $C(x) = \Sigma_{l\in\mathcal{L}} C_l(y_l)$

subject to the constraints

$$\begin{array}{ll} x_l^k \geq 0, & \text{for each } l \in \mathcal{L} \text{ and } k \in K \\ \Sigma_{k\in K} x_l^k \leq b_l, & \text{for each } l \in \mathcal{L}, \\ Ax^k = R^k, & \text{for each } k \in K \end{array}$$

In this problem there are three constraints, the first one states that link loads should be positive, the second one is the capacity constraint and the third one is so called *conservation of flow constraint*, which states that the traffic of each OD-pair incoming to a node has to be equal to the outgoing traffic from that node. As a result, the load balancing gives the linkwise portions x_l^k of the original demands $d_k$but the routes for these demands through the network are not necessarily unique. However, the routes are guaranteed to be loop-free.

This is just an example of the network design problems. A complete reference can be found in [PM04]. These problems can be solved in terms of various objective functions. The functions can be categorized to linear and non-linear, resulting in linear optimization programs (LP) and non-linear optimization programs (NLP), respectively. A simple linear optimization problem is the *minimum cost flow problem.* In that problem, each link l has unit cost a_l.
The linkwise cost is then

$$C_l(y_l) = a_l y_l \text{ for all } l \in \mathcal{L}$$

If the cost weights are selected to be $a_l = 1$ for all l, the optimization problem minimizes the amount of used resources. On the other hand, if only one path is allowed to each OD-pair, the problem is the same as the so called *shortest path problem.*

These are the kind of problems faced by the network dimensioning function for long and medium term traffic engineering control loops. The objective of network dimensioning is to fix an optimal state for network operation. Under dynamic conditions the network will evolve to states aside from the optimal, and the function of the traffic engineering techniques is to minimize the deviation.

3.3.5 Single and Multilayer Traffic Engineering

In Chapter 2 it has been stated that contemporary data networks typically follow a two-tiered approach, comprising of a logical (IP layer) topology that is mapped onto the actual physical optical topology by means of light-paths, as shown in Figure 3.5. Hence traffic engineering can take place along two dimensions. The first is the routing of light-paths on the physical network to construct the appropriate logical topology. The second dimension where traffic engineering can take place is configuration of the IP routing state to control the flow of traffic over the logical topology. While traffic engineering the physical network to configure transport light-paths is an important aspect of network operation, this process typically takes place over a scale of months or even years, even though great attention is posed in

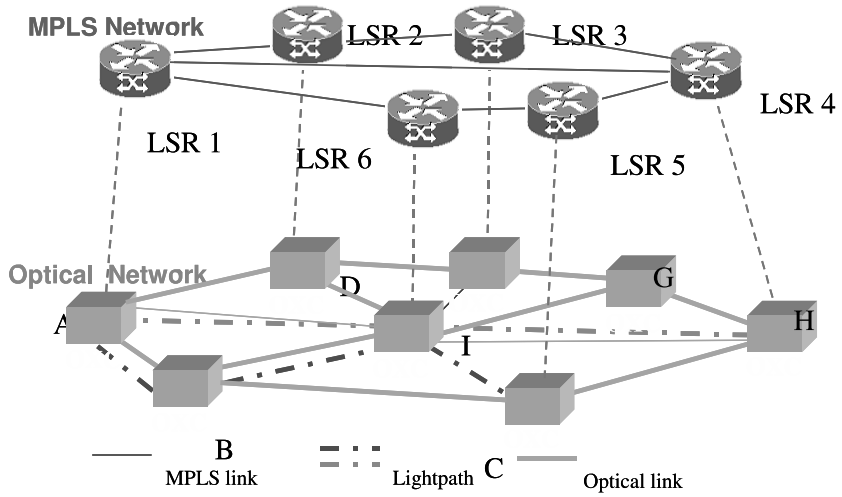

Figure 3.5: Multilayer reference scenario

the provision of switched connections in the optical transport network. On the other hand, traffic engineering on the logical topology is a more dynamic operation which takes place on a day-to-day basis and hence has a more immediate effect on network performance. Furthermore, proper grooming of traffic flows on the logical topology can help defer costly capacity and network upgrades of the physical network as well as avoid reconfiguration of light-paths which can be a time consuming process. This thesis is focused on issues related to the latter aspect, namely traffic engineering over the logical topology.

3.4 MPLS Traffic Engineering

One of the most significant application of MPLS is augmenting native IP networks with traffic engineering capabilities. The requirements of traffic engineering over MPLS are described in [AMA+99]. In addition to explicit routing (i.e., not constrained by the destination based forwarding paradigm), MPLS provides mechanisms to route traffic between two edge routers along several paths, which offers several advantages. For instance, traffic can be routed successfully in the case of link failures using alternative paths. However, the most important benefit of traffic splitting is the ability to balance the load. Load balancing reduces congestion and therefore improves the performance of the network.

The basic element of traffic engineering over MPLS is the *traffic trunk,* that consists of the traffic that belongs to the same class and is routed along the same path. A traffic trunk is an abstract representation of traffic to

which specific characteristics can be associated, i.e., is a routable object, that is, the path through which a traffic trunk traverses (the actual Label Switched Path) can be changed. Network operators can define the attributes of traffic trunks, which will be routed by LSPs computed using Constraint-Based Routing (CBR).

Additionally, through explicit label switched paths, MPLS permits a quasi circuit switching capability to be superimposed on the current Internet routing model.

There are three basic problems in providing traffic engineering over MPLS:

- The first problem is how to map packets to FECs, which involves the definition of rules to group packets using certain common characteristics, i.e., the Virtual Routing Forwarding (VRF) table , in the context of IP-VPN applications [RR05]. This mapping is done at the Ingress LSR.
- The second problem is how to map FECs to particular traffic trunks, which involve basic scheduling functionality in the Ingress LSR. If traffic is splitted in several traffic trunks (e.g., using Classes of Services as splitting criteria), a dispatching function shall assign traffic to outgoing router queues to send traffic over the desired traffic trunk.
- The third problem concerns how to map traffic trunks onto the physical network through LSPs, which is the result of solving the load balancing routing problem.

The first two problems are faced by the hardware design process, i.e., these functionalities are built in the router. The third problem, which is basically the network path computation and setup process is analyzed in this section.

3.4.1 MPLS Traffic Trunks

A traffic trunk is an *unidirectional* entity, an *aggregate* of traffic flows belonging to the same class. In some contexts, it may include multi-class traffic aggregates. In a single class service model, such as the current Internet, a traffic trunk could encapsulate all of the traffic between an ingress LSR and an egress LSR. A traffic trunk is a routable object, and is distinct from the LSP through which it traverses. In operational contexts, a traffic trunk can be moved from one path (LSP) onto another.

In practice, a traffic trunk can be characterized by its ingress and egress LSRs, the forwarding equivalence class which is mapped onto it, and a set of attributes which determine its behavioral characteristics.

Although traffic trunks are conceptually unidirectional, it is useful in operational contexts to simultaneously instantiate two traffic trunks with the same endpoints, but which carry packets in opposite directions. The two

traffic trunks (named *forward* and *backward* trunks), are logically coupled together, and referred as one bidirectional traffic trunk (BTT), which must be instantiated and destroyed together. This condition is a requirement for management applications that setup such BTTs in the network. BTTs life-cycle resembles the Sub Network Connection (SNC) (see [ITU00]) definitions for transport networks:

- Establish: create an instance of a traffic trunk.
- Activate: assign actual traffic to the trunk (start). The establishment and activation of a traffic trunk are logically separate events, but may be implemented as one atomic action.
- Deactivate: de-assign traffic of the trunk (stop).
- Modify Attributes.
- Reroute: change traffic trunk route (by management or automatically by the underlying protocols).
- Destroy: remove an instance of a traffic trunk from the network and de-attach all resources allocated to it, such as label space and bandwidth.

The traffic trunk traffic engineering attributes are parameters assigned to it which influences its behavioral characteristics. These attributes can be explicitly (administratively) assigned, or they can be implicitly assigned by the classification and mapping traffic functions into FECs at the ingress to an MPLS domain. The basic attributes of traffic trunks are their *traffic parameters*, the *generic path selection and maintenance* mechanisms, *priority*, *preemption*, *resilience* and *policing* attributes.

The combination of traffic parameters and policing attributes enable the surveillance of a traffic contract (as in ATM networks). Priority and preemption can be regarded as relational attributes because they express certain binary relations between traffic trunks. Conceptually, these binary relations determine the manner in which traffic trunks interact each other as they compete for network resources during path establishment and path maintenance. For example, a traffic trunk "A" can preempt another traffic trunk "B", only if *all* of the following five conditions hold: (i) "A" has a relatively higher priority than "B", (ii) "A" contends for a resource utilized by "B", (iii) the resource cannot concurrently accommodate "A" and "B" based on certain decision criteria, (iv) "A" is preemptor enabled, and (v) "B" is preemptable.

Resource class affinity attributes can be used to specify the class of resources which are to be explicitly included or excluded from the path of the traffic trunk. These are policy attributes which can be used to impose additional constraints on the selection path process. Resource class affinity attributes for a traffic can be specified as a sequence of tuples:

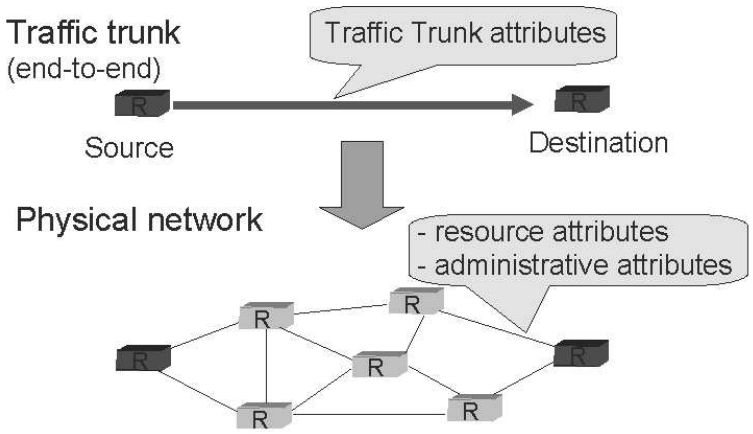

Figure 3.6: Traffic Trunk, Resource and Administrative attributes

<resource-class, affinity>; <resource-class, affinity>; ..

The resource-class parameter identifies a resource class for which an affinity relationship is defined, while the affinity parameter indicates the affinity relationship; that is, whether members of the resource class are to be included or excluded from the path of the traffic trunk. Resource class affinity attributes are very useful and powerful constructs because they can be used to implement a variety of policies. For example, they can be used to contain certain traffic trunks within specific topological regions of the network, more specifically, in a Differentiated Services environment, different regions of the network could be assigned in advanced for different Classes of Service, each region with it distinct operational policies. The significance of the traffic trunk attributes in relationship with the resource and administrative attributes are depicted in Figure 3.6.

Regarding the load balancing technique analyzed above, the aggregate traffic between two nodes may be splitted among multiple traffic trunks between the two nodes, such that each traffic trunk carries a proportion of the aggregate traffic. Note that is desirable to maintain packet ordering between packets belong to the same micro- flow (same source address, destination address, and port number).

3.4.2 Constraint Based Routing

The main purpose of a routing algorithm is to find the best suitable path to carry the data towards the destination. Though, the definition of the "best path" contains all the complexity of the decision. Particularly, the decision is usually constrained by a combination of some QoS and administrative requirements. It is usual to express QoS requirements in terms of bandwidth

and delay, obtaining a solution space for the constrained problem (i.e. non-unique solution). Hence, an objective function is added to the problem in order to find the "best constrained path". Typical objectives are fair load balancing over the network links, reducing the rejection probability for LSP demands, or minimizing the resource utilization cost.

On the other hand, routing algorithm efficiency is revealed by typical performance parameters: an efficient use of network resources is translated into a low *blocking probability* for new demands or a good load balancing that alleviates congestion bottlenecks. Moreover, survivability issues can be an additional evaluation criterion: algorithms are compared for their reaction to failures in the network.

The problem of finding an optimal path meeting a set of constraints is referenced as Constraint-Based Routing, and constitutes an important area of research. Constraint-Based Routing is a generalization of QoS routing (an evolution from traditional IGPs), that takes specified traffic attributes, network and policy constraints into account when making routing decisions. Constraint-based routing is applicable to traffic aggregates as well as flows.

Constraints are broadly classified in three types: concave (e.g. bandwidth or resource class), additive (e.g. delay) and multiplicative (e.g. loss probability). A cost function using additive constraints has the form

$$\Sigma(d_1, d_2, ..., d_n),$$

while a cost function using multiplicative constraints looks like

$$\Pi(d_1, d_2, ..., d_n)$$

In the case of concave constraints the path selection chooses links whose characteristics are always above a given constraint, e.g., bandwidth. A simple heuristic using the bandwidth constraint to prune the network graph, and applying a shortest-path algorithm to the resultant sub-problem is considered in section 3.5. While single additive or multiplicative constraints are tractable by, for example, the well-known Dijkstra [Dij59] or Bellman-Ford [FF62][Bel58] algorithms, the problem becomes NP-complete when two or more additive/multiplicative constraints are used. In this case several possible solutions must be evaluated, using heuristics to choose a feasible one.

Constraint-Based Routing is usually qualified as offline and online.

- *Offline CBR* performs path computation outside network elements, in a Path Computation Server (PCS). It takes as input a known static traffic matrix and, based on a detailed and accurate topology map (built with information gathered from the network), it computes the optimal network paths for that given traffic matrix. The drawback

of such approach is that a detailed traffic matrix has to be known in advance. The solution is valid for the given static input, but it cannot satisfy new traffic demands.

- *Online CBR* is a routing mechanism embedded on network elements intelligence. Such a routing process receives, as input, dynamic traffic requests and has no knowledge of future requirements. Given this traffic demand and based on a dynamic (and possibly incomplete) network state it computes feasible paths for that demand. The drawback of such approach is that it has to be performed under strict operational requirements (e.g., computational complexity, algorithm convergence time) and has to be resilient to transient network conditions.

Online approaches find a suitable path upon demand arrivals. LSPs can be calculated to meet demand requirements and service constraints, such as available capacity and experienced delay, as well as to optimize a given function of network economics. Calculated paths will be established using CR-LDP or RSVP-TE . In the on-line path calculation approach, the resulting global traffic distribution over time depends on the arrival order and size of demands. Actual resource allocation over time could become far from optimal with respect to some performance criteria (e.g. maximum link load or total delay) or economic criteria (e.g. quantity of paths or links) used to design the layout.

Offline LSP layout calculation allows for the setting up of a point of operation, globally optimal with respect to some performance criteria related to the network cost of operation. *Offline* calculation takes into account global information about the state of the network and traffic forecast, while on-line calculations are only allowed to take into account local and incomplete information. In fact, on-line and off-line calculations are complementary. The network can be optimally engineered and set up around a point of operation on a long term basis, and on-line decisions can be taken to accommodate traffic variations around that point on a shorter timescale.

Constraint-Based Routing Algorithms

Routing involves two entities, namely the routing protocol and the routing algorithm. The routing protocol has the task of capturing the state of the network and its available network resources and distributing this information throughout the network. The routing algorithm uses this information to compute shortest paths. Current best-effort routing performs these tasks based on a single measure like hop-count or delay. CBR however, must take into account multiple QoS and administrative requirements, the so-called *multi-constraint-path* problem (MCP), which unfortunately, is known to be a NP-complete problem. An extensive review of CBR algorithms can be found in [FTMP02]. A brief list of existing CBR algorithms is given bellow:

- Widest-Shortest Path
- Shortest-Widest Path
- Dynamic Alternative Path
- Dynamic routing with partial information
- Minimum interference routing algorithm
- Profile-based routing

The standard *shortest path problem* can be solved by the aforementioned Dijkstra and Bellman-Ford algorithms, which compute shortest paths from a given source node to all other nodes in the network. A simple heuristic (suggested by the normative document [AMA+99]), namely the Constrained Shortest Path First (CSPF) algorithm is given in section 3.5.

3.4.3 Dynamic Load Balancing

The concept of *induced MPLS graph* is important for traffic engineering in MPLS; it is analogous to a virtual topology in an overlay model. It is logically mapped onto the physical network through the selection of LSPs for traffic trunks. An induced MPLS graph consists of a set of LSRs (the nodes of the graph) and a set of LSPs (which provide logical point to point connectivity between the LSRs, and hence serve as the links of the induced graph). The induced MPLS graph abstraction is formalized below.

Let $G = (V, E, c)$ be a capacitated graph depicting the physical topology of the network. Here, V is the set of nodes in the network and E is the set of links; that is, for v and w in V, the object (v, w) is in E if v and w are directly connected under G. The parameter c is a set of capacity and other constraints associated with E and V. We will refer to G as the "base" network topology.

Let $H = (U, F, d)$ be the induced MPLS graph, where U is a subset of V representing the set of LSRs in the network, or more precisely the set of LSRs that are the endpoints of at least one LSP. Here, F is the set of LSPs, so that for x and y in U, the object (x, y) is in F if there is an LSP with x and y as endpoints. The parameter d is the set of demands and restrictions associated with F. Evidently, H is a directed graph. It can be seen that H depends on the transitivity characteristics of G.

Thus, the induced MPLS graph is the actual solution of the problem of allocating network resources to a system of demands under certain constraints and objective function. Computed offline, it is a solution of the static load balancing problem. When traffic conditions change unexpectedly, adaptive methods are needed. The known approaches seek for distribution of the load

in a balanced way based on measurements, such as end-to-end monitoring, or monitoring of each link individually.

MPLS Adaptive Traffic Engineering (MATE) [EJLW01] is a distributed adaptive algorithm for balancing the load. The algorithm tries to equalize congestion measures among the LSPs by approximating the gradient-projection algorithm, which transfers traffic toward the direction of the gradient projection of the objective function. The concept of splitting traffic at the flow level into multiple paths is introduced in [SKL+03]. The LSP for the incoming traffic is selected from the fixed set of paths based on congestion and the length of the path. The MATE approach bases traffic engineering decisions on the measured traffic conditions between each ingress and egress pair. However, these measures offer overlapping information since LSPs might use same links. The second algorithm efficiency depends largely on the flow-level dynamics, and it remains unclear whether the granularity of the traffic splitting at the flow level is fine enough to provide stable network conditions. In both proposals the ingress nodes play a key role classifying and scheduling traffic in the chosen LSPs. An operational implementation of these strategies should take into account the extra CPU load on these nodes.

3.4.4 MPLS Support for Differentiated Services

Differentiated Services (DiffServ) enables scalable network design with multiple Classes of Service. MPLS traffic engineering enables resource reservation, fault-tolerance, and optimization of network resources. MPLS/DiffServ together with the routing protocol combines the advantages of both. The result is the ability to give strict Quality of Service (QoS) guarantees and use the network resources in an optimal way. The MPLS and DiffServ approaches share some key ideas, like pushing the complexity to the edges of the network, preventing core routers to handle a huge amount of flows. MPLS and DiffServ are complementary; DiffServ defines the different behaviours in a router while MPLS together with the routing protocols determines the path between different nodes.

DiffServ is a scalable and operationally simple solution because it does not require per-flow signalling. However, it cannot guarantee QoS, because it does not influence a packet path. In case of congestion or failure even high-priority packets will be droped in a DiffServ only environment. MPLS explicit and constraint-based routing capabilities can provide the lacking bandwidth guarantees (the so-called *DS-TE model*). The Per Hop Behaviour (PHB) is mapped into the MPLS header using either the E-LSP mode, or the L-LSP mode. These (and other) mechanisms to support DiffServ in a MPLS environment are described in [FWD+02].

A possible way to combine DiffServ and MPLS is to perform traffic engineering of different classes of service separately so that each class of Service

is transported on different MPLS tunnels, as mentioned in section 3.4.1. One reason of doing so is to apply different fast restoration policies to the different classes of service. Another reason might be the use of separate Constraint Based Routing in order to meet the different QoS objectives of each Class of Service. In the DS-TE mode, the CoS-base bandwidth guarantee is achieved by two network functions: separate bandwidth reservations for different set of traffic classes admission-control procedures applied on a per-class basis.

To describe these two functions, the DS-TE model introduces two new concepts:

- Class-Type (CT) is a grouping of Traffic Trunks (TT) based on their CoS values so that they share the same bandwidth reservation, and where a single CT can represent one of more classes; and
- Bandwidth Constraint (BC) is a limit on the percentage of a link bandwidth that a particular CT or a group of CTs may take up.

There are two bandwidth constrained models which define the relationship between CTs and BCs: Maximum Allocation Model (MAM)[FL05] assigns a BC to each CT. From a practical point of view, the link bandwidth is simply divided among the different CTs. Russian Dolls bandwidth allocation model (RDM) defined in [LF05] improves bandwidth efficiency over the MAM model by allowing CTs to share bandwidth. RDM assigns BC to groups of CTs in such a way that a CT with the strictest QoS requirements (e.g., CT7 for VoIP) receives its own bandwidth reservation, BC7; a CT with the next strictest QoS requirements, CT6, shares bandwidth reservation BC6 with CT7 (BC6 $>$ BC7); and so on, up to CT0 (e.g., best effort traffic) which shares BC0 (i.e., the entire link bandwidth) with all other types of traffic.

The DS-TE model also defines a mechanism that allows the release of shared bandwidth occupied by lower priority traffic when higher priority traffic arrives. The disadvantage of RDM in comparison to MAM is that there is no isolation between the CTs. Preemption must be used to ensure that each CT get its amount of bandwidth no matter of the level of contention by other CTs. In order to implement DS-TE, the routing and signalling protocols must be extended beyond the currently defined traffic engineering extensions to carry additional information as described in [Fau05].

3.4.5 Protection and Restoration

In previous sections it has been stated that path computation algorithms are often associated with a protection path calculation, ensuring dedicated survivability. The explosive evolution of transmission rates in network backbones dictates that survivability is becoming a real issue: a short network failure could lead to huge data loss. Hence, new routing algorithms are associated with a protection path calculation module. Load balancing, presented

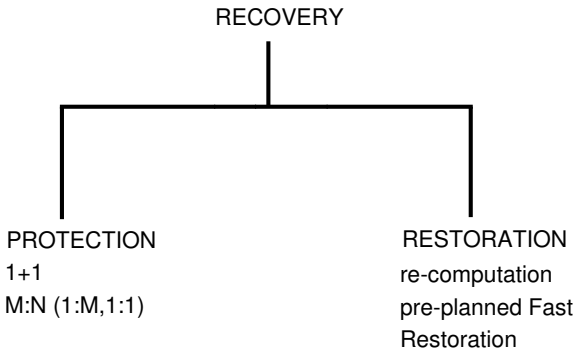

Figure 3.7: Recovery in MPLS networks

throughout the previous sections, is a suitable solution, because it permit to route traffic on the remaining paths when some of them are affected by a failure.

Typically, local protection achieves the highest recovery time but consumes more resources compared to global techniques that need more signalling (and more time) to propagate the failure but reduce resource utilization. The diversity of recovery techniques in an MPLS environment increases the complexity of the calculation process and leads to a big variety of calculation techniques.

The solution for the connection survivability problem depends on the recovery strategy implemented in the network. Different recovery strategies could be enabled in MPLS, which basically involve *protection* or *restoration*, as sketched in Figure 3.7. A general framework for MPLS-based recovery is defined in [SH03].
Moreover, two recovery classes may be distinguished:

- *path-level*, in which a failure notification is propagated till the end nodes of the affected LSP and there solved (a.k.a. end-to-end);
- *span-level*, in which a failure is notified and solved at intermediate nodes, next to the failed resource (a.k.a. local repair).

Protection strategies (like the load balancing approach mentioned earlier) perform pre-calculation and pre-allocation of a backup LSP or set of spans, while restoration strategies dynamically allocate a new LSP (or set of spans) at time of failure (i.e., on-the-fly). Another restoration strategy is to pre-computed and only booked for a future restoration (Fast Restoration).

Because of the sub-optimality of the resulting backup paths, span level strategies are prone to waste resources in the network, whereas end-to-end recovery strategies are more efficient, because they provide the computation

of the best end-to-end backup path in the network. Moreover, restoration fits better the dynamical assignment/release of the network resources with respect to protection; but, in case of a fault, a higher blocking probability for the restoring traffic might be experimented, due to the failure handling by control plane mechanisms instead of hardware ones (e.g. detection, notification and mitigation).

A common requirement for all the recovery strategies shown above is the disjointness between the resources (links or nodes) used by the primary route and by its backup. This is needed to minimize the blocking probability of the dynamical recovery action in case of fault. Different levels of disjointness for LSPs could be defined:

- *node*, in which different nodes (and different links) are crossed by the primary-backup pair of LSPs;
- *link*, in which only different links are crossed by the two LSPs;
- *SRLG*, in which the Shared Risk Link Group lists of the two LSPs have no intersection.

A set of links may constitute a Shared Risk Link Group (SRLG) if they share a resource whose failure may affect all links in the set. For example, two fibers in the same conduit would be in the same SRLG (note that a link may belong to multiple SRLGs). The SRLG information is administratively configured (i.e., is an administrative constraint for the CBR process).

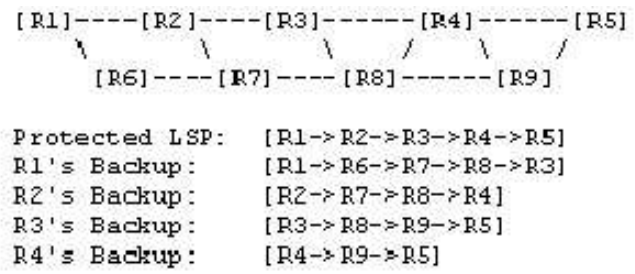

Figure 3.8: One-to-One backup technique (source: IETF)

The Fast Reroute technique, defined in [PSA05], enable recovery times of tenths of milliseconds, comparable with SDH protection times. The RFC defines two methods:

- The *one-to-one backup method* creates detour LSPs for each protected LSP at each potential point of local repair (Figure 3.8).

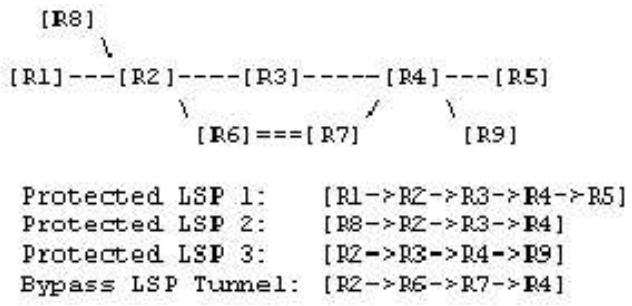

Figure 3.9: Facility backup technique (source: IETF)

- The *facility backup method* creates a bypass tunnel to protect a potential failure point; by taking advantage of MPLS label stacking, this bypass tunnel can protect a set of LSPs that have similar backup constraints (Figure 3.9).

Both methods can be used to protect links and nodes during network failure. The described behavior and extensions to RSVP allow nodes to implement either method or both and to inter-operate in a mixed network.

3.5 Control Plane Based Provisioning

As defined in the introduction of the thesis document, provisioning is a matter of allocating resources to traffic demands, and is composed of two phase processes which involve *Path Computation* and *Connection Establishment*. Several techniques have been discussed to approach the path computation problem, in different timescales of the traffic engineering process. Two link-state routing protocols with traffic engineering extensions have been defined as the path computation components of the MPLS architecture:

- Open Shortest Path First (OSPF-TE) and
- Intermediate System-Intermediate System (IS-IS-TE)

defined in normative documents [KKY03] and [SL04] respectively.
Regarding connection establishment, two signalling protocols have been defined by the MPLS normative, namely the

- ReSerVation Protocol with Traffic Engineering extensions (RSVP-TE), and the
- Constraint-Based LSP Setup using LDP (regarded as CR-LDP)

defined in [ABG+01] and [JAC+02]. After a few technical and political discussions in the IETF, it soon became clear that most vendors would adopt RSVP-TE as the standard signalling protocol. The decision was made official in [AS03], stating the IETF MPLS Working Group consensus to continue to develop RSVP-TE as the signalling protocol for MPLS Traffic Engineering applications, and not to undertake any new work related to CR-LDP. For this reason only the former will be considered afterwards.

3.5.1 Link-state routing protocols with traffic engineering extensions

Constraint based routing requires additional link-state attributes beyond the usual connectivity information. Common link attributes include maximum bandwidth, maximum reservable bandwidth, unreserved (available) bandwidth per priority and resource class (color). Other attributes include protection type on a link and Shared Risk Link Group (SRLG).

These attributes help constraint based routing to choose links with appropriate protection in diverse path computation. Enhanced link state IGPs flood link state updates if there is a change in the link attributes e.g. available bandwidth parameter. Excessive flooding is avoided by imposing a ceiling on flooding frequency and/or ensuring link state updates are done only when there is a *significant* change in bandwidth.

Note that the routing protocols solve only a part of the path computation problem, i.e. the actualization of the topological database of network nodes with the enhanced link-state attributes, for usage by a CBR computation engine. The pure Control Plane based path selection is performed by the head-end (ingress LSR) of an LSP, using some variant of the Constrained Shortest Path First (CSPF) algorithm. The pseudo-code for the CSPF is given below:

Algorithm 1 CSPF Algorithm

```
For LSP = (highest priority) to (lowest priority){
  Prune links with insufficient bandwidth
  Prune links that do not contain an included resource class
  Prune links that contain an excluded resource class
  Calculate the shortest path from the ingress to egress
  Select among equal-cost paths
  Pass the explicit route to the signalling protocol
}End For
```

[XHBN00] suggest a procedure for the computation of backup LSPs. The idea is to repeat the above procedure on the remaining graph, that is, with resources used by the primary LSPs deducted. Note that the pruning will

also affect all links and nodes used by the corresponding primary LSP, to avoid any single point of failure for both the primary and the backup LSPs.

Constraint-Based Routing is crucial for providing traffic engineering in MPLS networks, but must be complemented by a set of processes that feed information to (for instance, the IGPs link state advertisements) and consume results from the CBR process (notably, the signalling of LSPs). A typical commercial Ingress LSR performs this processes (using the CSPF algorithm) as depicted in Figure 3.10.

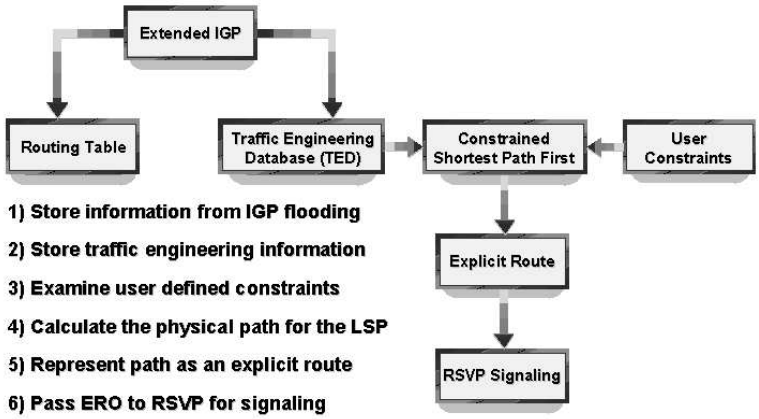

Figure 3.10: MPLS TE process in an Ingress LSR (source: Juniper)

3.5.2 Path signalling

Once the CBR process determines an Explicit Route for a Traffic Engineered LSP (TE-LSP), the setup procedure start. RSVP is a separate protocol at the IP level, i.e. it uses IP datagrams to communicate between LSR peers. RSVP-TE is a soft state protocol (i.e. it does not require the maintenance of TCP sessions), therefore it must handle the loss of control messages. The basic flow for setting up an LSP using RSVP-TE is shown in Figure 3.11 and described hereafter:

- The path selection process has determined that an LSP from ingress LSR A towards egress LSR C will use the Explicit Route (B,C). LSR A forwards a PATH message to LSR B which contain the traffic parameters requested for the new route and the Explicit Route Object (ERO), among other attributes.
- LSR B receives the PATH request, determines that it is not the egress for this LSP, and forwards the request along the route specified in the ERO. It modifies the explicit route in the PATH message and passes

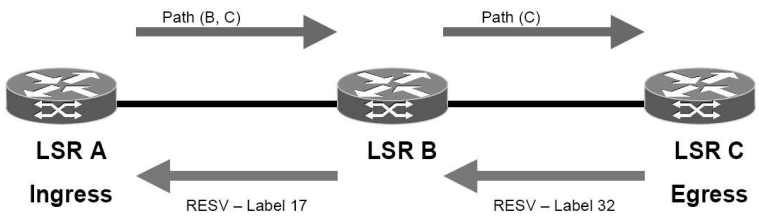

Figure 3.11: RSVP-TE path establishment (source: Data Connection)

the message to LSR C. A *path state* or session has been created for this reservation as a result of the PATH message processing.

- LSR C determines that it is the egress for this new LSP, determines from the requested traffic parameters the amount of bandwidth it needs to reserve and allocates the resources required. It selects a label for the new LSP and distributes the label to LSR B in a RESV message, which also contains actual details of the reservation required for the LSP.
- LSR B receives the RESV message and matches it to the original request using the LSP ID contained in both the PATH and RESV messages. It determines what resources to reserve from the details in the RESV message, allocates a label for the LSP, sets up the forwarding table, and passes the new label to LSR A in a RESV message.
- The processing at LSR A is similar, but it does not have to allocate a new label and forward this to an upstream LSR because it is the ingress LSR for the new LSP. A RESVConf message is returned to the egress LER confirming the LSP setup.

It should be noted that, none of the downstream, upstream or refresh messaging between LER and LSRs are considered to be reliable, because raw IP datagrams are used as the communication mechanism. Another important remark is that intermediate nodes may modify the ERO before forwarding the PATH message. This capability is exploited by one of the possible implementations of the Routing and Management Agent (RMA) presented in section 5.2.

RSVP-TE feature set is robust and provides significant capabilities to provide traffic engineering in MPLS:

- QoS and Traffic Parameters.
- Failure Notification, Crankback: upon failure to establish an LSP or loss of an existing one, head-end is notified.

- Failure Recovery, Fast Rerouting: "make before break" when rerouting.
- Loop Detection: required for loosely routed LSPs only, also supported for re-pathing.
- Management: LSP ID identifies each LSP, thereby allowing ease of management to discrete LSPs.
- Record Route objects: provide the ability to describe the actual setup path to interested parties.
- Path Preemption: the ability to "bump" or discontinue an existing path so that a higher priority tunnel may be established.

The protocol is in permanent evolution and new features are added as requirements are identified in the (G)MPLS architecture.

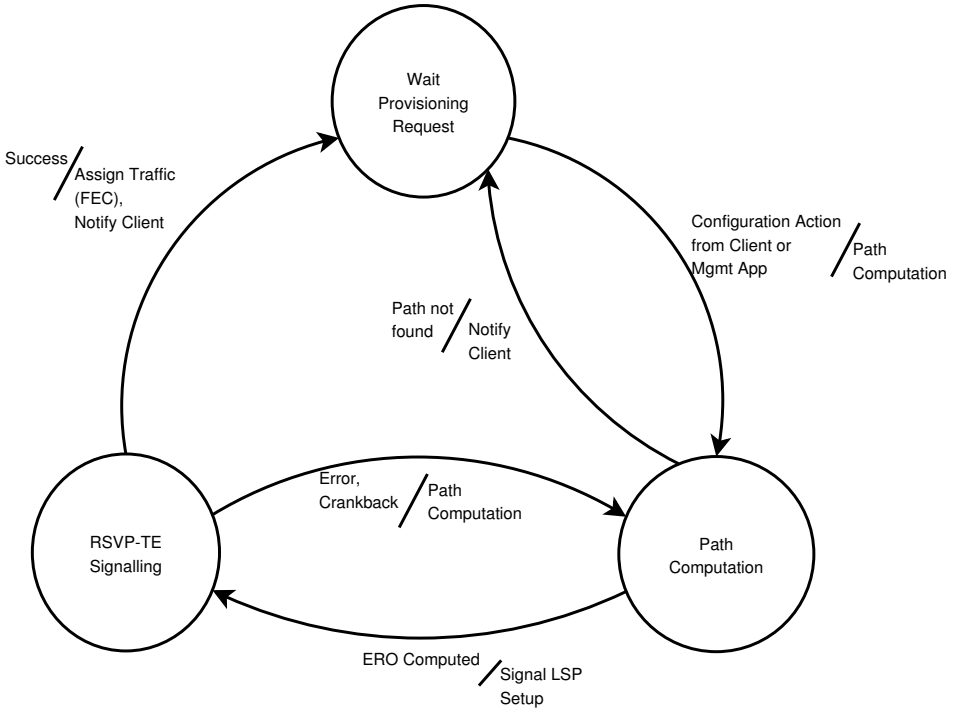

Figure 3.12: Ingress LSR Provisioning FSM

3.5.3 Provisioning Scenario

The previous discussion reveals that actual components of Control Plane based provisioning are:

- The enhanced link-state routing protocol

- The signalling protocol
- The path computation process running on the ingress LSR

The ingress LSR is the active actor of the provisioning process, and its functionality in this regard can be described by the Finite State Machine depicted in Figure 3.12.

The path computation process was already depicted in Figure 3.10, and is summarized in algorithm 2.

Algorithm 2 LSR Path Computation

```
newLSP = get_provisioning_request();
// newLSP is a reference to the provisioning request
newERO = compute_path(newLSP);
// the compute_path(); process uses:
// the Topological Database
// the User Constraints
// the objective function and administrative constraints
RSVPTE_signal(ERO,QoS & other attributes);
```

Note that according to the FSM, this process is re-run in case of path signalling error (crankback procedure).

3.6 Conclusions

MPLS provides a rich set of Traffic Engineering features, including load balancing and survivability capabilities, enabling realizable QoS models such as the Differentiated Services DS-TE model. The Control Plane based provisioning is based in two components of the MPLS architecture, namely the enhanced link-state routing protocols and the signalling protocols, and a Constraint-Based Routing functionality. This functionality is performed by the head-end of the provisioned LSP, using algorithms like CSPF that can compute feasible network path under certain (restricted) constraints. In conclusion, the overall provisioning process benefit from enhanced routing and signalling functionalities, and could be further improved with and evolved Constraint-Based Routing process. This is taken into account in the design of the proposed RMA solution in Chapter 5.

Chapter 4

MPLS Management

4.1 Introduction

The evolution path towards the next generation IP over optical networks is supported by the MPLS common Control Plane that enables inter-operation of the IP client layer with the Optical Transport Network, following well defined models, as described in Chapter 2. However, there are alternative proposals to support the integration of IP and the OTN provisioning, based on the Management Plane. For instance, the IST WINMAN project specified an Integrated Network Management System (INMS) for providing IP over WDM connectivity services, mostly using management functions supported by control functions wherever applicable [RHK+03]. The provisioning of LSPs supported by Optical SubNetwork Connections (path-based connectivity services) is integrated into the traditional OSI management FCAPS (Fault, Configuration, Accounting, Performance, Security) management functions, with a policy-based approach. Other management systems presented in Section 4.3, confirm that the intelligent Control Plane provide an important part of the operational solution in MPLS networks, which shall be complemented with well-known, trusted management techniques and tools. In this regard, a set of definitions and tools that provide support for Operation and Management (OAM) of MPLS networks are becoming available [NSF04],[AN05]. This chapter discusses both simple OAM tools and advanced management frameworks that may inter-operate to deliver an operational solution for network operators.

4.2 Basic MPLS Management Tools

This section presents a brief review of the existing OAM tools, and the MIB modules defined for MPLS and related protocols. OAM is intended to assist management applications using the data plane, and are mainly devoted to fault localization and notification. These OAM tools are different from Con-

trol Plane features that permit similar functionality, e.g., the aforementioned extensions to RSVP-TE in the GMPLS framework to detect and notify faulty conditions (see section 2.5.1).

The well-known SNMP management framework, defined in [HPW02], comprises the *agents*, which are entities resident in network nodes, containing command responder and notification originator applications, at least one *manager*, which is an entity containing command generator and/or notification receiver applications, and a management protocol (SNMP), used to convey management information between the aforementioned entities. Another basic element in the framework is the information model, or Structure of Management Information, a collection of managed objects, residing in a virtual information store, termed the Management Information Base (MIB). Collections of related objects are defined in *MIB modules*. MIB modules defined for MPLS and related application (notably traffic engineering) will be reviewed.

4.2.1 OAM Tools for LSP Connectivity Management

The MPLS OAM, as defined in [ITU02b], provides OAM techniques based on the following OAM packets, designed primarily to support explicit routed LSPs:

- Connectivity Verification (CV): the CV flow is generated at the Ingress LSR with a nominal frequency of 1/s and terminated at the Egress LSR, in a per-LSP basis. The CV packet contains a network-unique identifier (TTSI) so that all types of defects can be detected.
- Forward Defect Indication (FDI): the FDI flow is generated in response to detecting defects (e.g. from the CV flow). Its primary purpose is to suppress alarms in layer networks above the level at which the defect occurs. It is generated at either:
  1. the LSR, which first detects a dServer/dUnknown defect, or
  2. the LSP terminating LSR for all MPLS layer defects.
- Backward Defect Indication (BDI): the BDI flow is injected on a return path (such as a return LSP) to inform the upstream LSR (which is the source of the forward LSP) that there is a defect at the downstream LSP's LSR sink point. BDI therefore tracks FDI in terms of its period of generation. BDI packets may be useful in 1:1/N instances of protection switching.

Note that to be able to send the BDI upstream, it is required to have a return path. A return path could be:

- A dedicated return LSP.
- A shared return LSP, which is shared between many forward LSPs.
- A non-MPLS return path, such as an out of band IP path.

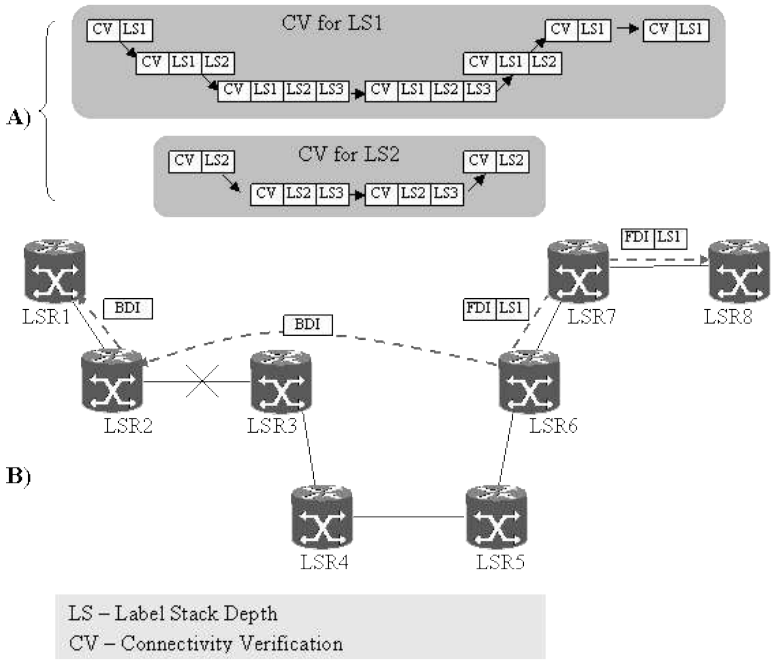

Figure 4.1: MPLS OAM functionality

All OAM packets are identified within a LSP traffic stream by the use of globally well-known and reserved label codepoint, defined by [Oht02].

CV OAM packets are distributed from ingress to egress building a label stack depths. Figure 4.1 A) describes how the CV packets are sent using level depth 1 and level depth 2 in the label hierarchy.

Figure 4.1 B) describes what happens when a failure is detected, which LSR detects the failure and how it tells the others about the failure. The LSRs belongings to different LSPs and uses a label hierarchy to reach from ingress to egress LSR as follows:

- LSP1 from LSR4 to LSR5 has label stack depth one;
- LSP2 from LSR2 through LSR3 and over LSP1 to LSR6 uses a label stack depth of two;
- and finally LSP3 from LSR1 over LSP2 through LSR7 to LSR8 uses a label stack depth of three.

Let us consider a failure between LSR2 and LSR3. This will have consequences for both LSP2 and LSP3. Both LSR6 and LSR 8 will detect that a failure has occurred even when the failure actually is at LSP2. To suppress alarms for LSP3 at LSR8, LSR6 have to inform this router by sending FDI packets along the same path as the LSP3 would be using before failure occurred. It is not only necessary to inform the downstream egress LSRs, LSR6 have to inform LSR2 (LSP2's ingress LSR), which in its turn will inform LSR1 about the failure as well by sending BDI packets (BDI packets are sent in the way described above).

4.2.2 MPLS Ping

MPLS ping, defined in [KS05], is a simple and efficient mechanism that can be used to detect data plane failures in MPLS LSPs, which cannot always be detected by the MPLS control plane. This mechanism provides a tool that enables users to detect traffic "black holes" or misrouting and to isolate faults. It is modelled after the ICMP echo request and reply, used by *ping* and *traceroute* to detect and localize faults in IP networks.

The basic idea is to test that packets that belong to a particular FEC actually end their MPLS path on an LSR that is an egress for that FEC. Therefore, an MPLS echo request carries information about the FEC whose MPLS path is being verified. The MPLS ping packet is encapsulated by an UDP packet and contains parameters like Sequence Number and Time Stamp. This echo request is forwarded just like any other packet belonging to that FEC. When the ping packet reach the end of the path, it is examined at the control plane of the LSR, which then verifies that it is indeed an egress for the FEC. In traceroute mode, which is the fault isolation mode, the packet is sent to the control plane of each transit LSR, which performs various checks that it is indeed a transit LSR for this path.

The MPLS echo reply must travel upstream in response to a MPLS echo request. The first option to forward the reply in reversed direction towards the echo request source is to set the Reply Mode to the value Router Alert. When a router receives this option, it must forward the packet as an IP packet. The second option is to send the echo reply via the control plane, which is only defined for RSVP-TE LSPs.

4.2.3 MPLS MIBs

SNMP is mainly used for monitoring purposes, such as the described methods for estimation of the traffic matrix in section 3.3.2.2. In SNMP terms, this means that applications use more frequently *get* requests than *set* operations, and this situation is coherent with the fact that most of the set options are often ignored by SNMP agent implementations on network devices. As a consequence, managing a commercial router is unavoidably tightened to pro-

prietary management applications (i.e. Element Managers). Nevertheless, some interesting set features can be found in the MPLS MIB modules, such as write/create access provided by the Traffic Engineering MIB (TE-MIB) to configure an TE-LSP at an Ingress LSR (triggering the signalling protocols to actually setup the LSP).

The considered MIB modules,are grouped under the MPLS Object Identifier (OID) tree structure , as depicted in Figure 4.2.

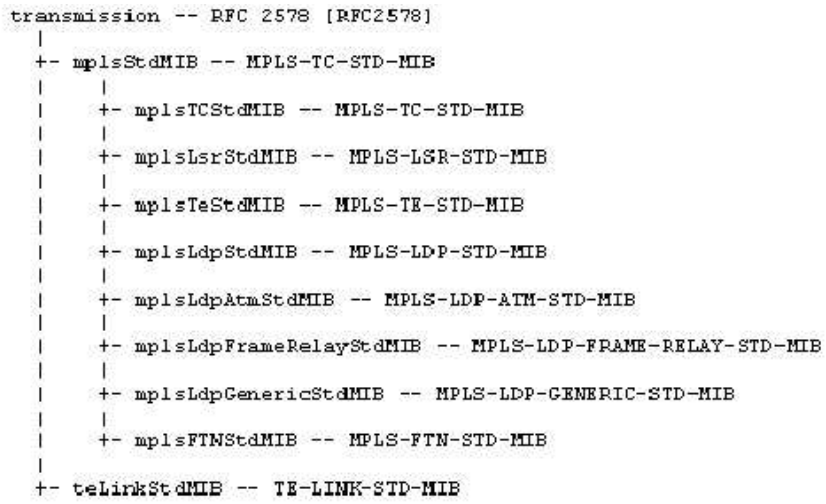

Figure 4.2: MPLS MIB OID tree (source: IETF)

A brief description of each module is given bellow:

- MPLS-TC-STD-MIB [NC04], defines textual conventions that may be common to MPLS related MIB modules. These conventions allow multiple MIB modules to use the same syntax and format for a concept that is shared between the MIB modules. All of the other MPLS MIB modules import textual conventions from this MIB module.

- MPLS-LSR-STD-MIB [SVN04], describes managed objects for modeling an MPLS Label Switching Router (LSR). This puts it at the heart of the management architecture for MPLS. This MIB module is used to model and manage the basic label switching behavior of an MPLS LSR. It represents the label forwarding information base (LFIB) of the LSR and provides a view of the LSPs that are being switched by the LSR in question. Since basic MPLS label switching is common to all MPLS applications, this MIB module is referenced by many of the other MPLS MIB modules. In general, MPLS-LSR-STD-MIB provides a model of incoming labels on MPLS-enabled interfaces being mapped

to outgoing labels on MPLS-enabled interfaces via a conceptual object called an MPLS cross-connect. MPLS cross-connect entries and their properties are represented in MPLS-LSR-STD-MIB and are typically referenced by other MIB modules in order to refer to the underlying MPLS LSP.

- MPLS-LDP-STD-MIB [CSL04]describes managed objects used to model and manage the MPLS Label Distribution Protocol (LDP).
- The MPLS-LDP-GENERIC-STD-MIB module provides objects for managing the LDP Per Platform Label Space and is typically implemented along with the MPLS-LDP-STD-MIB module.
- The MPLS-LDP-ATM-STD-MIB module is typically supported along with MPLS-LDP-STD-MIB by LDP implementations if LDP uses ATM as the Layer 2 medium.
- The MPLS-LDP-FRAME-RELAY-STD-MIB module is typically supported along with MPLS-LDP-STD-MIB by LDP implementations if LDP uses Frame Relay as the Layer 2 medium.
- MPLS-TE-STD-MIB [SNV04], describes managed objects that are used to model and manage MPLS Traffic Engineered (TE) Tunnels. This MIB module is based around a table that represents TE tunnels that either originate from, traverse via or terminate on the LSR in question. The MIB module provides configuration and statistics objects needed for TE tunnels.
- MPLS-FTN-STD-MIB [NSV04]describes managed objects that are used to model and manage the MPLS FEC-to-NHLFE (FTN) mappings that take place at an ingress Label Edge Router (LER). In the case of an IP-to-MPLS mapping, the FEC objects describe IP 6-tuples representing source and destination address ranges, source and destination port ranges, IPv4 Protocol field or IPv6 next-header field and the Diff-Serv Code Point (DSCP).
- TE-LINK-STD-MIB describes managed objects that are used to model and manage TE links, including bundled links, in an MPLS network. The TE link feature is designed to aggregate one or more similar data channels or TE links between a pair of LSRs. A TE link is a sub-interface capable of carrying traffic engineered MPLS traffic. A bundled link is a sub-interface that bonds the traffic of a group of one or more TE links.

Figure 4.3 shows the relationship between the MPLS MIB modules described above. Note that all the MPLS MIB modules depend on MPLS-TC-STD-

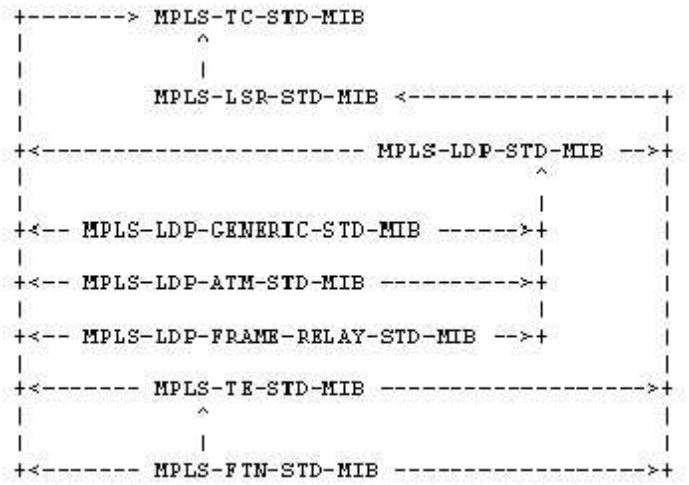

Figure 4.3: MPLS MIB modules interdependencies (source: IETF)

MIB, and many reference objects in MPLS-LSR-STD-MIB. It worth mentioning that many of the MPLS MIB modules have dependencies on the Interfaces MIB [MK00], which shows the predominance of such MIB in SNMP application, mostly related with interface counters monitoring. In particular, TE-LSPs (MPLS Tunnels) are considered logical interfaces, then they are represented as entries in the ifTable (meaning that existing monitoring applications can be used for TE-LSPs without modifications).

Regarding the usability of the described MIB modules, all LSPs may appear in MPLS-LSR-STD-MIB, and TE-LSPs (MPLS tunnels) may be represented in MPLS-TE-STD-MIB with their cross-connects indicated in MPLS-LSR-STD-MIB. Tunnels are often (although not always) set up with a series of constraints that may be represented in MPLS- TE-STD-MIB. Note that a distinguishing feature of a tunnel is that it has an ingress and an egress, where LSPs established through LDP may be end-to-end or may be hop-by- hop. These considerations suggest that these are the most important modules for application developers, together with the generic interfaces MIB, as mentioned before.

4.3 MPLS Management Frameworks

The state of the art in management applications for the IP over Optical network architecture presents a clear asymmetry between the the Optical Layer, where management has been historically dominant due to the absence of control plane functionality on network devices, and IP, where routing and signalling (i.e. device intelligence) have been in place since the foundation

of the protocol. MPLS control plane enables layer integration, augmenting IP with traffic engineering capabilities as extensively discussed in Chapter 3. As a consequence, complex functions have been added to network devices (i.e constraint-based path computation), which need ever growing resources to cope with this complexity in terms of processing time, for instance. Finding optimal, even feasible network path that meet certain constraints is a challenging task for head-end routers.

Few traffic engineering systems have been proposed to assist network devices in path computation and resource assignment in MPLS networks; they are described in the following subsections.

4.3.1 RATES

The Routing and Traffic Engineering Server (RATES), defined in [AKK+00], is a software system developed at Bell Laboratories for MPLS traffic engineering, and is built using a centralized paradigm. Provisioning is performed by configuration of the head-end (ingress LSR), which spawns off signaling from the source to the destination for LSP setup. This communication is considered a policy decision by RATES, and consequently the communication is realized using the Common Open Policy Service (COPS) protocol [DBC+00]. The system uses a relational database for persistence, and implements the Minimum Interference Routing Algorithm (MIRA) *[KKL]* for path computation of LSPs. RATES presents a component-based, expandable architecture. The major modules are listed below:

- Explicit Route computation using MIRA.
- COPS Server for communication with head-end routers.
- Data Repository.
- Network Topology and State Discovery component, running OSPF.

Moreover, it uses a distributed CORBA-based bus/message dispatcher, featuring a Graphical User Interface (GUI) and an open Application Programming Interface (API), with a message bus connecting these modules. Some of the ideas presented by this proposal will be considered by the RMA presented in chapter 5, in particular the usage of OSPF to gather network information is an important contribution. RATES operational environment is dynamic, receiving provisioning requests one by one, without knowledge of future demands. However, RATES capability for resource optimization is restricted by the absence of monitoring functionality.

4.3.2 TEQUILA

Traffic Engineering for QUality of service in the Internet, at LArge scale (TEQUILA), presented in *[MCG+03],* proposes an integrated architecture

for providing end-to-end QoS in a DiffServ-based Internet. In TEQUILA, a framework for Service Level Specification has been produced, an integrated management and control architecture has been designed.

The TEQUILA architecture includes control, data and management planes. The management plane aspects are related to the concept of Bandwidth Broker (BB), where each Autonomous System shall deploy its own BB. The BB includes components for monitoring, traffic engineering, SLS management and policy management.

The Traffic Engineering subsystem is integrated into a resource provisioning cycle. The Traffic Forecast module produces a traffic matrix which is used by the Network Dimensioning component to allocate resources. The resultant resource availability matrix feeds the online Admission Control mechanisms. Network monitoring and notifications functions are in place to trigger Network Dimensioning process in case available capacity can not be found to accommodate new connection requests.
This architecture is very interesting and shows a similar approach for MPLS networks design and management compared to TEAM, referenced below.

4.3.3 TEAM

The Traffic Engineering Automated Manager (TEAM) claims to be an adaptive manager that provides the required quality of service to the users and reduces the congestion in the network [SAdO+04]. These objectives are achieved by reservation of bandwidth resources, and efficient distribution of the load, using online measurements of the network state. TEAM major components are:

- the Traffic Engineering Tool (TET), which adaptively manages the bandwidth and routes in the network,
- a Measurement and Performance Evaluation Tool (MPET), which measures important parameters in the network and inputs them to the TET,
- and a Simulation Tool (ST), which may be used by TET to consolidate its decisions.

The TET tool perform offline resource and routing management, including LSP setup/dimensioning, LSP capacity allocation , LSP preemption and LSP routing. The MPET introduces enhancements of the popular MRTG tool [OR05], whereas the ST component is a dimensioning aid to the MPET.

4.3.4 Wise<TE>

Wise<TE> Traffic Engineering server for MPLS networks, presented in [CYC+02], present a complete analysis of the caveats of IP-based traffic

engineering, the limitations of CSPF routing running on network nodes and other issues discussed in Chapter 3, but reaches the limited conclusion that the solution is an *offline* traffic engineering tool. The design of the system is component based, using a CORBA bus as distributed environment. Wise <TE> major components are the following:

- Traffic Measurement and Analysis Server, which performs collection, characterization and analysis of traffic information.
- Routing Advisor for Traffic Engineering, a planning tool with CBR capabilities.
- Resource Monitoring Server, a topology and configuration information repository
- Policy Server, which configure policies on network devices using vendor specific device agents.

In conclusion, this proposal is similar to the RATES system. Its major contribution regarding the RMA proposal is the dynamic gathering of network states to feed the traffic engineering functionality.

4.4 Provisioning Scenario

A typical provisioning scenario using an evolved MPLS management system, as shown in Figure 4.4, will follow these steps:

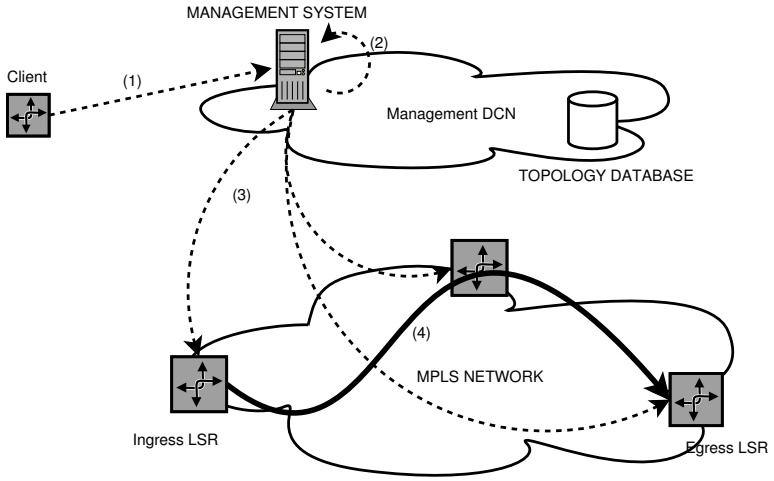

Figure 4.4: Management based LSP setup

1. The Client requests the establishment of an LSP to the management system. The system maintains network states participating in the IGP and using monitoring processes.
2. The management system computes a path that fulfill the requirement
3. The management system contacts the appropriate LSRs in the network for LSP configuration
4. The requested LSP is operational

Note that in a generic management system, the topology database is not guaranteed to be synchronized with the network link-states. provisioning time is also affected by communication of the management platform with the network devices that must be configured for each provisioning operation.

4.5 Conclusions

A number of simple, data-plane based tools have been devised for user level LSP monitoring, along with MIB definitions that enable a complete support for SNMP-based MPLS management.

	RATES	TEQUILA	TEAM	Wise<TE>
Framework	Distributed CORBA components	Unknown	Centralised server	Distributed CORBA components
Monitoring	No	Yes	Yes (enhanced MRTG)	Yes
Topology	Run IGP	Unknown	Static, from routers configuration	Run IGP
Path Computation	Online MIRA Explicit Route	Offline based on Traffic Matrix	k-Shortest Path	CSPF
Client Communication	COPS	Unknown	TFTP	Device Agent
Path establishment	Signalling	Unknown	Configuration	Configuration

Table 4.1: MPLS management systems comparison

Some advanced MPLS management systems have been presented, which exhibit a number of similarities and some interesting features that shall be

considered in the design the Routing and Management Agent (RMA) provisioning solution. Table 4.1 summarizes relevant aspects of the proposals.

The overall review of the characteristics advise a component-based system, with monitoring capabilities, which participates in the IGP process. The communication with network devices is largely dependent on specific routers capabilities; anyhow, COPS is widely deployed and could be a reasonable choice. Path computation is non-standard by nature, and each proposal implement algorithms that claim to fulfill certain performance objectives. The RMA, being a brand new proposal, shall be flexible in this respect to be able to support different capabilities and do not be restricted by computational resources. Finally, the LSP establishment process is performed by typical management mechanisms (e.g. transferring device configuration using TFTP), except in the RATES prototype, which uses head-end signalling.

This approach is promising because the Control Plane not only performs this task faster than typical management information transfer processes, but it also provides the crankback capability which permit to re-compute an LSP if a failure occurs, among other advantages already discussed.

Chapter 5

Contribution to Intra-domain Provisioning: the Routing and Management Agent

5.1 Introduction

This chapter presents the thesis contribution to the problem of intra-domain resource allocation in MPLS networks. The proposal stands for decoupling path computation from path establishment and packet forwarding on Label Switched Routers (LSRs), based on the functionality of an independent entity called Routing and Management Agent (RMA), which peers with both the Control Plane and the Management Plane.

Network nodes (LSRs) main function is packet forwarding; MPLS Control Plane adds signalling capabilities to LSRs, which can timely perform path establishment, making use of the standard signalling protocol RSVP-TE [ABG+01]. In the proposal, the CBR computation is delegated to a specific purpose server, the RMA. This agent can perform offline and online (in near real time) CBR, using arbitrary large computation power. To achieve this goal, the RMA makes use of existing algorithmic and computing techniques, such as classic High Performance Computing (HPC) strategies (i.e. problem parallelization). These issues are discussed in section 5.10.2.

As shown in Figure 5.1, the RMA is a routing and signalling peer node in the Control Plane, with network monitoring capabilities. Being a routing peer enables the RMA to gather network topology updates directly from link state advertisements. CBR computation functionality is orthogonal to traffic forwarding; therefore, the RMA functionality may reside in any network node, and specifically in nodes with good topological knowledge of the network, such as the Area Border Routers (ABRs), as it is discussed in Section 5.7. The proposal considers a specialized RMA entity, which is topologically co-located in the network but avoids traffic forwarding.

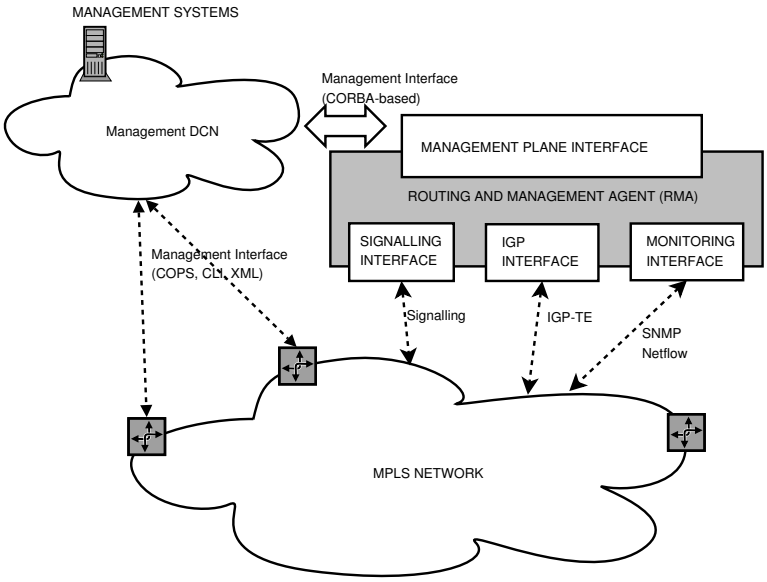

Figure 5.1: Routing and Management Agent

The RMA, while boosting Control Plane-based provisioning, can be used as a Traffic Engineering tool by management applications. The Management Plane interface can be used to establish cooperative relationship with other RMAs, as described in the inter-area and inter-domain sections 5.7 and 6.4.

The basic RMA functionality can be summarized as follows:
a given ingress LSR, when signalling a new LSP, demands a path computation to the RMA, which computes an explicit route that satisfies the given constraints. Once the route is known, normal LSP setup is achieved using the standard signalling.

A communication protocol is needed between the LSRs and the RMA, and among RMAs in the network. A first proposal, presented in the following section, consist of an innovative usage of the standard RSVP-TE signalling protocol. Other alternatives are discussed in Section 5.5.

Connectivity provisioning may be requested by the management applications and/or by client routers, using management protocols and/or signalling. A suitable protocol for connectivity request is COPS-PR [CSD+01], which permits to download arbitrary configuration information to network devices expressed as policies. This will be also discussed in the aforementioned Section 5.5.

The intra-area provisioning scenarios considered in this proposal are:

- Basic LSP setup using the standard signalling (unidirectional), and

- Reliable LSP setup using the RMA (uni and bidirectional, with load sharing and protection alternatives)

The inter-area considered scenarios are:

- Inter-area LSP provisioning with an Omniscient RMA, and
- Inter-area LSP provisioning using Per-Area RMAs.

These scenarios will be described and evaluated in the following sections.

5.2 Basic LSP setup using the standard signalling

In this section a basic successful unidirectional TE-LSP setup scenario will be considered, to depict the fundamental aspects of the RMA functionality and the usage of the RSVP-TE signalling protocol. Some potential problems and complex issues are reviewed afterwards.

Let us consider a LSP setup upon reception of a request at an Ingress LSR. Network nodes run the standard RSVP-TE protocol, and the RMA runs a RSVP-TE agent capable of processing messages in order to provide the path computation functionality. This basic RMA functionality can be described by the algorithm depicted in Figure 5.3.

Figure 5.2 shows the numbered sequence of events to establish a TE-LSP as described hereafter:

1. The management application (1), or the client (1'), configures the ingress LSR by means of a suitable protocol like COPS-PR. This implies the specification of a Forward Equivalence Class (FEC), QoS and other constraints, and a Explicit Route (ER) towards destination. The LSP setup is based in the specification of a "dummy" ER whose first loose hop is the RMA. The resultant Explicit Route Object is specified as follows:

   IngressLSR, RMA:loose, EgressLSR:loose

2. The ingress LSR initiates a LSP setup issuing a RSVP-TE PATH Message towards the RMA, the first loose hop of the Explicit Route Object (ERO), as configured in the previous step.

3. The RMA receives the PATH Message, and computes an Explicit Route based on the QoS descriptor and other constraints.

4. Once computed, the RMA replaces the "dummy" ERO object by the calculated ERO and sends the modified PATH Message downstream to the Egress LSR. In this first version of the solution, the modified ERO contains exclusively *strict* hops. An example with 2 intermediate nodes follows:

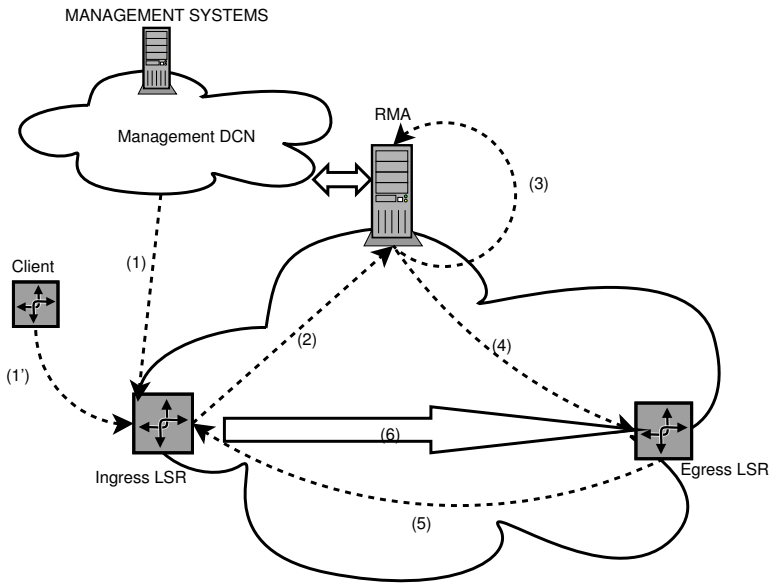

Figure 5.2: LSP setup using the RMA

IngressLSR, IntermediateLSR01, IntermediateLSR02, EgressLSR

5. When the RSVP-TE PATH Message progresses towards the Egress LSR, this node issues a RSVP-TE RESV Message upstream to the Ingress LSP following the ERO computed by the RMA. This message, while passing through the network, signals the reservation of the resources needed by the Traffic Engineered LSP.

6. Once the RESV Message reaches the Ingress LSR, the LSP is established and traffic can be assigned to the appropriate Forwarding Equivalent Class (FEC), as specified in the initial step (1).

The signalling interface processes incoming PATH messages, feeding the CBR process. Note that other incoming messages are just discarded; while this is acceptable for a proof of concept, a functional implementation shall perform error control, because many unexpected conditions may arise due to the manipulation of the RSVP-TE messages. The evolution of the LSP setup messages is shown in the sequence diagram on Figure 5.4, and described bellow:

1. Initial ERO is: *IngressLSR, RMA:loose, EgressLSR:loose*

2. While the initial PATH message progress towards the RMA across some intermediate nodes, say A, B, C, the ERO transform into: *IngressLSR, NodeA, NodeB, NodeC, RMA, EgressLSR:loose*

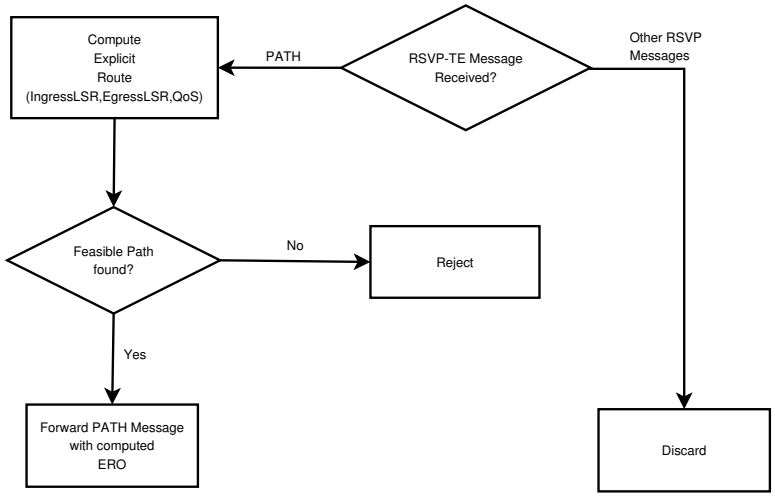

Figure 5.3: Simple RMA algorithm

3. The ERO computed by RMA is: *IngressLSR, IntermediateLSR01, IntermediateLSR02, ...,EgressLSR*

   where *IntermediateLSR01, IntermediateLSR02,....* may be different than *NodeA, NodeB, NodeC...*

4. Once the Egress LSR is reached, the RESV messages sent upstream by-pass any node excluded from the computed Explicit Route.

Potential problems that may arise:

- The RSVP-TE module in nodes A, B, C will remain waiting for either a REJECT or a RESV message. If neither is received, PATH_REFRESH messages will be issued, potentially flooding the RMA. A straightforward solution is that the RMA send REJECT messages downstream, but this may lead to a complete tear-down of the LSP reservation process at the Ingress LSR.
- Node *IntermediateLSR01* in the example (Ingress LSR next-hop in ERO computed by the RMA) would receive a PATH message from the RMA with the Ingress node *IngressLSR* as antecessor; a secure RSVP-TE implementation could check and detect such an inconsistency and would reject the intended PATH reservation.

These problems where avoided in the proof of concept, but other (e.g., commercial) implementations could fail to complete a LSP setup in the described manner. The result is that some aspects of the signalling protocol shall be modified to fulfill the intended functionality, as discussed in Section 5.5.

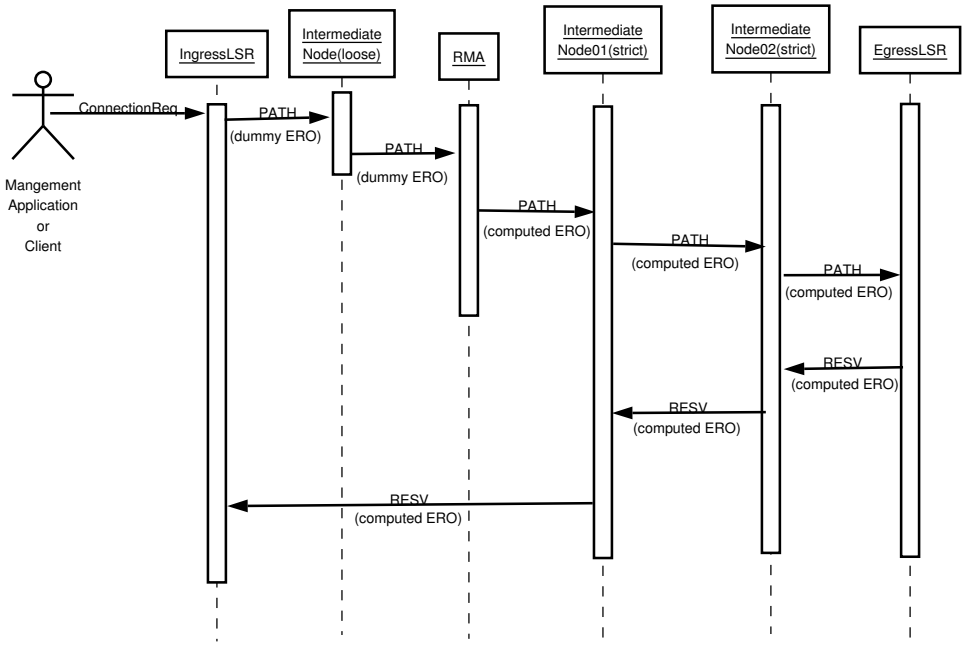

Figure 5.4: LSP setup sequence diagram

5.3 Reliable LSP setup with load sharing

The simple procedure presented in the previous section may reject a connection if there is not enough bandwidth to allocate the request in a single LSP. Splitting the load can solve this problem and contribute to an overall better sharing of resources. Moreover, it is easier to reroute smaller LSPs in case of link failures (see section 3.4). Therefore, a mechanism that permits to compute and signal more than one LSP per request is needed.

Furthermore, the discussed potential problems in the signalling advise the introduction of modifications in this regard. The proposal shall relax the restriction of using only RSVP-TE signalling and adopt a request/reply communication protocol between the LSRs and the RMAs, since the relationship between these entities is client/server. The requirements and possible options for such protocol are discussed in Section 5.5. Assuming that this request/reply protocol is in place, the connectivity setup process may be modified according to Figure 5.5, and described in the numbered sequence of events mentioned below:

1. The management application (1), or the client (1'), configures the ingress LSR by means of a suitable protocol like COPS-PR. This implies the specification of a Forward Equivalence Class (FEC), QoS and other constraints and the destination *EgressLSR*.

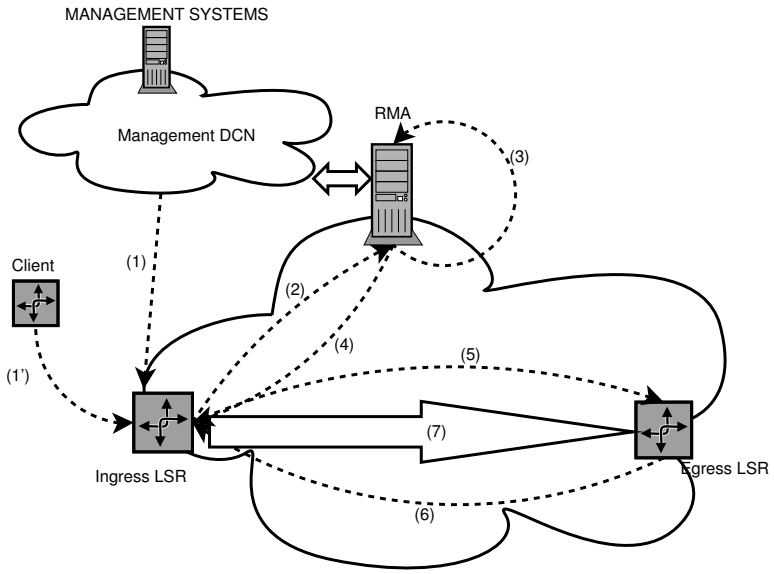

Figure 5.5: Reliable connectivity setup using the RMA

2. The ingress LSR issues a Request to the RMA using an ad-hoc protocol, specifying Origin, Destination, QoS parameters and other constraints, as signaled in step 1.

3. The RMA receives the request, and computes the needed resources to satisfy the demand. If the demand fits into a single LSP (complying with the given constraints), proceed to step 4; if not, the RMA must choose an option:

   (a) compute a bundle of LSPs to satisfy the demand, or

   (b) finish and notify the ingress LSR. In this case there is nothing more to be done.

4. The RMA computes the LSP or bundle of LSPs and issues a Reply to the ingress LSR, using the aforementioned ad-hoc protocol:

   LSP01: IngressLSR, IntermediateLSR011, IntermediateLSR0is12,... EgressLSR

   LSP02: IngressLSR, IntermediateLSR021, IntermediateLSR022,... EgressLSR

   ....

....

LSPn: IngressLSR, IntermediateLSRn1, IntermediateLSRn2,...
EgressLSR

5. The ingress LSR initiates the PATH phase of the establishment of the computed LSP(s) towards the egress LSR, using the standard RSVP-TE signalling.

6. The egress LSR issues the needed RESV messages upstream to the ingress LSR, following the Explicit Routes computed by the RMA. This messages, while progressing through the network, signal the reservation of the resources needed by the TE LSP(s).

7. Once the RESV message reach the ingress LSR, the LSPs are established forming a Traffic Trunk, and traffic can be assigned to the appropriate Forwarding Equivalent Class (FEC), as specified in the initial step (1).

This solution overcomes the potential problems of the basic proposal, at the cost of adding a request/reply protocol for the proper usage of the RMA by the ingress LSRs. Note that the core network nodes do not need to support the overhead of the extra signalling, and they continue to use just plain RSVP-TE.

5.4 Open issues regarding the RMA proposal

A careful review of the proposals presented in sections 5.2 and 5.3 raises a number of issues, enumerated below:

- The RMA gathers the topological network information as an IGP peer. The CBR computation process is based on a network graph built using this information, which may contain inaccuracies, as mentioned in Section 3.3.2.1. This implies that some of the resources assigned to a specific demand may be unavailable in path establishment time; in this case crankback procedures applies. When a LSP setup request is blocked by links or nodes without sufficient resources, the crankback scheme determines that setup failure information is returned from the point of failure to allow new setup attempts to be made avoiding the blocked resources. Crankback can also be applied to LSP recovery to indicate the location of the failed link or node, as defined by [FSI+05]. The RMA is transparent to crankback procedures; such event shall be managed by the Ingress LSR, e.g. re-requesting a path computation or using a backup path if it was requested.

- In the general case, many RMAs may reside in a given network, for example for fault-tolerance and/or load sharing reasons. Therefore, ingress LSRs need a RMA discovery and recovery mechanism. The proposed implementation described in section 5.10.2 overcomes this potential problem using a fault-tolerant, redundant design; anyhow, a complete implementation shall take it into account.

- Load sharing comes at a cost for the ingress LSR, which shall assign traffic to the established LSPs. This task involves traffic classification and scheduling to the proper outgoing queue, which would require the configuration of ingress filters and FECs to appropriately manage the incoming traffic. A suitable approach to cope with this complexity is to make use of configuration policies, which shall be loaded to the LSRs using a protocol capable of exchanging policy information.

- Another important aspect of the provisioning process is that most connectivity services (excepting broadcast/multicast streaming and the like) are bidirectional. When a client requests a bidirectional service using a management application, both a downstream and an upstream path (i.e. a bidirectional traffic trunk) shall be provided to complete the request. When the provisioning is in control of the management application, it shall take the necessary steps to complete the request (e.g., request two LSPs to the underlying provisioning manager, that is, the RMA). Another possibility is to signal the ingress LSR requesting a bidirectional traffic trunk. In this case the request/reply communication protocol shall provide the possibility to signal such a requirement. This functionality can be achieved using a *"push"* mechanism from the RMA to the egress LSR, meaning that when the RMA receives a bidirectional provisioning request, it shall compute a downstream *and* an upstream LSP (or bundle of LSPs), and signal both the ingress LSR (as before) and the egress LSR with the push mechanism.

  Once signalled, the path establishment proceeds as usual, but in this case both the ingress and the egress LSRs initiate a reservation process in opposite directions. The overall result is that using the push mechanism, bidirectional connectivity can be achieved signalling only the ingress LSR by the management application and/or the client. Moreover, due to the overlapping reservations, the process can be completed in the same timescale as in the unidirectional case, as shown by the proof of concept results in Section 5.6.

- Finally, the problem of path protection have to be considered. As mentioned in Chapter 3, load sharing inherently tackles down this issue, because in the event of a link failure, some of the paths in a bundle shall survive, if link/node/SRLG diversity was taken into account in path computation time. The obvious consequence is that link/node/SRLG

diversity is required when computing a load sharing bundle. This requirement may be explicitly contained in the request to the RMA, which may rely in the existing protection mechanisms such as Fast Rerouting, described in Section 3.4.5.

5.5 Signalling aspects of the RMA Architecture

Decoupling path computation from path establishment into different entities implies the existence of a communication mechanism between these entities. Even if they reside in the same physical facility (e.g. an ABR), the protocol can still be used.

First of all, requirements for the protocol must be defined, in order to be able to either choose an existing protocol (probably with some modifications), or design a new one that fulfill the requirements. Considering the previous sections the following requirements may be highlighted:

- The edge LSRs have a *client* role, whereas the RMA has a *server* role. The usual behaviour, as described before, is that the LSRs (clients) send a path computation request to the RMA (server), which replies with a path response once computed. This is a typical *client/server* or *request/reply* model. Nevertheless, as noted in Section 5.5, in the case of bidirectional connectivity setup, a *push* mechanism is needed, or, in other words, *unsolicited notifications* must be supported from the RMA to the LSRs.

- As mentioned in the previous section, a RMA *detection* (i.e discovery) *and recovery* mechanisms are needed.

- Other usual healthy requirements for a client/server protocol include *reliable* message exchange, support for *asynchronous* communication, *scalability* and *extensibility* features.

- *Security* and *privacy* requirements are out of scope of this work, but should be considered in a real world implementation.

It is worthy to consider extensibility in more detail. CBR may be based on a number of different constraints and objective functions, some of them known and many to be defined in the future. Consequently, is vital not only for the communication protocol but for the CBR engine itself (i.e. the RMA) to be flexible, not hard-coded to solve a set of fixed cases, but open to redefine the behaviour on execution time. This requirement leads directly to the need of using a *policy-based* approach for the definition of the supported path computation options (see Section 5.10.2). This requirement must be also supported by the request/reply protocol, so the chosen one shall be capable of encoding and transporting routing policy information.

The set of defined requirements leads to the consideration of the following protocols:

- The first approach proposed in Section 5.2, i.e. the use of the plain signalling protocol, revealed some difficulties that can only be overcame with extensions to the RSVP-TE. In fact such extensions have been proposed by [Vas05], which defines the *Path Computation Message* and the associate objects that fulfill the intended request/reply functionality. While this option is functionally valid and may cover the requirements, it implies the modification of the signalling protocol, at least in the edge nodes.
- Other existing protocols that may fulfill the given requirements are the Common Open Policy Service Protocol (COPS) [DBC+00] and the Border Gateway Protocols (BGP) [RL95], which support extensions to transport arbitrary information, and may be suitable for the RMA signalling.

The Common Open Policy Service (COPS) protocol is a simple query and response protocol that can be used to exchange policy information between a policy server (Policy Decision Point or PDP) and its clients (Policy Enforcement Points or PEPs). A classical example of a policy client is a router that needs to enforce policy-based admission control over RSVP usage.

COPS protocol has a simple but extensible design. The main characteristics of the COPS protocol include:

- COPS employs a client/server model where the PEP sends requests, updates, and deletes to the remote PDP and the PDP returns decisions back to the PEP.
- COPS uses TCP as its transport protocol for reliable exchange of messages between policy clients and a server.
- COPS is extensible, meaning that it is designed to support self-identifying objects and can support diverse client specific information without requiring modifications to the COPS protocol itself. COPS was created for the general administration, configuration, and enforcement of policies.
- COPS provides message level security for authentication, reply protection, and message integrity. COPS can also reuse existing protocols for security such as IPSec or TLS to authenticate and secure the channel between the PEP and the PDP
- COPS is stateful in two main aspects: (1) Request/Decision state is shared between client and server and (2) State from various events (Request/Decision pairs) may be inter-associated.

- Additionally, COPS allows the server to push configuration information to the client, and then allows the server to remove such a state from the client when it is no longer applicable.

The Border Gateway Protocol (BGP) is an inter-Autonomous System routing protocol that runs over TCP. The primary function of a BGP system is to exchange network reachability information with other BGP systems. This includes information on the list of traversed Autonomous Systems (ASs), which is sufficient to construct a graph of AS connectivity from which routing loops may be pruned and some policy decisions at the AS level may be enforced.

BGP4 provides a set of mechanisms for supporting different type of information exchange, and many extensions have been defined for diverse uses such as Capabilities Advertisement [CS02], carrying MPLS Label Information [RR01], and VPN Route Distribution in BGP/MPLS IP VPNs [RR05], among others.

After a transport protocol connection is established, the first message sent by each side via BGP is an OPEN message. If the OPEN message is acceptable, a KEEPALIVE message confirming the OPEN is sent back. Once the OPEN is confirmed, UPDATE, KEEPALIVE, and NOTIFICATION messages may be exchanged. These messages may transport different objects, as mentioned; in this regard, they may be adapted for the communication of path computation information between LSRs and the RMA. Moreover, in a network where the BGP/MPLS IP VPN model is implemented, the PE (Provider Edge) routers already run MPLS and BGP, therefore the cost of adding the RMA functionality may be marginal in this environment.

At first glance, both BGP and COPS could be adapted for the intended functionality; COPS was chosen for the prototype, since there is experience in this application of the protocol in the reference MPLS management systems reviewed in Chapter 4. An operational implementation may arrive to different conclusions due to network peculiarities; e.g., the BGP/MPLS IP VPN case recently mentioned.

An interesting possibility offered by the existence of a push mechanism is that being the RMA a Control Plane entity connected to management applications, it may be used directly as a path computation and provisioning tool. In this case the provisioning phase is shortened, avoiding an initial configuration of the Ingress LSR and a request/reply phase with the RMA. This result is important from the operational point of view for network operators.

5.5.1 Definitions of the IETF Path Computation Element (PCE) Working Group

The IETF has recently chartered the Path Computation Element (PCE) Working Group, proposing a model where path computation is performed by an external entity different from the head-end (ingress) LSR. This entity is the Path Computation Element, which has the same role as the RMA, whereas the ingress LSRs are qualified as Path Computation Clients (PCCs). The PCE WG is working on the application of the PCE architecture in a single domain, with possible extensions for "small group of domains" (where a domain is a layer, IGP area or Autonomous System with limited visibility from the head-end LSR). The WG considers that applying this model to Internet inter-domain is not possible for the time being.

The WG has produced three working documents, namely the *Path Computation Element (PCE) Architecture* [FVA06], the *PCE Communication Protocol Generic Requirements* [ALR06] and the *Requirements for Path Computation Element (PCE) Discovery* [LR06]. The latter present a comprehensive discussion of the protocol requirements which is useful to characterize possible candidates to play the role of the needed request/reply protocol. While this thesis have identified COPS as a valid candidate, the PCE WG is in the process of designing a new ad-hoc protocol named *Path Computation Element (PCE) communication Protocol (PCEP) [Vas06].* PCEP is transported over TCP, which guarantees reliable messaging and flow control. It defines four basic messages:

- PCReq: message sent by a PCC to a PCE to request a path computation.
- PCRep: message sent by a PCE to a PCC in reply to a path computation request. A PCRep message can either contain a set of computed path(s) if the request could be successfully satisfied or a negative reply otherwise, potentially with a set of less-constrained path(s).
- PCNtf: notification message either sent by a PCC to a PCE or a PCE to a PCC to notify of specific event.
- PCErr: message related to a protocol error condition.

The objects carried by such messages resemble much of the previous work with RSVP-TE, particularly the aforementioned extensions proposed in [Vas05]. Moreover, PCEP provides the ENCAP object, which is used to carry objects defined by other protocols such as COPS, RSVP-TE and its extensions.

The work presented in this thesis is not directly related with the PCE WG, since it was started much earlier. Anyhow, it is basically aligned with the WG definitions; moreover, an Internet Draft was submitted to the WG defining a management interface for the PCE, based on the work with the RMA [Gra06].

5.5.2 Communication of Clients and Management Applications with Ingress LSRs

In this chapter introduction it was mentioned that connectivity provisioning may be requested by the management applications and/or by client routers, using management protocols and/or signalling. It was also stated that a suitable protocol for connectivity request is COPS-PR. Since COPS was already chosen as a candidate for the LSR-RMA communication, is reasonable to assume that this will be the protocol of choice for both relationships.

This election is advantageous not only because the protocol is flexible enough to cover the requirements of both relationships, but because the ingress LSR needs to implement just one management protocol. Moreover, COPS is widely deployed in network devices and can be easily adapted to fit in the RMA architecture. Note that other usual management protocols such as SNMP, Extensive Markup Language (XML) or Command Line Interface (CLI) will continue being used by router vendors.

5.6 Evaluation of intra-area cases

The proposed provisioning processes presented in sections 5.2 and 5.3 have been tested in a simulated environment, using RSVP-TE [CFV05] and COPS [SL05] extensions to the NS-2 simulator, while an initial implementation is being developed in a trial MPLS multiservice network, as described in Appendix A.2. Basic and reliable connectivity setup scenarios using the RMA have been tested. For each scenario, a comparison between a pure Management Plane approach, a pure Control Plane approach and the RMA solution has been accomplished, using different topologies, and varying the node degree of the RMA. This means that the RMA role is alternatively assigned to nodes with different number of links connecting to their neighbors, in order to determine if there is a correlation between the RMA response time and how "well-connected" it is, in order to derive a rule of thumb for operational networks.

The intra-area cases have been tested using 10, and 100 node topologies, with different generation laws provided by the BRITE generator : Waxman and Barabasi-Albert. The former generation model builds a random topology using Waxman's probability model for interconnecting the nodes of the topology [Wax88], while the latter considers a power law in the frequency of outdegrees in network topologies, and interconnects the nodes according to the incremental growth approach. The major part of the work have been done using Waxman graphs; Barabasi-Albert 100 node topologies have been also evaluated. These models are discussed in Appendix A.1.

5.6.1 Results for the RMA Basic LSP Setup

The basic cases have been tested using a baseline of 10 node, and extensive testing have been conducted over a 100 node topology. First of all let us present a functional description of the basic prototype, using the 10 node topology shown Figure 5.6.

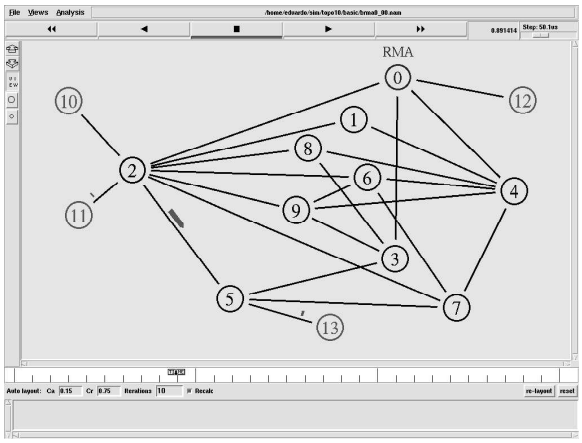

Figure 5.6: Basic RMA testing - Traffic following Shortest Path

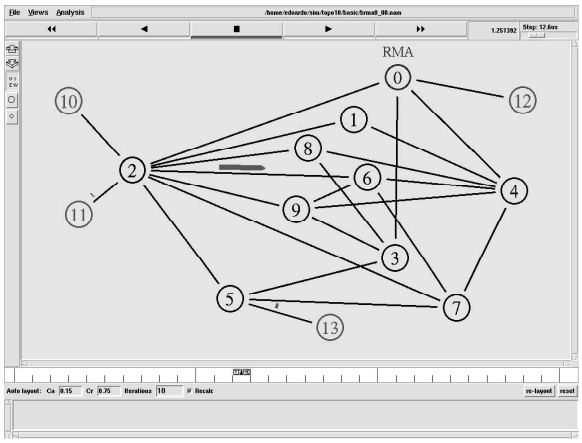

Figure 5.7: Basic RMA testing - Traffic following provisioned ERO

The *tcl* code excerpt presented in algorithm 3 redefines the standard RSVP-TE Agent upcalls to implement the basic functionality. Node n0, which is defined as the RMA in this case, captures PATH messages coming from node n2 (the Ingress LSR) and defines an LSP from n2 to the auxiliary node n11 connected to n5 (the Egress LSR) with a pre-computed ERO (n2-n6-n7-n5).

Node n2 binds the traffic flow to the established LSP.
Figure 5.6 shows the traffic from n11 to n13 (auxiliary traffic source and sink) which traverses the network following the default shortest path n2-n5. Node n0 have the RMA role as mentioned before. Once the LSP is established, as signalled by the RMA, the traffic follows the configured ERO (n2-n6-n7-n5), which is shown in Figure 5.7.
The numerical results are obtained averaging several realisations of a unidirectional LSP provisioning, varying three relevant parameters:

- Position of the RMA node in the topology
- Distance from the Ingress LSR to the RMA (in hops)
- LSP number of hops

The results for the LSP setup time varying the RMA node degree are summarized in Figures 5.8 and 5.9 for 10 and 100 node topologies, respectively. Note that the LSP setup time shows a clear dependence on the number of hops, but exhibits negligible variation for node degree on both topologies.

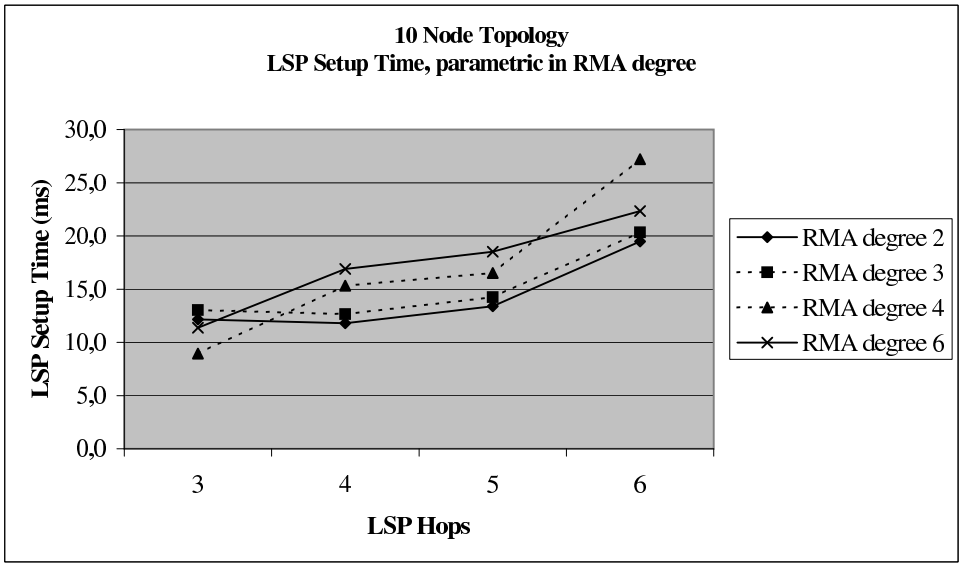

Figure 5.8: LSP setup time, parametric in RMA degree, 10 node topology

These results lead to the tentative conclusion that the position of the RMA in the node graph is not relevant to LSP provisioning time, and its main dependence is on the number of LSP hops. To further confirm this result the third variable is considered: the distance in hops from Ingress LSR to the RMA. The results for the 100 node topology are summarized in Figure 5.10.

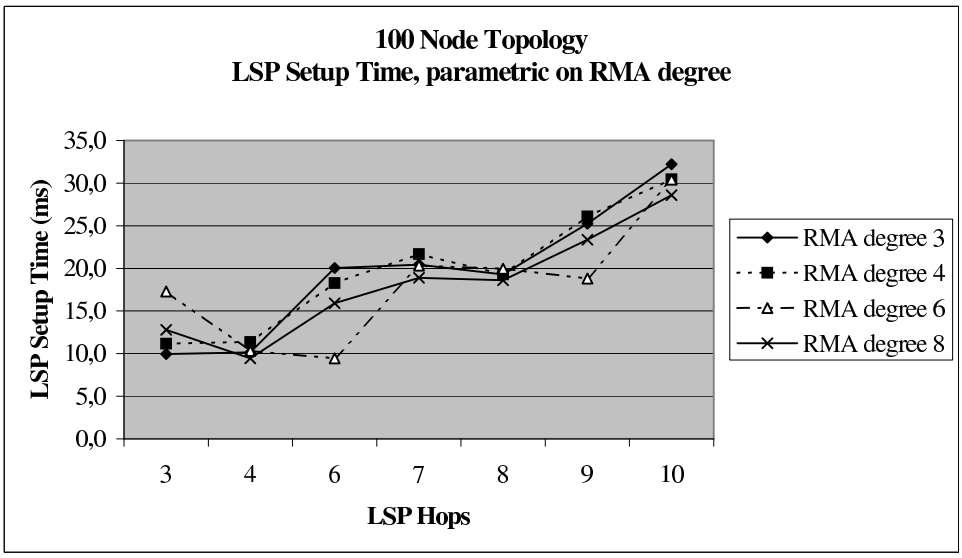

Figure 5.9: LSP setup time, parametric in RMA degree, 100 node topology

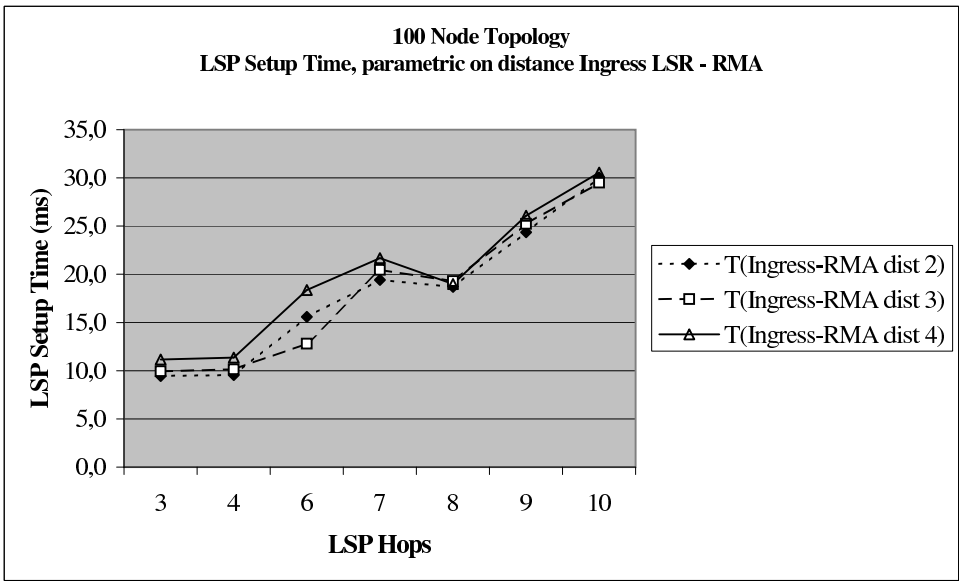

Figure 5.10: LSP setup time, parametric in distance Ingress LSR - RMA, 100 node topology

Algorithm 3 RSVP-TE upcall definitions for basic RMA

```
# Unidirectional LSP triggered by Ingress LSR request to RMA
# RMA in n0 (degree 3), Ingress n2, Egress n5 (1 hop)
# Precomputed ERO 2-5: 2_6_7_5_ (3 hops)
# Redefine the PATH upcall
Agent/RSVP instproc upcall-path { sid rate bucket sender } {
 global ns now LSRmpls2 n5 n11
 set node [[$self set node_] node-addr]
 set now [$ns now]
 if { $node == "0"} {
    $ns at $now "$LSRmpls2 create-crlsp $n11 $n5 2_6_7_5_"
    puts "intial time LSR from LSRmpls2 setup: $now"
    }
 puts "path upcall: $node time: $now"
}
# Redefine the RESV upcall to start sending a flow for
# this session as soon as the LSP is established
Agent/RSVP instproc upcall-resv { sid rate bucket sender } {
  global LSRmpls2 LSRmpls5 ns now
  set node [[$self set node_] node-addr]
  set now [$ns now]
  if { $node == "2" && $sender == "11" } {
      $ns at $now "$LSRmpls2 bind-flow-erlsp 13 2 1"
      puts "total time LSR from LSRmpls2 setup: $now"
  } else {
       puts "rsvp upcall: $node time: $now"
 }
}
#------start of simulation script here--------------
```

The independence of the RMA position in the network graph is an important result, because in practical terms it means that the network operator do not need to build a specially crafted DCN to communicate the edge LSRs with the RMA. For example, is not necessary to build a dedicated full mesh of LSPs for this purpose.

With this result in mind, it is possible to compare the performance of the three variants of provisioning using the number of hops as the relevant variable:

- Control Plane based
- Management Plane based

- RMA based

The testing for the Control Plane and the RMA based provisioning cases is similar: since the path establishment is done using the standard signalling, the difference between these two cases is the communication time with RMA added to the CBR processing time. Since an ERO is pre-computed, processing time is not significant. Note that the simulation stresses the signalling aspects of a functional prototype; CBR processing may consume non-negligible time in a general case, so the presented results are a lower bound to the RMA performance. Regarding the Management Plane based provisioning, it is assumed that the head-end (ingress LSR) takes the management role, communicating one by one with each of the LSRs along the LSP path. Therefore, the provisioning is linear with the number of LSP hops, and provides an upper bound for performance comparison. The results for the 10 and 100 node topologies are summarized in Figures 5.11 and 5.12.

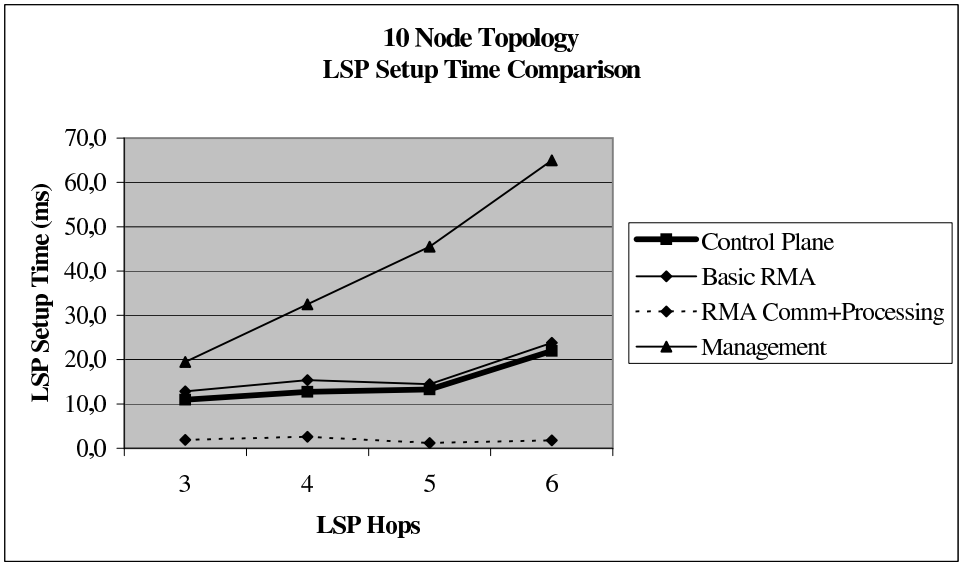

Figure 5.11: LSP Setup Time Comparison - 10 node topology

These results show that the basic RMA solution using the RSVP-TE signalling for path computation requests adds very little time to the pure Control Plane LSP setup (taking into account the warning regarding CBR computation time). Additionally, the RMA solution is independent from the RMA node degree, as mentioned. The quantitative evaluation reveals a slightly slower response of the RMA solution, when compared to the pure Control Plane provisioning time, which is the cost of an improved routing function. Also note the "halfway" nature of the RMA with respect to the Control and Management Plane.

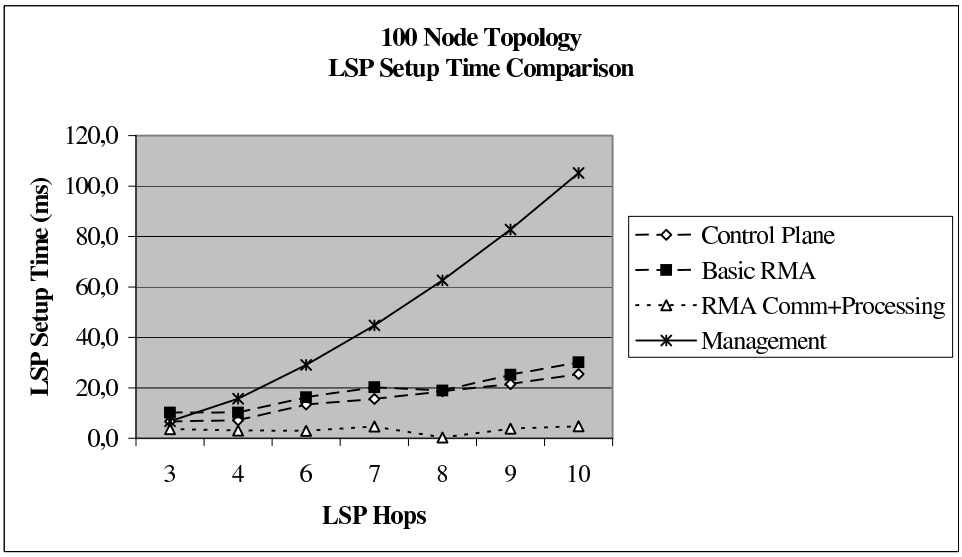

Figure 5.12: LSP Setup Time Comparison - 100 node topology

5.6.2 Results for RMA Reliable Connectivity Setup

This case introduces a new component of the prototyped solution: the COPS protocols and its agents: the Protocol Decision Points (PDPs) and Protocol Enforcement Points (PEPs), which are the basic building blocks of the Three Tier Policy Based Network Management model.

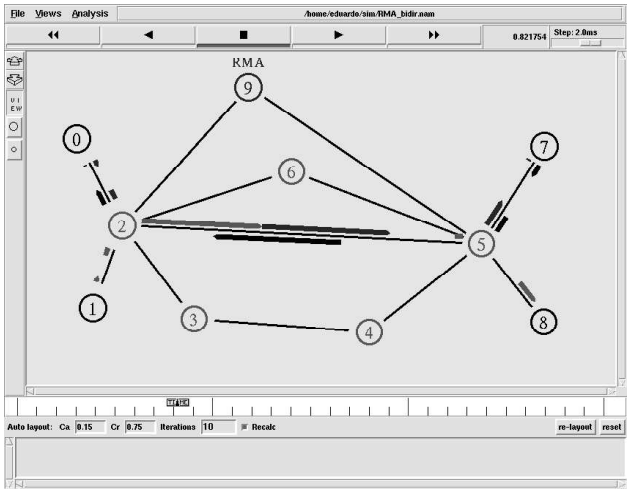

Figure 5.13: Bidirectional Connectivity Setup

The RMA assumes the PDP role, whereas the edge LSRs are the PEPs in

the network. Another important innovation of this case is that it implements the push mechanism for bidirectional connectivity provisioning.

Let us consider a simple example using a topology with five core LSRs and four clients, and an extra node which assumes the RMA (PDP) role. As seen in Figure 5.13, n0 an n7 exchange bidirectional traffic, while n1 sends traffic to n8. The clients are connected to adjacent LSRs n2 and n5; as a consequence, all the traffic is transported over the n2-n5 link. Nodes n2 and n5 have the PEP role; when n2 request a path computation to the RMA, this entity configures both the head-end and the tail-end of the bidirectional traffic (using pre-computed EROs in both cases). The result, shown in Figure 5.14 is that traffic from n0 to n7 uses the path n2-n3-n4-n5, and the traffic in the opposite direction uses the path n5-n6-n2.

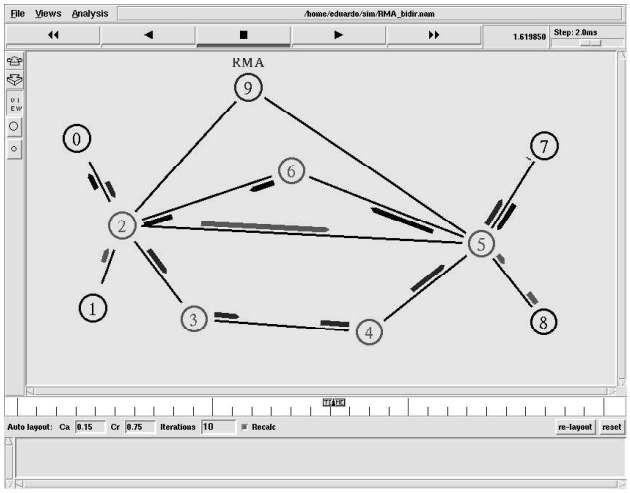

Figure 5.14: Bidirectional Connectivity Setup with diverse paths

The *tcl* code excerpt presented in algorithm 4 shows the agents definitions for this simple case. The script shows the usage of a PDP (the RMA) to setup a explicitly routed LSP to divert traffic from the shortest-path link; when the Ingress LSR (n2) request a route, a "push" route for the reverse direction is sent to the Egress LSR (n5). The traffic is assigned to the established LSRs in the RSVP upcall of the originating node, as shown in Algorithm 5.

The script triggers the establishment of the upstream and downstream LSPs using a COPS configuration request from the PEP installed in ingress node n2. The explicit routes are precomputed, as in the basic prototype. Note that the *decision* is a piece of *tcl* code that will be executed at the PEP; this shows the extensibility of the implementation, since arbitrary code may be downloaded to the network node to be executed.

Algorithm 4 COPS upcalls redefinition in ns simulator

```
# This upcall function is redefined so that as a PEP
# receives the positive response from the PDP,
# the former sends a request
COPS/PEP instproc cops-upcall-accept { } {
   $self request 8;
   # 8 means configuration information
}
# This upcall function is redefined
# so that the PDP sends decisions when
# a request from a client arrives
COPS/PDP instproc cops-upcall-request {pepid} {
   global LSRmpls2 LSRmpls5 n0 n2 n5 n7
   if {$contextR == 8} {
   if {$pepid == [$n2 id]} {
      # the decision is just a piece of the Tcl code
      # that will be executed at the PEP
      set decision1 "$LSRmpls2 create-crlsp $n0 $n5 2_3_4_5_"
      $self decision $pepid install 0 $decision1
      puts "sent $decision1 to $pepid"
      set pepid [$n5 id]
      set decision2 "$LSRmpls5 create-crlsp $n7 $n2 32 5_6_2_"
      $self decision $pepidinstall 0 $decision2
      puts "sent $decision2 to $pepid"
   } else {
      # it is the NULL decision
      $self decision $pepid NULL 0
   }
  }
}
# This upcall is redefined to analyze a decision arrived from
# the PDP and to execute the decision
COPS/PEP instproc cops-upcall-decision {decision} {
   global ns now
   # check that the PDP has returned the non-NULL decision
   if {($command == "install") && ($contextR == 8)} {
      eval $decision puts "eval $decision"
   }
   # report successful execution of a decision
   set now [$ns now]
   puts "time request/reply: $now"
   puts "PEP: $self report $handle success"
   $self report $handle success
}
```

Algorithm 5 RSVP-TE RESV upcall redefinition in ns simulator

```
# Redefine the Resv upcall to start sending a flow for this
# session as soon as a reservation is established
Agent/RSVP instproc upcall-resv { sid rate bucket sender }{
   global LSRmpls2 LSRmpls5
   ns now set node [[$self set node_] node-addr]
   # the traffic is assigned to the established LSR when
   # the RESV msg reaches the originating node "n2"
   set now [$ns now]
   if { $node == "2"} {
      $ns at $now "$LSRmpls2 bind-flow-erlsp 7 1 0"
      puts "total time LSR from LSRmpls2 setup: $now"
   } elseif { $node == "5"} {
      $ns at $now "$LSRmpls5 bind-flow-erlsp 0 3 1"
      puts "total time LSR from LSRmpls5 setup: $now"
   } else {
      puts "rsvp upcall: $node time: $now"
  }
}
```

COPS usage has been limited to Access Control configuration (e.g. queueing parameters and filters on router interfaces), but the potential usage following the PBNM guidelines is promising.

Once the basic scenario has been presented, the quantitative results can be reviewed. Again, timing information of several realisations of a bidirectional connectivity provisioning scenario have been averaged for 10 and 100 node topologies. results are summarized in Figures 5.15 and 5.16.

Note that the solution based on PDP/PEP communication using COPS for the RMA is more expensive in time than the basic approach based on redirection of the RSVP-TE messages. The overall setup time remain below 50 milliseconds, which is a very acceptable time for path provisioning. Moreover, in this case both the upstream *and* the downstream LSPs are configured simultaneously, which is a notorious advantage from the operational point of view.

5.6.3 Global evaluation of results

The presented quantitative evaluation permits to achieve in the first place a proof of concept (i.e. a prototype) of the RMA proposal. Second, the simulated environment permits to estimate the provisioning timing performance, in comparison with two extreme cases: the pure Control Plane and the pure Management Plane solutions. Both prototypes perform well in this metric,

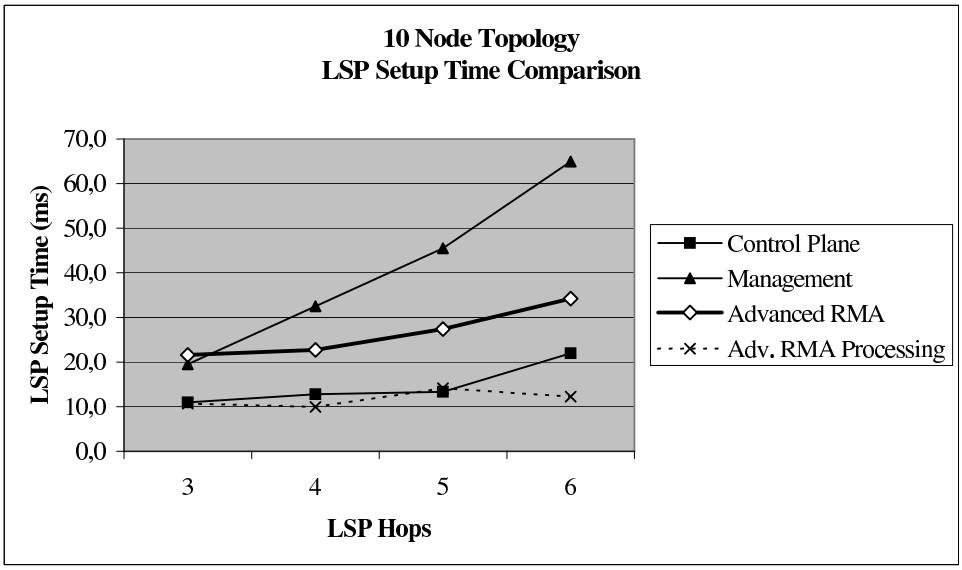

Figure 5.15: LSP Setup Time Comparison - 10 node topology

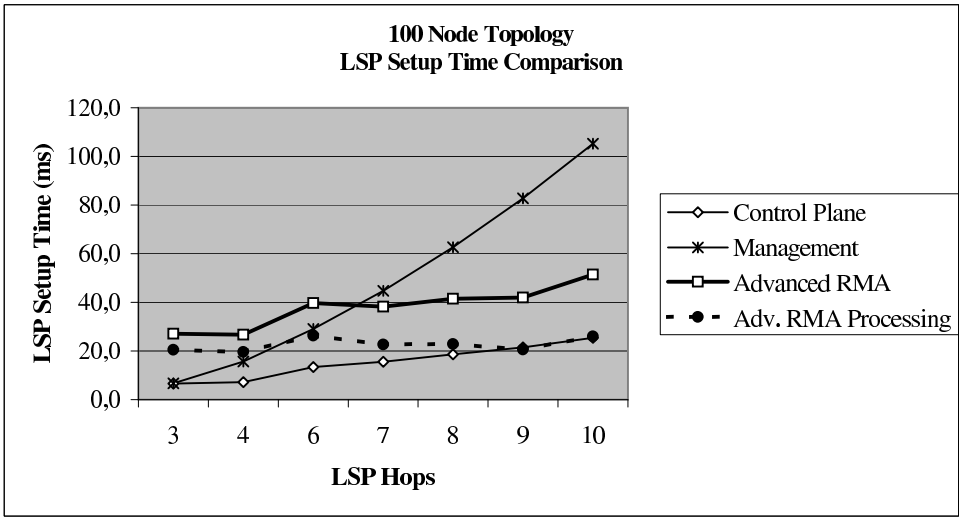

Figure 5.16: LSP Setup Time Comparison - 100 node topology

showing a "halfway" performance between both bounds. These promising results are incentives for the implementation in a real testing environment.

Besides this quantitative aspects, a practical application of the policy-based PDP/PEP paradigm has been tested.

To finish this section a summary of the timing performance of the basic and the advanced alternatives for the 100 node topology comparison (excluding the Management Plane reference) is shown in Figure 5.17. Note that the so-called "Advanced RMA" case performs bidirectional LSP setup, though a fair comparison involving "LSP throughput" per time unit would show the half of the time for this solution.

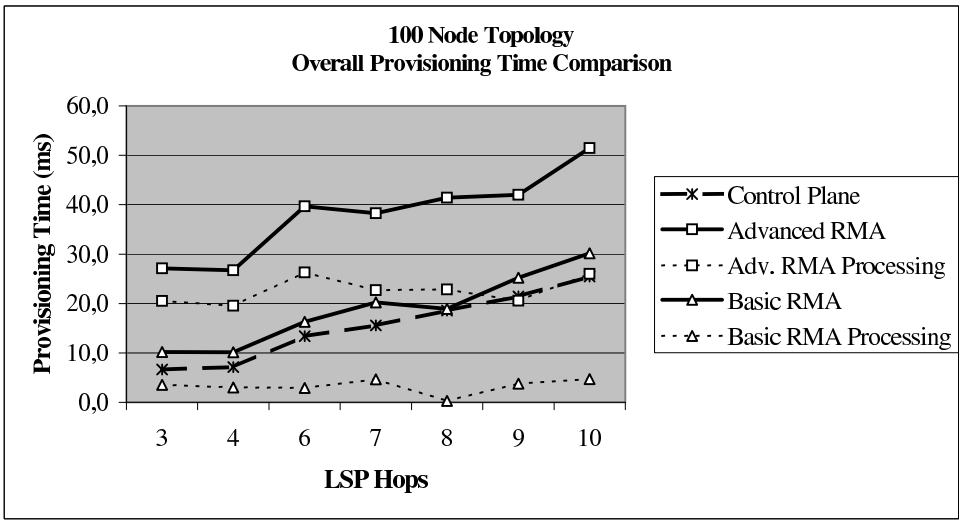

Figure 5.17: Overall Provisioning Time Comparison

5.7 Cooperation between RMAs. The Inter-Area Case

As stated in [RVB05], the current set of MPLS Traffic Engineering mechanisms has been to date limited to use within a single IGP area. The extension of MPLS TE capabilities to support inter-area resource optimization basic obstacle is given by the fact that detailed topological knowledge can only be achieved within a single IGP area for scalability purposes (i.e. IGP are hierarchical).

Two different approaches can be foreseen to provide a solution using the RMA architecture:

- a centralized solution with an (or a set of) omniscient RMA with global topological knowledge across IGP areas, or

- a distributed solution where per area RMAs cooperate to compute inter-area paths.

An important objective of inter-area traffic engineering is the protection of Area Border Routers (ABRs), which are key elements because they have detailed topological knowledge of the areas where they reside, and transport all inter-area traffic. Thus, ABRs may be considered natural candidates to host a Path Computation Server functionality. This option is the one considered in next sections.

5.7.1 Centralized Inter-Area LSP Provisioning - Omniscient RMA

This option is the natural extension of the intra-domain RMA architecture. It consist of a "know-everything", Omniscient RMA (ORMA), which handle global connectivity demands for a given set of routing areas (i.e. an Autonomous System). The ORMA has a complete topological knowledge of the network, and can solve the LSP provisioning as described for the intra-domain case. The setup procedure is depicted in Figure 5.18. The *IngressLSR* initiates a request to the ORMA, which computes a solution and issues a reply containing the computed path(s). Afterwards, the LSP (or bundle of LSPs) are established using standard signalling towards *EgressLSR*.

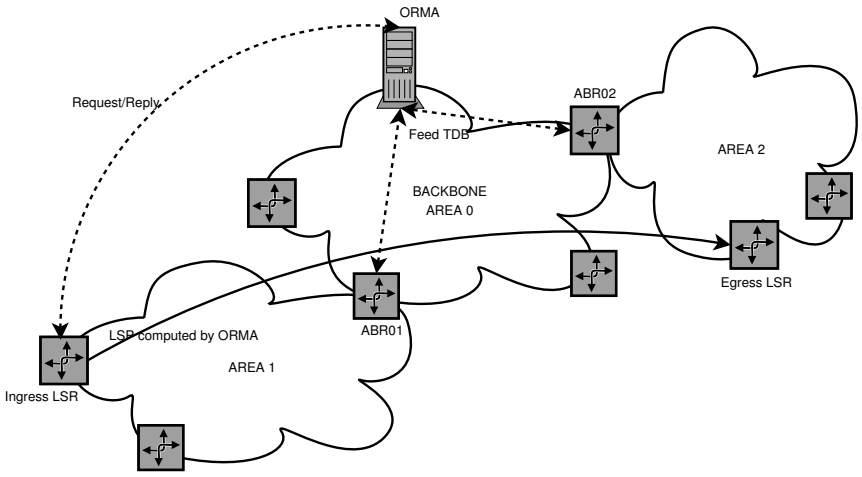

Figure 5.18: ORMA connectivity setup

Note that Figure 5.18 depicts a communication between the ABRs and the ORMA, which are IGP peers. This permits ORMA to feed its network-wide topological database. An alternative to a single ORMA is to host the functionality in the ABRs, as mentioned before. The resultant model

is that each ABR provides RMA service for it directly attached IGP area. This is straightforward assuming a two-level hierarchy, but in the case of generic hierarchical protocols such as PNNI, the levels of network topology abstraction may prevent a omniscient solution to be implemented.

A major concern is the size of the topological database that the ORMA needs to handle; the frequency of path computation requests would also be higher. Furthermore, the ORMA is a single point of failure and this issue shall be considered in detail in the implementation phase. On the other hand, global knowledge enables the ORMA to compute network-wide optimum paths. A first conclusion is that this alternative could be beneficial for inter-area connectivity setup, taking into account the aforementioned implementation issues.

5.7.2 Distributed Inter-Area LSP Provisioning - per area RMAs

In this case there is just one RMA per IGP area, with detailed topological knowledge of its area. Each RMA knows the location of other area RMAs and may cooperate with them to compute end-to-end paths. This cooperation can assume several configurations:

- Management Plane based.
- Using the request/reply protocol previously defined.
- Triggered by the Control Plane.
- Non-cooperative, in charge of the Ingress LSR.

Management Plane based cooperation

In this approach, a management overlay (the DCN in ITU-T terms) is built to communicate the RMAs and other management applications. When an area RMA receives a request for an inter-area path computation, it can determine strict hops intra-area, and loose hops for the rest of the route. Other RMAs shall be contacted to determine the strict hops to cross the unknown areas, and once an ERO is built exclusively with strict hops, it is returned to the Ingress LSR in the reply message.

Let us consider the establishment of an inter-area LSP from an Ingress LSR in Area 1 towards an Egress LSR in Area 2, as shown in Figure 5.19. The sequence of LSP setup using the RMA is the following:

1. The Ingress LSR in Area 1 demands a route computation to RMA1, which constructs an ERO with strict hops in Area 1 towards the ABR, a loose hop between ABRs in the backbone Area 0 and another loose hop to destination in Area 1:

IngressLSR, IntermediateLSR11, IntermediateLSR12,...,ABR01, ABR02:loose, EgressLSR:loose

2. RMA1 request RMA0 and RMA2 to specify the loose hops with the given constraints. It would receive the partial responses:

   ABR01, IntermediateLSR01, IntermediateLSR02,...,ABR02, EgressLSR:loose

   from RMA0 and

   ABR02, IntermediateLSR21, IntermediateLSR22,..., EgressLSR

   from RMA2

3. RMA1 construct a global ERO with strict hops and send it to the Ingress LSR,

4. which proceeds with LSP setup as usual.

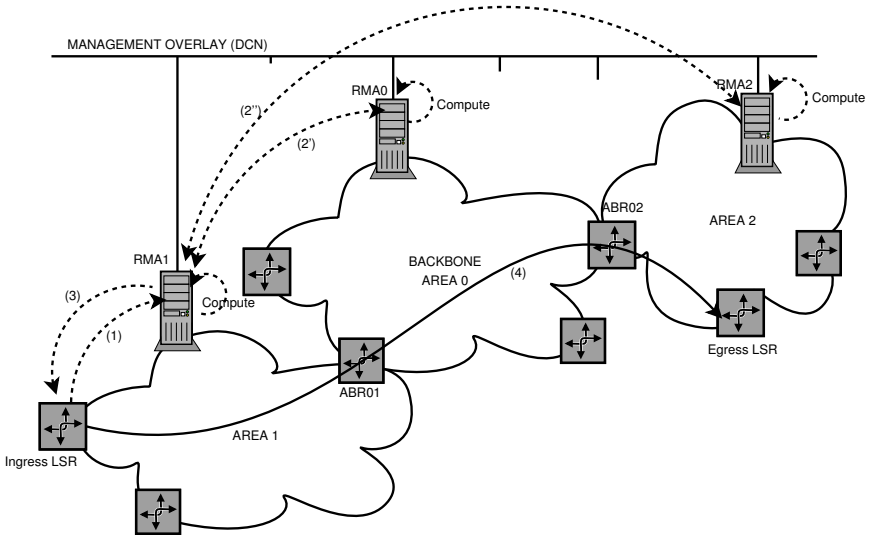

Figure 5.19: Management Cooperation for Inter-Area LSP Setup

Cooperation using the request/reply protocol

This case is basically identical to the previous one, the only difference is the communication protocol. The requested RMA shall contact its peers to build a proper end-to-end path to reply to the Ingress LSR.

Inter-Area LSP provisioning triggered by the Control Plane

In this case the RMAs are unaware of each other, and they're used by the LSRs as needed. If we consider the same example as before, the sequence of LSP setup is the following:

1. The Ingress LSR in Area 1 demands a route computation to RMA1, which constructs an ERO with strict hops in Area 1 towards the ABR, a loose hop between ABRs in the backbone Area 0 and another loose hop to destination in Area 1:

   IngressLSR, IntermediateLSR11, IntermediateLSR12,...,ABR01, ABR02:loose, EgressLSR:loose

2. The computed path is sent to the Ingress LSR, which initiates a setup procedure issuing a PATH message downstream towards the Egress LSR.

3. When the PATH message reaches the ABR01, loose hops must be determined, and demands a route computation to RMA0. Once computed, the ERO is transformed in:

   IngressLSR, IntermediateLSR11, IntermediateLSR12,...,ABR01, IntermediateLSR01, IntermediateLSR02,...,ABR02, EgressLSR:loose

4. The PATH message progresses through Area 0.

5. When the PATH message reaches the ABR02, it demands a route computation to RMA2 to solve the remaining loose hop. Once computed, the ERO is completed:

   IngressLSR, IntermediateLSR11, IntermediateLSR12,...,ABR01, IntermediateLSR01, IntermediateLSR02,...,ABR02, IntermediateLSR21, IntermediateLSR22,..., EgressLSR

6. The PATH message reaches the Egress LSR using the ERO strict hops, and

7. the RESV message travels upstream to the Ingress LSR, establishing the TE-LSP as usual.

Note that no cooperation is directly achieved and each step is basically an intra-area case; anyway this is a case very likely to be deployed in the absence of a management overlay or with incomplete implementations of the request/reply protocol.

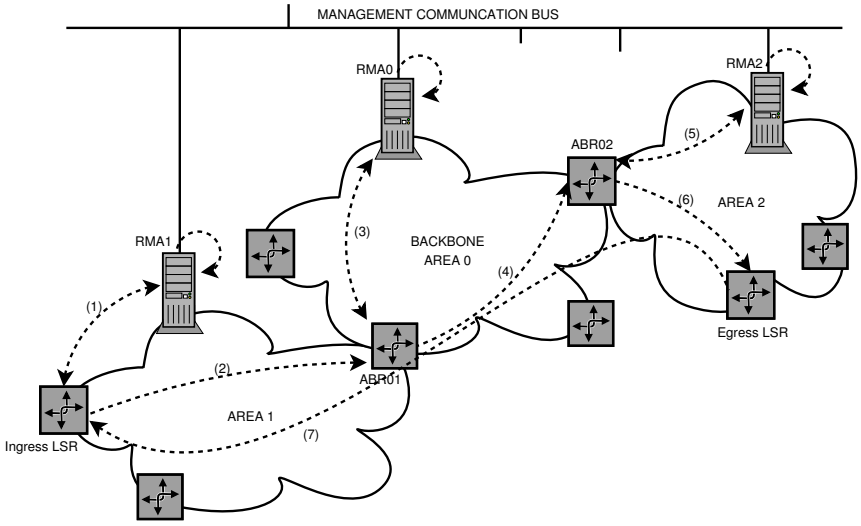

Figure 5.20: Inter-Area LSP Provisioning triggered by the Control Plane

Non-cooperative Inter-Area LSP setup

In this case the RMAs are also unaware of each other, and the Ingress LSR is responsible to request the partial computation needed to build an end-to-end path loose hop free. This case is not interesting since no cooperation is achieved and each step is basically an intra-area case.

It worths noting that inter-area LSPs traverse the ABRs, which are candidate points of failure. The existence of alternative paths between routing areas (i.e. diverse ABRs) is a matter of network design (the larger timescale in the Traffic Engineering control loop, see Section 3.2). This means that the RMA capability to provide alternative paths is constrained by the underlying topology. The inter-area provisioning policies shall prioritize load balancing and path diversity among ABRs, in order to provide preventive protection to configured LSPs.

5.8 Evaluation of inter-area cases

The Omniscient RMA approach to inter-area provisioning is basically identical to the single or intra-area case, with some potential complex implementation issues such as the size of the Traffic Engineering Database, fault-tolerance, inter-area visibility, among others. From a prototyping point of view it is indistinguishable from the already studied cases, if the mentioned hazardous issues are left aside of the implementation.

On the other hand, the distributed case presents some peculiarities (and several alternatives, as described in the previous section). A first issue to con-

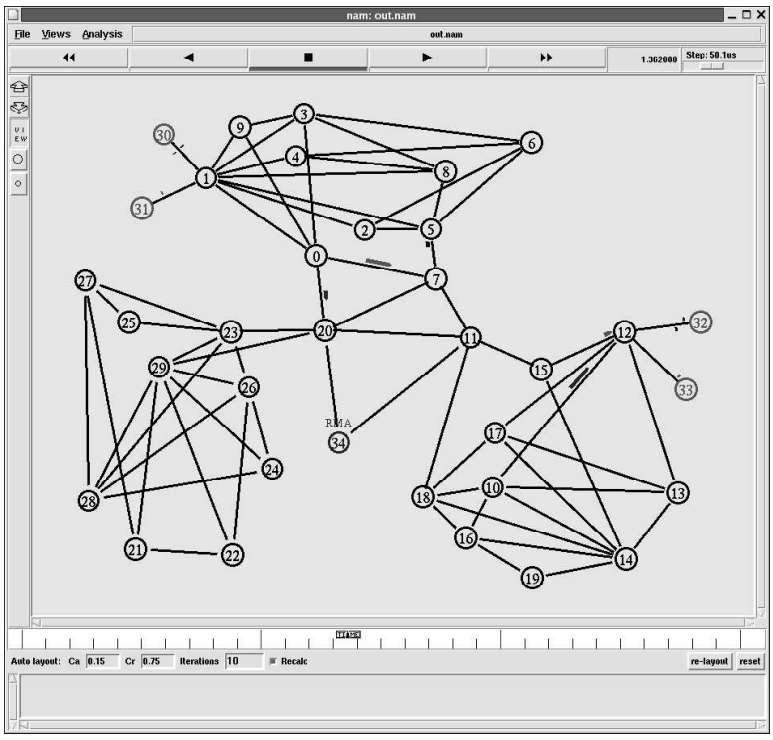

Figure 5.21: Omniscient RMA scenario

sider when crafting a simulation environment is to build a valid hierarchical topology. The inter-area cases have been tested using two-level hierarchical topologies, resembling OSPF routing structure, with stub areas connected to a backbone area. Both figures 5.21 and 5.22 show a three area topology (plus the backbone), namely Area 1, Area 2 and Area 3, and the backbone Area 0, comprised of ten nodes each.

A number of realisations of this scenario has been executed for the Omniscient RMA, realising that the behaviour is the same as in the intra-area case. In these tests the RMA has been located in the backbone area 0, as shown in Figure 5.21.

Regarding distributed/per area RMAs, basic functional testing was done for the case triggered by the Control Plane. The fundamental building block of the solution is to catch the RSVP PATH upcall in the nodes directly connected to the distributed RMAs.

Note that additional nodes have been added for clarity to play the RMA role, but any network node could be defined as a PDP, though running RMA functionality. This upcall redefinition is shown in Algorithm 6, and the network scenario is shown in Figure 5.22.

Algorithm 6 RSVP PATH upcall redefinition for Distributed Inter-area RMA

```
Agent/RSVP instproc upcall-path { sid rate bucket sender } {
   global ns now n11 n27 pdp0 pep(n11) pep(n20)
   set node [[$self set node_] node-addr]
   set now [$ns now]
   if { $node == "11"} {
      set pep(n11) [$n11 add-cops-pep]
      $pdp0 connect $pep(n11)
      $ns at $now "$pep(n11) open"
      puts "open pep(n11): $now"
   } elseif {$node == "27"} {
      set pep(n27) [$n27 add-cops-pep]
      $pdp0 connect $pep(n27)
      $ns at $now "$pep(n27) open"
      puts "open pep(n27): $now"
   } else {
      puts "path upcall: $node time: $now"
   }
}
```

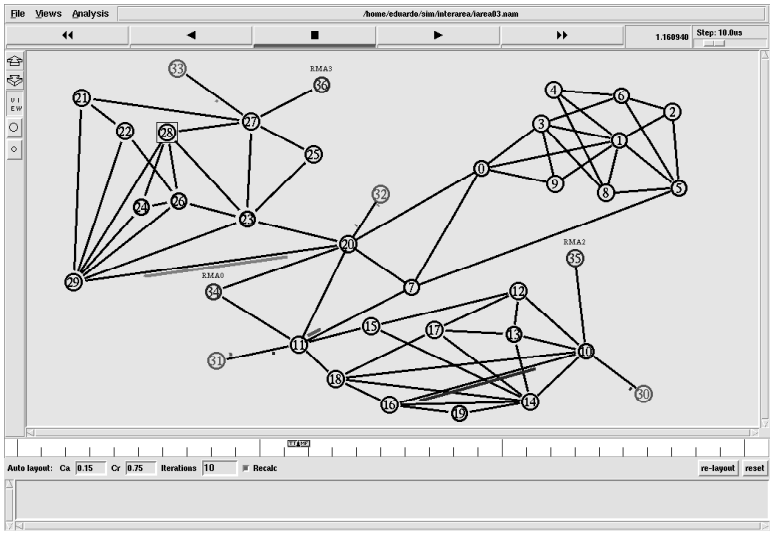

Figure 5.22: Distributed RMA scenario

The conclusion for the inter-area testing is that timing performance is identical to the intra-area case for the Omniscient RMA, and it takes more time in the case of distributed RMAs, shown by the functional testing per-

formed for this case. The latter approach admit several sub-cases, which cannot assure that the end-to-end path for a given inter-area LSP is optimal, due to the hierarchy of the underlying routing protocol.

5.9 The RMA as an Offline Traffic Engineering tool

As defined in section 3.4.2, offline CBR computes a global optimum allocation of traffic demands taking into account the system of constraints and the available resources, under a given objective function, which express the network administrator policy. The RMA is a Path Computation Server, and its functionality can be used by the Control Plane mechanisms, as stated in previous sections, and also as a TE component, driven by Management applications. This functionality is achieved using the aforementioned RMA Management Plane interface.

The RMA can be routinely used as an optimization tool, when deviation from the global network TE objectives are verified, or when a pre-defined cycle time is completed (i.e. every other week). This would initiate a new TE long-term control loop, referring to the model depicted in Section *3.2.*

The pseudo-code for this optimization loop follows:

Algorithm 7 RMA Global Optimization

```
while TRUE do {
  time=0;
  global_objectives=TRUE;
  while[(time<CYCLE_TIME)&&(global_objectives==TRUE)]{
    time++;
    global_objectives=Check_Global_Objectives();
  }
  Global-Optimization();
}
```

The *Check_ Global_ Objectives();* procedure comprises a monitoring functionality in the network, and a processing of performance information for comparison against certain thresholds. Information from customer-care processes can also be incorporated into the correlation process. The algorithm suggests that the result of this process is a boolean, but more accurately it should be thought as a rate or index of accomplishment of network objectives (i.e 99,999% of network availability). According to the SPs policies, a simple threshold function can be constructed to generate a boolean value, that is to decide if global objectives are fulfilled or not. For example a satisfaction rate equal or greater than 90% could be defined as "TRUE" while values below 90% would be "FALSE". The construction of such performance indexes

in operational environments is very complex and is out of the scope of this work.

Once a global optimization round is complete, and a complete mapping of demand to resources is computed (i.e. there is a global definition of the needed LSPs to satisfy the demand), it is time to re-configure network nodes accordingly. This is a challenging task, since the obvious option of tear-down every connection and run a complete re-configuration is not possible to undertake in operational environments. The rational option is, (1) to use an objective function that tends to minimize network reconfiguration as used in [Bek04], and (2) to use a make-before-break strategy to reconfigure the LSPs. The configuration is driven by the Management Plane using the standard Element Manager interfaces.

Another useful application of the RMA for a Service Provider is as a planning and design tool. The CBR capabilities of the RMA can be used to compute the effects of a planned demand. In this case no configuration action is taken, and different objective functions can be used.

5.10 RMA Architecture and Functional Components

The functionality of the RMA and associated requirements have been analyzed and evaluated in different scenarios throughout the previous sections of this chapter. Now is possible to review a comprehensive architecture that comply with such requirements, that would assist a practical implementation.

The main architecture components are the following:

- Routing and Management Agent (RMA): this is the Path Computation Server, which provides the path computation functionality.
- Path Computation Clients (Ingress LSRs): they request path computation from the RMA, using the Control Plane interface.
- Management applications: these are software components which may have the client role towards the RMA, using the Management Plane interface.
- Four Communication Protocols have been identified:
  - a request/reply protocol between LSRS and RMAs (COPS, PCEP).
  - a protocol to communicate the RMA with the management applications (the former request/reply protocol is a valid alternative).
  - a configuration protocol between the client routers or management applications and the LSR (COPS, SNMP, CLI..).

– RSVP-TE signalling protocol for path establishment.

Apart from these architectural components, other basic functionalities are:

- Network topology gathering: RMA is an IGP peer.
- Network monitoring.
- The selection of objective functions: these must be built-in the CBR engine, and could be modified in a policy-based manner, meaning that a careful design of the CBR engine may permit to change behaviour in execution time. This is also useful for the definition of administrative policies, which are used as constraints in the path computation process.

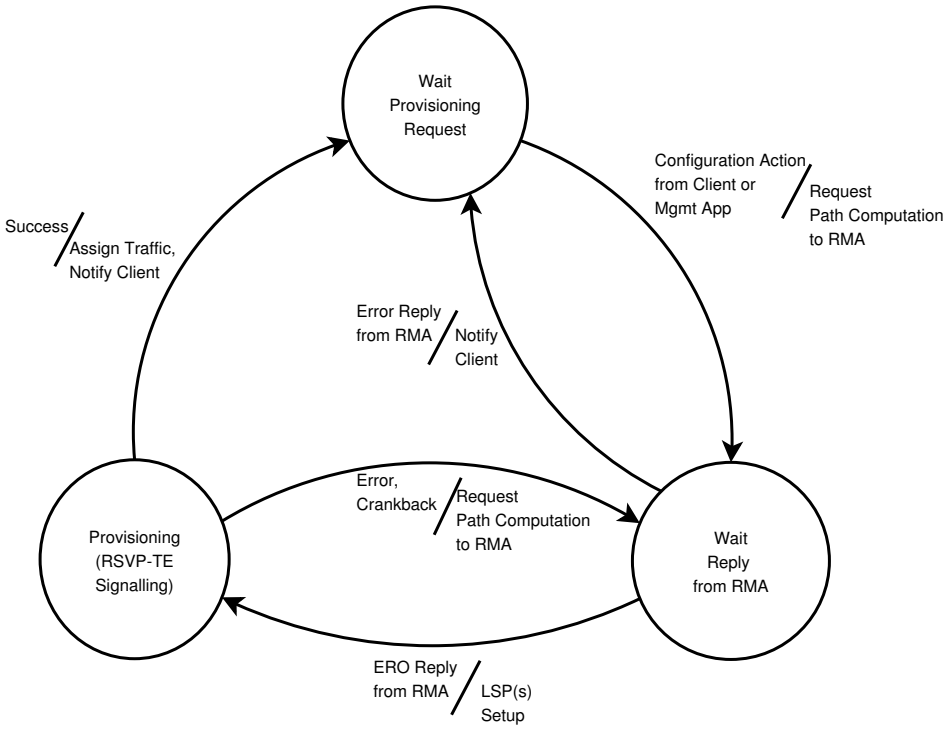

Figure 5.23: Ingress LSR Provisioning FSM

The major actors of the architecture are the Ingress LSRs and the RMA. Their basic functionality regarding connectivity provisioning can be described by the Finite State Machines depicted in Figures 5.23 and 5.24.

Note that the Ingress LSR Provisioning FSM is an hybrid management-control plane component, which runs concurrently with the LSR data plane, i.e. with the main routing and traffic forwarding functions that are always

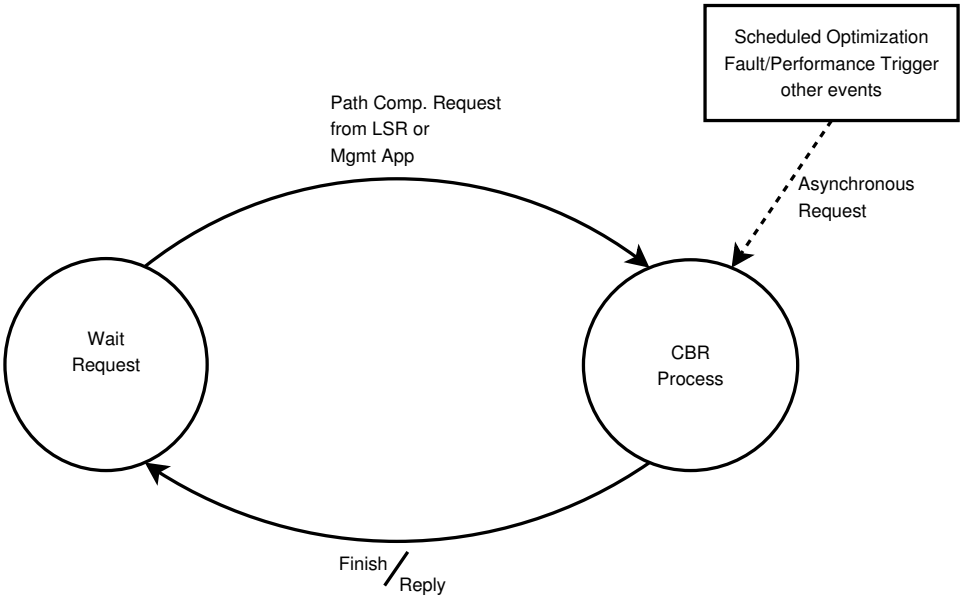

Figure 5.24: RMA Provisioning FSM

present (not part of the presented FSM). Also note that the diagram is identical to the FSM for the pure Control Plane based provisioning depicted in Figure 3.12, but the path computation functionality have been transferred from internal CSPF to external RMA CBR.

Regarding the RMA Provisioning FSM, note that besides the requests from LSRs and management applications, CBR computation may be triggered by asynchronous events such as the violation of a performance objective threshold, as stated by Algorithm 7. The CBR Process is profiled by the attributes of the request (Ingress, Egress, QoS and administrative constraints, Protection and Load balancing Options), and the network policies stored in the Traffic Engineering Database, as will be described in next section.

5.10.1 RMA components

The RMA is built using a component-based framework, which provides basic scheduling and other supporting components needed to build the described functionality. The interfaces and "core" components shown in Figure 5.25 are described below:

Signalling Interface

This interface implements the Request/Reply signalling (i.e. the COPS

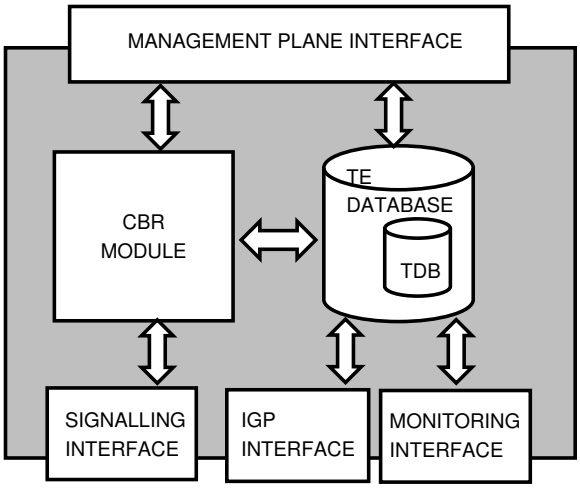

Figure 5.25: RMA Functional Components

protocol), maintaining sessions with the nodes requesting path computation, and with the CBR engine.

IGP Interface

This interface implements the communication at the routing protocol level, running OSPF-TE [KKY03] or ISIS-TE [SL04] (depending on protocol deployed in the network). This interface performs network topology and state gathering, to maintain the Topology Database (TDB), which is part of the Traffic Engineering Database (TE-DB), the basic information source for CBR computation. Besides its participation in the IGP, the RMA could also implement a monitoring interface (i.e. using SNMP) to gather information not provided by the IGP, as proposed in *[CMM+03]*. The design of the TE-DB is vital in order to speedup CBR computation with minimal inaccuracies.

Monitoring Interface

This interface collect statistical information useful for performance management functions and as a complement of the network states gathered by the IGP Interface. Usual passive and active (i.e. using probes) monitoring techniques using SNMP, Netflow or RMON, as described in Section 3.3.3, are applicable.

Management Plane Interface

This component implements the interaction with management applications, which enables the RMA to be used as a Traffic Engineering component for high-level applications. Besides this, network policies are fed to the RMA using this interface. This interface may be based in well-known distributed

component frameworks like CORBA, widely used by telecommunication operators, and/or J2EE, .NET or other frameworks in use in the enterprise and Internet environments.

CBR Computation Component

This is the core of the RMA, which provides the intended functionality: a computation engine for Constraint-Based Routing. The component implements the needed algorithms to solve the Path Computation problem with multiple restrictions. Well-known algorithms and heuristics can be used to accomplish the intended goal, as described in Section 3.4.2, making sure that route computation time is limited (i.e. by the usage of polynomial-time CBR algorithms).

Traffic Engineering Database Component

The TE-DB contains the up-to-date information regarding link states in the network, gathered by the IGP Interface. Additional information, like constraints and administrative policies are also persistent in the TE-DB. This information, which defines the TE objectives of the network, will typically come from Policy-Based Management applications.

5.10.2 Implementation issues

The RMA is designed as a component-based system, as detailed in the previous section. Several issues arise when the implementation phase is undertaken:

1. High availability, fault-tolerance, stability of the solution.

   The proposal in this regard is to eliminate potential bottleneck and provide fault tolerance by means of:

   - Two tier server design: use a reliable communication front-end with a computation back-end cluster. The design is shown in Figure 5.26. This is a proven architecture used by Service and Content Providers for high-availability services such as web-server farms, VoD head-ends, E-Mail distributed servers. In the figure there are two front-end sets, one to handle Control Plane communication, and the other for the Management Plane. This separation, while not mandatory, is advisable given that very different kind of protocols need to be supported.
   - High availability is given by two factors:
     (a) arbitrary large set of front-end (i.e. signaling) processors and,
     (b) arbitrary large set of computation nodes in the back-end cluster.

The remaining point of failure is network connectivity (both internal and external). Internal connectivity (i.e. between front and back-ends) can be protected by redundant LAN switches, while different options exist to overcome potential external connectivity failure. A straightforward (and expensive) solution is to place disjoint RMA clusters in the network, while an acceptable solution is to have a multi-homed approach, i.e. with multiple load-sharing links. Other useful techniques include VRRP [Hin04] and DNS Round-Robin, among others. This type of distributed architecture has been implemented in operational environments as described in [MRG03].

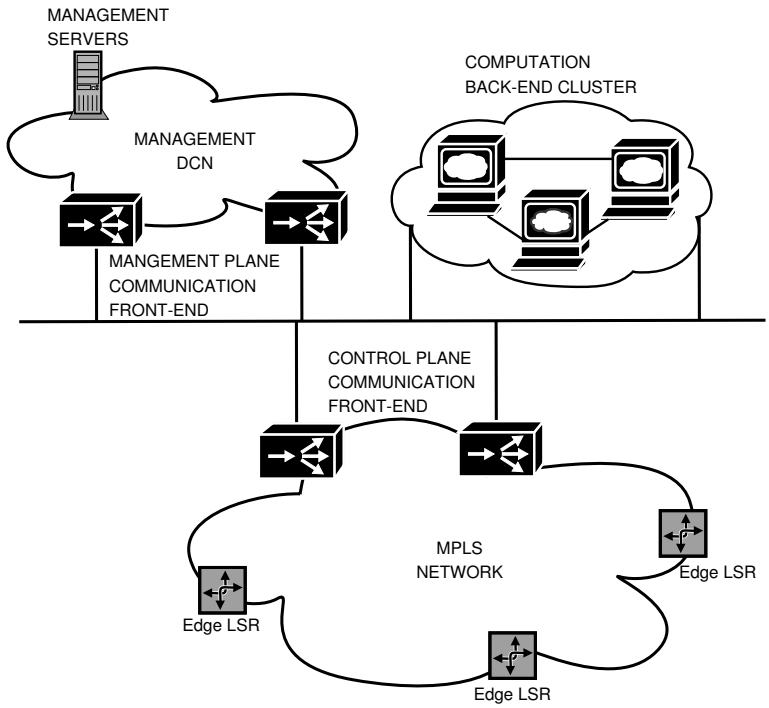

Figure 5.26: RMA cluster architecture

2. Autodiscovery of RMAs

Once installed in the network, the LSRs need a way to know where the RMA(s) is(are) located. The simplest way to accomplish this task is to rely on LSRs' local configuration, and avoid any autodiscovery mechanism (*static* solution). This solution is very simple, but needs configuration from the management application. The second option is to implement an autodiscovery functionality, similar to DHCP. This solution is robust since RMA IP address can change easily, RMAs can come and go without service disruption (*dynamic* solution). The cost

of this solution is the implementation of yet another protocol both in the LSRs and the RMA.

For the time being the static solution is chosen. Since the RMA architecture is redundant and fault-tolerant, this is considered a minor drawback.

3. Traffic Engineering Database (TE-DB)

   The construction of the TE-DB involves two asynchronous processes:

   (a) update of Topology Database (TDB) by the IGP,

   (b) policy and administrative information insertion from management applications.

   As stated in [MB03] and [AGKT99], the topology database (TDB) suffers intrinsic inaccuracies, due to the update mechanisms of the IGPs. Both proposals may be implemented to reduced the gap between the gathered TDB and the actual network state. This, consequently, will reduce the blocking probability and the need for crankback procedures in provisioning time. Moreover, the implementation of such inaccuracy-reduction mechanism involves changing the update policies/thresholds in every network node (i.e. IGP code has to be modified).

   A possible solution, built using techniques borrowed from classical data-base technology is to implement an ad-hoc Two Phase Commit (2PC) algorithm to conciliate stored and actual network information. In [GBS03] the management framework plays the coordinator role, whereas the network devices are the cohorts of the 2PC protocol. Initial implementation work has been conducted, using a policy-based, object oriented approach. Nevertheless, the cornerstone of the implementation is to achieve that the proprietary device agents (or element managers) behave as expected by the protocol; some kind of proxy has to be implemented, compromising the timing aspects of information conciliation process.

4. The CBR process

   As mentioned in Section 3.4.2, there are many heuristics and a few exact algorithms to solve the CBR process in near real time. The implementation shall evaluate the applicable approaches to the RMA, taking into account the objective functions, the system of demands, network and administrative constraints that need to be satisfied.

   The RMA uses simple routing policies to assist the CBR process; for example, a simple policy is to deny the establishment of LSPs with

bandwidth greater than a certain value, to tune the load sharing of traffic demands and minimize blocking probability.

Since the objective of this thesis is centered in signaling and management aspects of connectivity provisioning, the CBR problem is considered solved for in the evaluation of the proposed solution; in fact, the RMA database is fed by a precomputed solution (i.e. a set of EROs that satisfy the simulated demands).

5. Other issues

   (a) The distributed component environment has been extensively tested in previous work. Both IST WINMAN Project and the Multiservice Metropolitan MPLS Network Project (RMS, reviewed in Appendix A.2) software components are built using JacORB [Bro05] and TeleManagement Forum MTNM NML-EML information model (using connectivity objects such as the SubNetwork Connection (SNC), CrossConnection, Connection Termination Point (CTP), among others). MPLS extensions for such model has been developed, e.g. mapping the SNC object to a Bidirectional Traffic Trunk (BTT) and CTPs to labels.

   (b) The monitoring functionality applied to performance management was also implemented in the aforementioned projects, using SNMP passive and active monitoring techniques. In particular the functionality of the probes defined in the RTTMON MIB [Met05] has been implemented and tested. Also, Netflow collecting tools have been tried.

   (c) Stateful or stateless RMA?

   A stateless engine would rely on real time available information, whereas a stateful server may recall previous requests to determine for example if two LSPs are correlated and in check if they belong to an SRLG when computing path diversity, for instance. Moreover, a stateful server, and its associated database would be very valuable for a network operator, for performance management and billing purposes. This issue should be carefully considered by a real world implementation.

5.11 Conclusions

The complete list of issues arisen in this chapter discussion and its proposed/implemented solutions alternatives are summarized in Table 5.1:

Requirement	Proposed Solution	Implementation
CBR offline, arbitrary objective functions and constraints supported	CBR implementatio / Library of solvers	Planned
CBR online near real time	HPC approach	Planned
Management Plane integration	Standard interface, open information model	Tested (previous work)
Flexibility	Policy-Based	Tested (client side)
Standard Interfaces	Signalling, IGP, Mgmt	Tested
Auto-discovery	Non supported	Static
Protection	Buil-in the CBR process	Tested
Fault Tolerance/Recovery	Cluster, redundancy	Planned
Topology	Participating in IGP	Tested

Table 5.1: Capabilities of the RMA solution

The main advantage of this proposal is a timely LSP setup provided by the Control Plane, and the CBR accurate computation provided by the Management Plane, which has been prototyped in a simulated environment. The quantitative evaluation reveals the hybrid nature of the proposal, showing a "halfway" timing performance between the pure Control Plane solution (the lower bound) and the pure Management Plane solution (the upper bound) for LSP establishment. As a measure of the timing performance, it meets the 50 milliseconds threshold imposed by the SDH recovery mechanisms.

Bidirectional connectivity with path diversity establishment driven by the ingress LSR has been proposed and justified, using a push mechanism provided by the policy-based nature of the COPS protocol. Moreover, this mechanism reduces the provisioning cycle, triggering connections directly from the RMA, upon reception of a request either form a client or a management application.

An interesting side result, important in operational environments, is that

the location of the RMA in the network is not relevant regarding the timing performance. In practical terms this means that there is no need to deploy a special DCN to reach the RMA functionality.

Inter-area scenarios have been explored, showing that is possible to achieve the same performance as the intra-area case, at the price of a bigger Topological Database, for the Omniscient RMA case. The applicability of this approach depends on the size of the network as a whole.

Aspects to be evaluated:

- It has been shown that the RMA architecture can be inserted in the Traffic Engineering control loop as an optimization tool. Its functionality as a load balancer and network recovery mechanism need to be evaluated, and highly depends on the implementation of the CBR algorithms and the network policies. A tool that helps network operator to design good routing policies would be a major achievement.
- Since the CBR functionality is hard-coded in the prototype, it is has not been possible to evaluate the *quality* of the solution in terms of the resource allocation problem.

Chapter 6

Discussion of Inter-domain Provisioning

6.1 Introduction

The Internet is a complex interconnection of networks organized in Autonomous Systems (AS) without a centralized administration. Information flow relies on the existence of transit traffic agreements between connected ASs. An average communication over the Internet traverses several ASs from end-to-end. Therefore, connectivity can be broken if any intermediate AS denies transit for some reason (inexistence of agreement with end-systems, misconfiguration, etc). Furthermore, if Quality of Service (QoS) constraints must be met, the problem becomes hard to solve.

Routing in the Internet has two instances, namely intra and inter-domain. Inside a single administrative domain an Interior Gateway Protocol (IGP) is used to distribute topological information and build each router forwarding table. IGPs can roughly be classified in distance-vector and link-state routing protocols. The latter is chosen by most operators to distribute the entire network topology to all routers and select the shortest path according to some administrative metric. OSPF and IS-IS are examples of such link-state routing protocols in use in the Internet. On the other hand, inter-domain routing is handled by the Border Gateway Protocol (BGP), which is the de-facto standard to distribute reachability information across administrative domains, and to select the best route to each destination according to local policies specified by each domain administrator. Inter-domain routing only deals with the interconnection information, preventing the distribution of detailed topological knowledge to other domains, due to scalability and commercial reasons.

BGP based techniques have been in use for many years, suffering from inherent limitations for QoS routing across administrative domains. Innovative proposals try to build end-to-end peering relationship in order to improve

inter-domain Traffic Engineering (TE) capabilities. In this respect is worthy to mention MPLS, successfully used for this purpose.

This section considers the usage of the RMA to assist inter-domain routing and provisioning with Quality of Service (QoS) and policy constraints. The RMA inter-domain architecture follows the idea of establishing end-to-end peering relationships with TE capabilities, building MPLS-based tunnels across the Internet.

6.2 Classical Inter-domain Traffic Engineering

BGP is a path-vector protocol that works by sending route advertisements. A route advertisement indicates the reachability of a network prefix (IP address/subnetwork mask). A BGP router on a given AS will advertise routes to networks that belongs to the considered AS, and also to networks that have been advertised from another AS, if a transit agreement exist between both ASs. At this point it worth noting that there are two main relationships between ASs in the Internet. A customer-to-provider relationship is established when an AS (the customer) buys connectivity from a provider, while a peer-to-peer relationship occur between ASs that share the cost of a common link (see [Gao01] for a complete reference on AS relationships). Usually, an AS accepts to redistribute the routes learned from its customers to every neighbour, and the routes learned from its peers and from its providers to its customers. Nevertheless, it will not redistribute routes between its peers and its providers. This give us a hint of a crucial aspect of BGP-based inter-domain routing: policies applied by each AS affect end-to-end route advertisements in an unpredictable way while they traverse the Internet, affecting in turn the BGP decision process.

Another basic aspect to consider is the hierarchical structure of the Internet topology. The top level of the hierarchy (the core) is composed of a small number of large transit AS called Tier-1 Internet Service Providers (ISPs), connected in a full-mesh of BGP sessions. At intermediate levels, smaller transit ISPs are well connected together and use Tier-1 ISPs and upper level ISP as providers. The bottom level of the hierarchy, the stub AS, is composed by small, regional ISPs and large enterprise networks. A vast majority of the stubs ASs are multi-homed (connected to two or more providers) and do not provide transit. Furthermore, it has also been shown that most destinations are no farther than a few AS hops (4-6), and that a significant portion of the traffic of a typical stub AS is carried by a very small number of invariant AS-paths (meaning that these paths are stable, permanently present in the BGP tables).

Given that end-to-end traffic flows between pairs of stub ASs through the core, it is important to consider the techniques used to control traffic in a stub AS. First of all it is worthy to point out that while it is quite feasible

to control outgoing traffic, it is very difficult to control incoming one, due to the routing policies and filters applied by upper layer ASs. Most techniques to control incoming traffic rely on explicit inter-AS cooperation agreements.

Concerning the control of the outgoing traffic, `local-pref` attribute is used by many ISPs to choose the best route for a given destination. If an ISP receives full-routing from its upstream provider, the size of the BGP routing table would make the setting of this local-pref attribute a very difficult combinatorial problem [Pel03]. In practice, heuristics to setup `local-pref` and other attributes based on traffic load (and other parameters like end-to-end delay) measurement for a subset of destinations can perform very well.

Regarding the optimization of incoming traffic, a first approach is to announce different route advertisements on different links, based on the measured load of the incoming traffic per announced prefix. An obvious disadvantage of such method is that if any link fails, the prefixes announced over the affected link become unreachable.

Another method, based on the fact that the `AS-Path` length is a kind of hop metric, is to rank route advertisements on different links artificially increasing the length of the `AS-Path` attribute. This technique is called `AS-Path` prepending, and is often used by stub multi-homed ASs. Existing studies and operational practice show that `AS-Path` prepending is not appropriate to efficiently control the incoming traffic flow.

6.3 MPLS approach to Inter-domain Traffic Engineering

This section reviews some proposals that make innovative usage of BGP and QBGP (BGP with QoS extensions) [LKSJK02]. QoS extensions insert bandwidth and delay information (dynamic QoS metrics) into BGP advertisements, based in available bandwidth measurement. While the static metrics are constant, such as the link capacity and the AS hop count, the dynamic metrics vary according to different traffic load, such as the available bandwidth of a link or path. Given the fact that end-to-end (i.e. between stub ASs) QoS assurance is uncertain with "plain BGP routing", the natural step is to establish some sort of end-to-end tunnelling in order to circumvent the problems generated by the transit through the core of the Internet.

The overlay QoS Routing framework presented in [YFMB+04] installs Overlay Entities (OE) in peer stub ASs, and maintains an overlay routing control based on QoS measurements, in order to cope with a given Service Level Specification (SLS) for traffic exchange. Reacting to SLS violation, the overlay routing control reconfigures BGP metrics. Given the above mentioned difficulties to control incoming traffic distribution, the overlay routing control modifies the `local_pref` parameter in the remote AS (i.e. the outgoing traffic towards the AS that detected an SLS violation).

The Routing Control Platform presented in [FBR+04] relies on BGP and ah-hoc routing boxes out of the router to boost the global routing function, assuming that nowadays giga/terabit routers must be specialized in packet forwarding, delegating complicated tasks such as path computation and construction of the routing table to external entities.

Other initiatives are based in extensions of MPLS Traffic Engineering principles for inter-domain TE. Requirements for MPLS Inter-AS TE are being defined by the IETF [ZV05]. A recent proposal of RSVP-TE extensions to support inter-AS LSP establishment [PB03] allows efficient establishment of inter-domain LSPs. Optional fast restoration requirements, bypass tunnels, detour LSPs and other advanced features are supported, while complying with the aforementioned requirements, in particular confidentiality and protection. This proposal is a partial result of the IST Atrium project [Pel03], which reviews the state of the art in inter-domain routing and proposes several enhancements and extensions to existing protocols. There are also interesting proposals based on the Management Plane, evolving from the concept of Bandwidth Broker. For example the IST Mescal project [How04] proposes an architecture based in the advertisement of QoS capabilities between ASs, and builds a cascaded QoS peering model based in these advertised capabilities for soft QoS enforcement. QBGP is a fundamental building block of this proposal, and MPLS TE-LSPs are envisioned as a mechanism for inter-domain QoS with hard guarantees.

In the next section we will present the RMA inter-domain extension, which is inspired by the ideas presented in this section, exploring the possibilities of Control and Management Plane cooperation.

6.4 Inter-domain RMA extensions

Inter-domain connectivity agreement is based in the aforementioned pair of relationships: *customer-provider* and *peer-to-peer*. As described in Section 6.2, as far as end users updates concerned, is useful to establish QoS assurance policies on stub AS. According to [PB03], it is possible to establish TE-LSPs (a TE tunnel) between remote cooperating stub AS that agree exchanging traffic with certain QoS guarantees. Note that the TE tunnel enables the remote AS to establish a peer-to-peer relationship. This idea is illustrated in Figure 6.1.

This TE tunnel can be considered as a physical link for routing computation by the IGP, i.e. a routing adjacency is built, and AS1 and AS2 become peers (meaning directly connected) at the BGP routing level. Note also that other kind of services can be jointly deployed between remote peers, for instance inter-domain Layer 3 MPLS/BGP VPNs. This is possible thanks to the relationship both at the MPLS and BGP level. The TE Broker element in the figure is the termination point for the inter-domain TE tunnel. It can

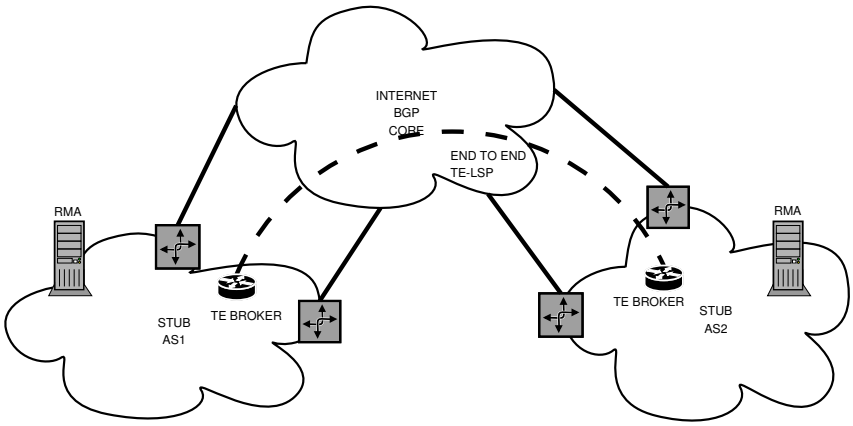

Figure 6.1: End to End TE Tunnel between remote stub ASs

be a stand-alone router or a logical entity, i.e. a loopback interface in a border BGP router. The TE broker implements the policy/filtering policies at the Data Plane, while the RMA is responsible for the intra-domain routing function and for the remote peering at the Management Plane (note that is possible to establish trusted relationships between RMAs in remote domains to cooperate in connectivity setup).

6.4.1 Inter-domain path setup using distributed RMAs

Cooperation between remote RMAs can be established over a management overlay, built over the end-to-end TE tunnel (by means of MPLS label stacking, for instance). These logical relationships are shown in Figure 6.2.

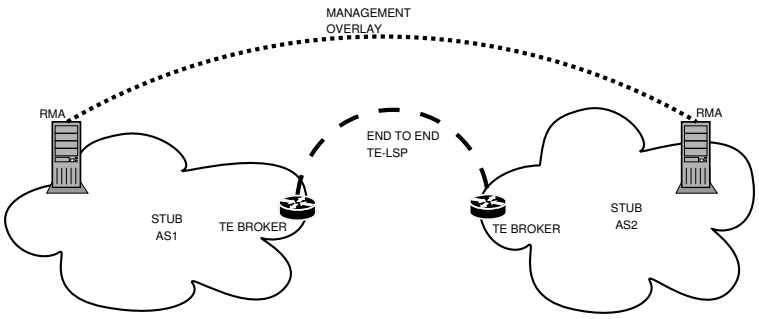

Figure 6.2: Logical relationships between remote AS

There are at least two possible ways to use such logical interconnection. The first option is to rely on BGP (and QBGP) attributes, establishing

the domain policies as filters in the border routers (the TE brokers). Note that when a TE tunnel is established between remote stub ASs and this link is advertised in the peering BGP session, at least two BGP paths exist between the considered ASs: (1) the usual IP AS-Path, and (2) the path over the forwarding adjacency (i.e. the TE tunnel). Thus, the inter-domain traffic may be classified as best-effort traffic, delivered over the standard BGP AS-Path, and guaranteed traffic, forwarded over the TE tunnel(s). This classification can be biased by network administrators at the RMA which in turn can configure suitable BGP policies in the TE Broker.

The usage of the RMAs in this option is restricted to bandwidth brokering. QoS guarantees are soft and enforced by BGP policies as mentioned above.

The second possibility is to let the RMAs cooperate to setup inter-domain LSPs transported by the TE tunnel. The capacity is then restricted to the TE tunnel bandwidth. While the first model is supported in IP and uses the TE tunnel only as a peering link, the second one unleashes the possibilities of the RMA for intra and inter-domain path computation, allowing network administrators to tightly manage inter-domain capacity. It is worth noting that in any case the QoS guarantees can only be enforced in the Ingress and Egress AS, while the inter-domain TE tunnel QoS enforcement depend on transit agreements. While it is possible to setup end-to-end inter-domain LSPs, the scalability of multiple independent TE-LSPs between network remote peers (i.e. cooperating stub ASs) is questionable. A more realistic approach is to establish TE end-to-end paths over previously configured end-to-end tunnels, supported by the TE Brokers and the RMAs. The setup of such end-to-end TE paths is depicted in Figure 6.3 and described hereafter:

1. A client or management application in AS1 configures the Ingress LSR, specifying the destination in AS2 and the QoS constraints.

2. The Ingress LSR requests computation from the RMA(AS1) with the given parameters.

3. RMA(AS1) realizes that the egress point belongs to a peering remote AS (AS2), checks credentials, and computes an intra-domain Explicit Route based on the QoS descriptors form the Ingress LSR towards the TE Broker in AS1. Using the management overlay, RMA(AS1) communicates with it peer RMA(AS2) in step (3'). The bandwidth to allocate the request on the TE tunnel is negotiated, and if there is agreement, resources are reserved in the TE tunnel and a corresponding intra-domain LSP in AS2 is signalled from TE Broker (AS2) to the egress point. The ER for this LSP is computed by the RMA(AS2) in step (3").

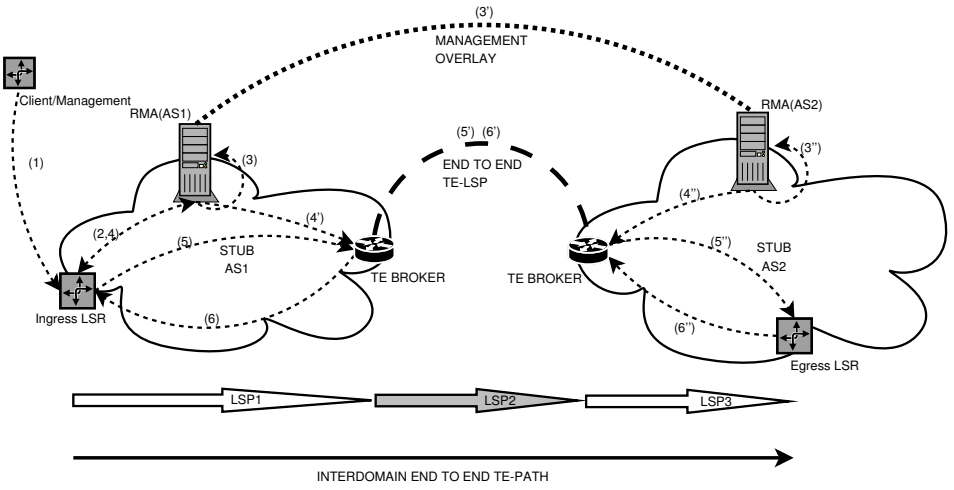

Figure 6.3: Inter-domain end-to-end path setup with cooperative RMAs

4. Once computed, the RMA(s) signals the corresponding LSPs in remote ASs, steps (4), (4'), (4"). Note that the assignment of resources in the TE Tunnel is done by the RMAs, configuring traffic policing in the tunnel endpoints. This simplifies the maintenance of the TE Tunnel signalling, since the global reservation is made only once (when configured), and the traffic is controlled by endpoints of the link in normal operation. Another option is to do label stacking, in which case an LSP is configured for each request.

5. Ingress LSR in AS1 issues a PATH message downstream to the TE Broker in AS1, while the TE Broker in AS2 issues a PATH message downstream to the Egress LSR in AS2 in step (5´´). Step (5') would be the PATH message from TE Broker (AS1) towards it correspondent TE-Tunnel endpoint, but this depends on the method chosen to control traffic, as mentioned in (4).

6. Upon reception of PATH message and label assignment, the TE Broker (AS1) issues a RESV message upstream to the Ingress LSR and the Egress LSR do the same towards the TE Broker (AS2) in step (6"). Again, Step (6') would be the RESV message from TE Broker (AS2) towards it correspondent TE-Tunnel endpoint, but this depends on the method chosen to control traffic, as mentioned in (4).

Once the Resv Message reaches the Ingress LSR, each LSP is established and traffic can be assigned to the appropriate Forwarding Equivalent Class (FEC). Note that the Inter-domain End-to-end Path is composed by three

corresponding LSPs (LSP1, LSP2, LSP3 in Figure 6.3), which are concatenated at the TE brokers using the MPLS Control Plane.

A relevant aspect of the proposal is that it ensures the confidentiality of the topology information among remote ASs. This is one of the fundamental requirements of the [ZV05] IETF document.

6.4.2 Evaluation of the proposal

A functional evaluation have been conducted using the topology depicted in Figure 6.4, where the link n7-n11 represents the inter-domain TE tunnel.

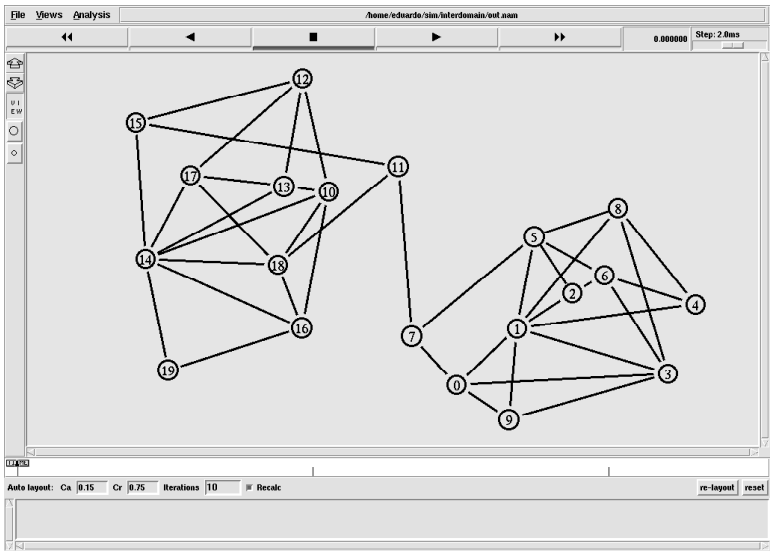

Figure 6.4: Inter-domain Topology with forwarding adjacency

Testing the RMA distributed solution over this topology shows no new results regarding the similar inter-area case. Moreover, timing information is not relevant in this case, since for inter-domain routing control is more important the quality of the connectivity, measured in AS-Path length, bandwidth assurance, and other QoS relevant parameters, than the establishment time, considering that the TE tunnel that supports the proposal would be established after a number of negotiations between transit ASs, where number of days may probably be an adequate metric for the elapsed time.

A more comprehensive future work evaluation intra-domain should take into account BGP peculiarities, with further specification of the RMA brokering function, and proper classification of traffic in the TE broker routers.

6.5 Conclusions

Inter-domain routing is a complex issue, and can not be easily tested. The establishment of TE-LSPs across Autonomous Systems in the Internet, using some of the proposals detailed in Section 6.3, give the foundation for RMA collaboration. The relationship between RMAs is similar to the inter-area cases, but routing opaqueness across Autonomous Systems, due to technical and business reasons, must be taken into account. Moreover, a bandwidth brokering mechanism is needed in the RMA relationship.

Chapter 7

General Conclusions

This concluding chapter provides a summary of the work presented in this thesis, revisits some partial conclusions given before, and identifies what has been achieved. A future work section is included, depicting promising aspects of the RMA architecture.

7.1 Review of contributions

This thesis presents a provisioning architecture that combines the strategies of the Control and Management Plane for connectivity provisioning into a hybrid process. The core of the architecture is an entity named Routing and Management Agent (RMA) which peers with both network devices and management systems. The performance of the proposed solution has been studied by simulation and compared with the pure Management and Control Plane solutions in the same environment. Results show a very reasonable degradation, in terms of provisioning time, in comparison with Control Plane LSP provisioning, because the solution enables a global optimization of network resources by means of an improved routing function outside LSRs. Anyway, response time is within the 50 millisecond bound. Bidirectional connectivity with path diversity solution has been achieved, using a push mechanism provided by the policy-based nature of the COPS protocol. Moreover, this mechanism allows to speed up the provisioning cycle, triggering connections directly from the RMA, upon reception of a request either from a client or a management application. An interesting side result, important in operational environments, is that the location of the RMA in the network is not relevant regarding the timing performance. In practical terms this means that there is no need to deploy a special DCN to reach the RMA functionality. Network administrators can experience an enhancement of their management capabilities, based on the network traffic engineering techniques enabled by the RMA architecture.

The solution presented "takes the routing out of the routers", enabling to

boost data forwarding and switching functionality of contemporary terabit routers, offloading CPU intensive tasks to external intelligence, namely the RMA agents. Consequently, congestion probability is alleviated, and traffic classification on provider edge routers is enhanced.

The RMA architecture is entirely based on standards protocols. A valid implementation of the PCE Architecture has been achieved, and an Internet Draft has been contributed to the IETF.

The solution has been extensively tested in the intra-area scenario, whereas a valid functional testing has been achieved for inter-area scenarios. Several possible implementations of the underlying protocols over the simulated and real testbed environment have been tested, achieving a successful integration of different patch modules for final quantitative evaluation. Further validation is being conducted in a real testbed composed of Linux-based and commercial routers, with an ongoing implementation of MPLS control plane and a management solution based on the RMA architecture.

The RMA architecture is the result of an integrated vision of diverse disciplines that permitted to approach the problem from different angles:

1. Optimization techniques for analysis of the CBR problem, network dimensioning and related problems.
2. Management, Control Plane, monitoring aspects of networking.
3. RMA fault-tolerant, redundant two tier architecture based on well-known clustering design of network server farms, and HPC techniques related with CBR problem parallelisation.

7.2 Future work

The thesis has achieved a comprehensive analysis of MPLS provisioning and related matters, biased towards the IP layer. A natural step forward is to study the provisioning of light-paths in the Optical Transport Network with GMPLS Control Plane, and its interactions with the client layers into a multi-technology, multilayer environment.

The RMA Constraint-Based Routing engine has been left aside in this thesis work, due to the complexity of that component of the architecture. A fundamental future work line is to explore the existing algorithms to fulfill the described functionality, and seek the achievement of interesting features such as effective load balancing and network recovery mechanisms.

The RMA architecture enables the application of provisioning policies using the management interface. A tool to assist network operator to design useful routing policies is also an interesting line of future work.

Finally, regarding the inter-domain provisioning, the study has shown that this case deserves further exploration, arriving to similar conclusions as the IETF Path Computation Element WG in this respect.

Appendix A

Technical Review

This appendix describes different software packages used in simulations in Section A.1, and an ongoing testbed implementation in Section A.2.

A.1 Simulation, topologies

The ns-2 simulator [ns205] is a complex discrete time object-oriented simulator, built using C++ and Otcl programming languages. It is an Open Source, ever growing project, with diverse contributions. Integration of features under a given operational environment is quite challenging. Many software patches and ns versions has been tried and integrated throughout the thesis work. The initial work has been developed using the KOM RSVP [kom05] package, which has the a-priori interesting advantage of being an integrated source code capable of generating executables for the simulator, an integrated emulator, and system environment. The promise of reusing the developed code to implement Linux routers with MPLS data plane and RSVP-TE signalling lead to try the package extensively, with very poor results. Other packages have been tried with similar results, finally arriving to a working environment integrated by the following pieces:

- ns-simulator version 2.26,
- RSVP-TE patch for n2 2.26 [CFV05],
- COPS patch for ns 2.27 [SL05].

An adaptation of the latter package had to be done to make the lot work together. The platform was also a major burden. Fedora Core version 2 and 3 machines refused to compile the combined packages, and finally a Debian machine with many tweaks could be used for the testing.

Building valid topologies is very important to enable validation of simulation results over practical operational networks. This thesis work was

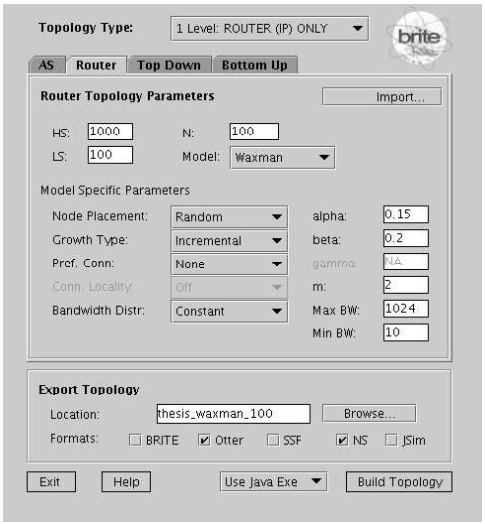

Figure A.1: Waxman 100 node topology generation parameters

based on the BRITE topology generator [MLMB05]. Waxman graphs where used extensively, and some validation testing was done using the Barabasi-Albert model. The generator builds topologies for the ns-2 simulator as a tcl file that can be integrated into the tcl simulation code. The parameters and output for the Waxman 100 nodes topologies used in the simulations are shown in Figure A.1. The generated topology semi-geographical layout is depicted in Figure A.2

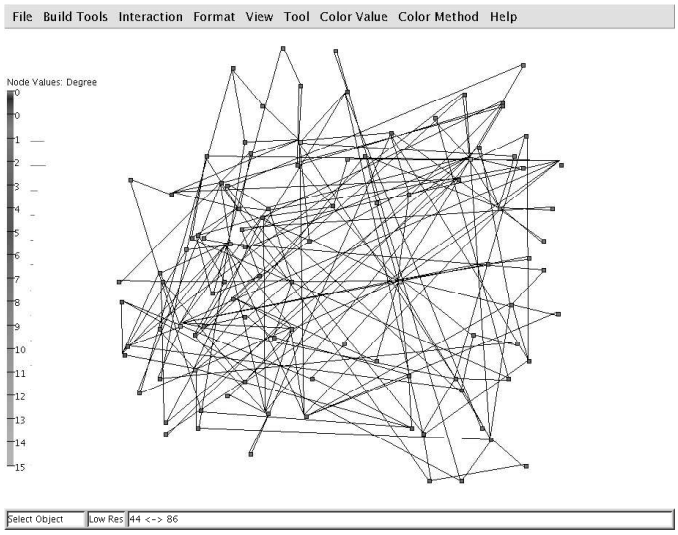

Figure A.2: Waxman 100 node semi-geographical layout

Results for Barabasi-Albert 100 node topology

The testing presented in Chapter 5 have been reproduced using a 100 node Barabasi-Albert (BA) topology. The generation parameters and semi-geographical layout are shown in Figures A.3 and A.4 respectively.

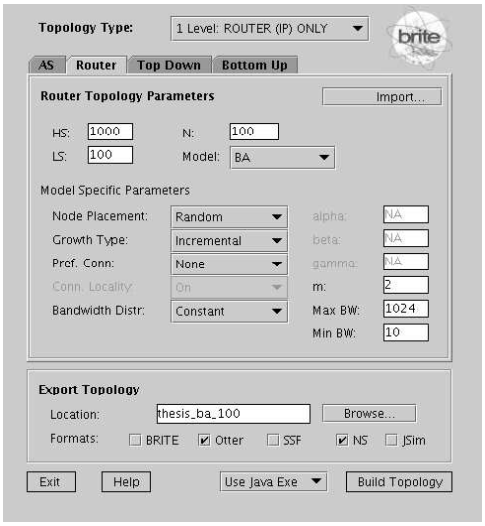

Figure A.3: Barabasi-Albert 100 node topology generation parameters

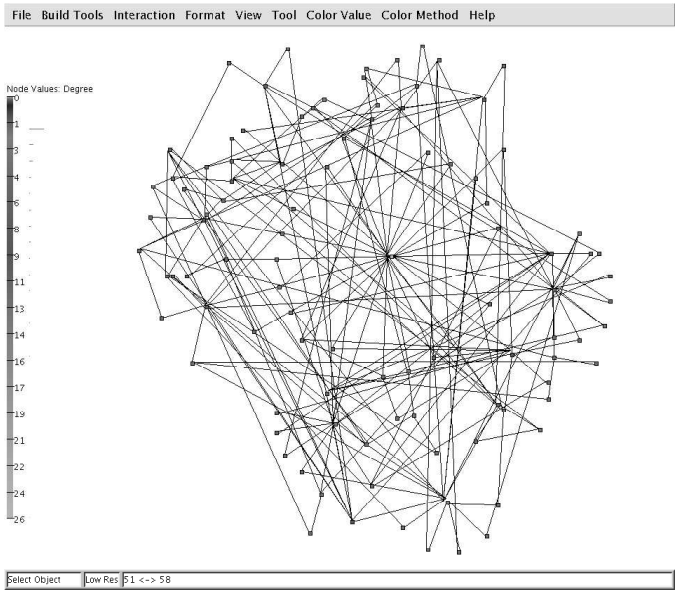

Figure A.4: Barabasi-Albert 100 node semi-geographical layout

An average of several realisations of the Reliable Connectivity Setup RMA case using 100 node BA topology is presented in Figure A.5. Note that the shape of the results is similar to the Waxman case, but path establishment takes longer times both in the pure Control Plane and the RMA case.

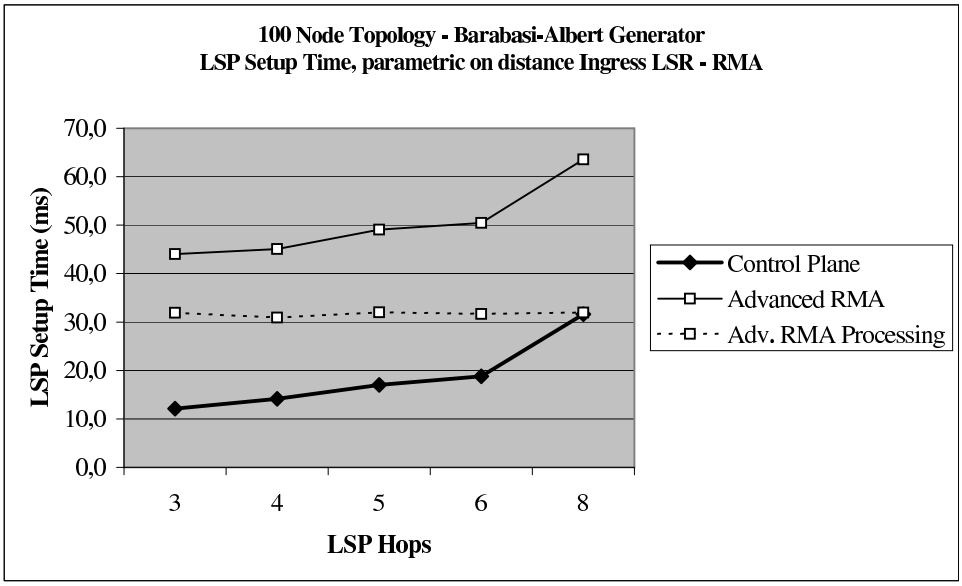

Figure A.5: 100 node topology results using Barabasi-Albert generator

A.2 Metropolitan Multiservice Network - RMS Project

RMS stands for "Red Metropolitana multiServicio" (Metropolitan Multiservice Network); this is an ongoing project undertaken by Universidad de la Republica (UdelaR) and Administracion Nacional de Telecomunicaciones (ANTEL), the public University and Telecommunication company of Uruguay, respectively.

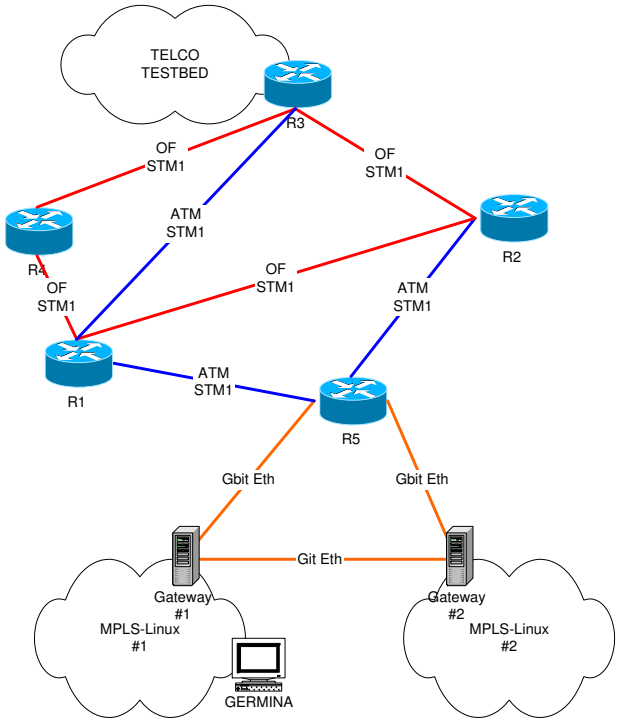

Figure A.6: RMS network layout

The objective of this project, started in 2003, is to build a metropolitan MPLS network as a testbed for Next Generation metropolitan multiservices. The core of the trial network is built using standard commercial MPLS routers, while the aggregation and access is composed of Linux-based routers, as shown in Figure A.6. The MPLS network runs the OSPF-TE routing protocol, and uses RSVP-TE signalling for the establishment of TE-LSPs.

The RMS management system is being developed using the RMA Architecture as a basis, integrating existing Provisioning and Inventory components built over JacORB. An external Traffic Engineering planning module is also being integrated to the solution. The management system is coined "GERMINA" (GEstor de Red del Grupo MINA [min05]); its general architecture is depicted in Figure A.7.

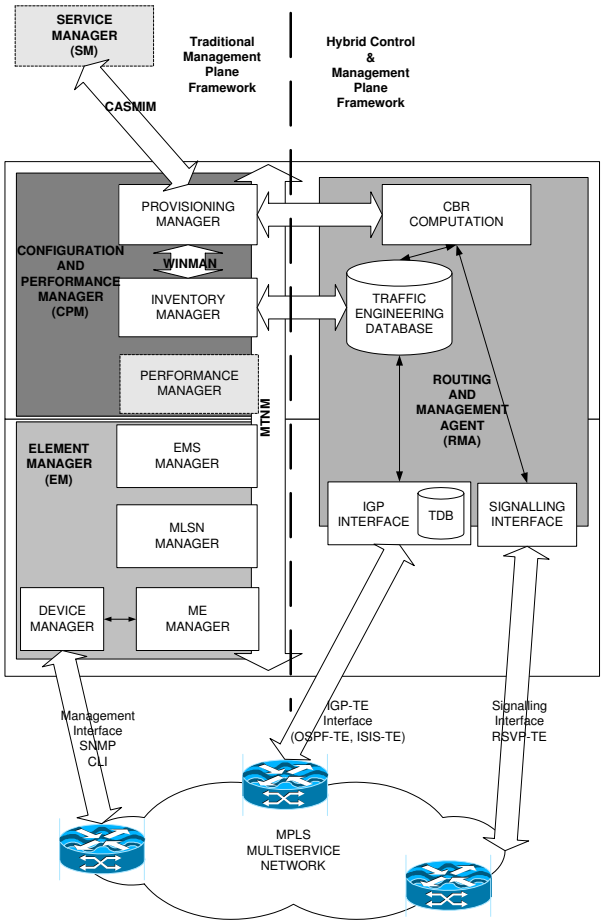

Figure A.7: GERMINA Management System

The GERMINA management system is located in one of the access clouds and gathers network information using a Management Virtual Private Network built over the MPLS infrastructure. A set of tools for traffic generation and monitoring are used to emulate the users' behaviour, modelled using real network traces obtained by Netflow on strategic nodes of the public IP network.

As seen on Figure A.6, the infrastructure comprises point to point STM-1 optical links, ATM switched STM-1 and Gigabit Ethernet links. The core is composed of Cisco 7206 VXR routers and the Gateways are Intel servers equipped with PRO/1000 MF Dual Port adapters, running MPLS-Linux [Leu05] and Quagga routing software [qua05].

The management system is in the process of integration. Net-SNMP tools are installed in the Linux nodes, which implement a subset of the MPLS

MIBs. Regarding the signalling, existing Linux RSVP-TE have quite poor functionality, though a home-grown implementation is being undertaken, to complement the data plane capabilities of MPLS-Linux. Partial testing of the building blocks have been achieved, whereas a complete proof of concept is planned for the end of 2005.

Appendix B

Acronyms

ABR Area Border Router
AS Autonomous System
ASON Automatically Switched Optical Network
ASTN Automatically Switched Transport Network
BGP Border Gateway Protocol
BTT Bidirectional Traffic Trunk
CBR Constraint Based Routing
COPS Common Open Policy Service
DCN Data Communication Network
DHCP Dynamic Host Configuration Protocol
DSLAM Digital Subscriber Line Access Multiplexer
FCAPS Fault, Configuration, Accounting, Performance, Security management functional areas
FEC Forward Equivalence Class
FSM Finite State Machine
GMPLS Generalized MPLS
IETF Internet Engineering Task Force
IGP Interior Gateway Protocol
IP Internet Protocol
IPO IP over Optical
IS-IS Intermediate System - Intermediate System routing protocol
ISP Internet SP
ITU-T International Telecommunication Union Telecom Standardization
LDP Label Distribution Protocol
LSP Label Switched Path
LSR Label Switched Router
MIB Management Information Base
MPLS MultiProtocol Label Switching
NE Network Element
NGN Next Generation Networks

NNI Node-to-node interface
OAM Operation and Management
OIF Optical Internetworking Forum
OSPF Open Shortest Path First routing protocol
OTN Optical Transport Network
OXC Optical CrossConnect
PCE Path Computation Element
PCC Path Computation Client
PVC Private Virtual Circuit
QoS Quality of Service
RMA Routing and Management Agent
RMON Remote Network Monitoring
RSVP Reservation Protocol
SDH Synchronous Digital Hierarchy
SLA Service Level Agreement
SNMP Simple Network Management Protocol
SP Service Provider
SRLG Shared Risk Link Group
TDB Topology DataBase
TE Traffic Engineering
TT Traffic Trunk
UNI User to Network Interface
WDM Wavelenght Division Multiplexing

Bibliography

[ABG+01] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, and G. Swallow, *RSVP-TE: Extensions to RSVP for LSP Tunnels*, RFC 3209 (Proposed Standard), December 2001, Updated by RFC 3936.

[AGKT99] G. Apostolopoulos, R. Guerin, S. Kamat, and S. K. Tripathi, *Improving QoS Routing Performance Under Inaccurate Link State Information*, 16th International Teletraffic Congress, June 1999.

[AKK+00] P. Aukia, M. Kodialam, P. V. N. Koppol, T. V. Lakshman, H. Sarin, and B. Suter, *RATES: a server for MPLS traffic engineering*, IEEE Network **14** (2000), no. 2, 34–41.

[ALR06] J. Ash and J.L. Le Roux, *PCE Communication Protocol Generic Requirements*, Internet Draft <draft-ietf-pce-comm-protocol-gen-reqs-01.txt>. Work in Progress. Expiration Date, January 2006.

[AMA+99] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus, *Requirements for Traffic Engineering Over MPLS*, RFC 2702 (Informational), September 1999.

[AN05] D. Allan and T.D. Nadeau, *A Framework for MPLS Operations and Management (OAM)*, Internet Draft <draft-ietf-mpls-oam-frmwk-03.txt>. Work in Progress. Expiration Date, August 2005.

[AS03] L. Andersson and G. Swallow, *The Multiprotocol Label Switching (MPLS) Working Group decision on MPLS signaling protocols*, RFC 3468 (Informational), February 2003.

[ASB03] P. Ashwood-Smith and L. Berger, *Generalized Multi-Protocol Label Switching (GMPLS) Signaling Constraint-based Routed Label Distribution Protocol (CR-LDP) Extensions*, RFC 3472 (Proposed Standard), January 2003, Updated by RFC 3468.

[ATM02] ATM Forum, *Private Network-Network Interface Specification v.1.1*, af-pnni-0055.001, April 2002.

[Awd99] D. O. Awduche, *MPLS and Traffic Engineering in IP Networks*, Communications Magazine, IEEE **37** (1999), no. 12, 42–47.

[Bek04] S. Beker, *Techniques d'Optimisation pour le Dimensionnement et la Reconfiguration des Reseaux MPLS*, Ph.D. thesis, Ecole Nationale Superieure des Telecommunications - France, April 2004.

[Bel58] Bellman R., *On a Routing Problem*, Quarterly of Applied Mathematics (1958).

[Ber03a] L. Berger, *Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description*, RFC 3471 (Proposed Standard), January 2003.

[Ber03b] L. Berger, *Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions*, RFC 3473 (Proposed Standard), January 2003, Updated by RFC 4003.

[Bro05] G. Brose, *JacORB*, CS department of Freie Universität Berlin. http://www.jacorb.org/ Last visited, July 2005.

[CAI05] CAIDA, *Cooperative Association for Internet Data Analysis*, Website http://www.caida.org/ Last visited, July 2005.

[CFV05] C. Callegari and F. Fabio Vitucci, *RSVP-TE/ns network simulator*, Website http://netgroup-serv.iet.unipi.it/rsvp-te_ns/ Last visited, July 2005.

[CHINB02] A. Coates, A.O. Hero III, R. Nowak, and Bin Yu, *Internet tomography*, IEEE Signal Processing Magazine **19** (2002), 47–65.

[CMM+03] C. Casetti, G. Mardente, M. Mellia, M. Manufo, and R. Lo Cigno, *On-line routing optimization for MPLS-based IP networks*, 2003, pp. 215–220.

[CS02] R. Chandra and J. Scudder, *Capabilities Advertisement with BGP-4*, RFC 3392 (Draft Standard), November 2002.

[CSD+01] K. Chan, J. Seligson, D. Durham, S. Gai, K. McCloghrie, S. Herzog, F. Reichmeyer, R. Yavatkar, and A. Smith, *COPS Usage for Policy Provisioning (COPS-PR)*, RFC 3084 (Proposed Standard), March 2001.

[CSL04] J. Cucchiara, H. Sjostrand, and J. Luciani, *Definitions of Managed Objects for the Multiprotocol Label Switching (MPLS), Label Distribution Protocol (LDP)*, RFC 3815 (Proposed Standard), June 2004.

[CYC+02] T. S. Choi, S. H. Yoon, H. S. Chung, C. H. Kim, J. S. Park, B. J. Lee, and T. S. Jeong, *Wise<TE>: traffic engineering server for a large-scale MPLS-based IP network*, Network Operations and Management Symposium, 2002. NOMS 2002, April 2002, pp. 251–264.

[DBC+00] D. Durham, J. Boyle, R. Cohen, S. Herzog, R. Rajan, and A. Sastry, *The COPS (Common Open Policy Service) Protocol*, RFC 2748 (Proposed Standard), January 2000.

[Dij59] E. W. Dijkstra, *A note on two problems in connexion with graphs*, Numerische Mathematik (1959).

[EJLW01] A. Elwalid, C. Jin, S. Low, and I. Widjaja, *MATE: MPLS Adaptive Traffic Engineering*, vol. 3, 2001, pp. 1300–1309.

[Fau05] F. Le Faucheur, *Protocol Extensions for Support of Diffserv-aware MPLS Traffic Engineering*, RFC 4124 (Proposed Standard), June 2005.

[FBR+04] Nick Feamster, Hari Balakrishnan, Jennifer Rexford, Aman Shaikh, and Kobus van der Merwe, *The Case for Separating Routing from Routers*, ACM SIGCOMM Workshop on Future Directions in Network Architecture (FDNA) (Portland, OR), September 2004.

[FF62] L. R. jr. Ford and D. R. Fulkerson, *Flows in Networks*, Princeton University Press, 1962.

[FL05] F. Le Faucheur and W. Lai, *Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*, RFC 4125 (Experimental), June 2005.

[FSI+05] A. Farrel, A. Satyanarayana, A. Iwata, N. Fujita, and G.R. Ash, *Crankback Signaling Extensions for MPLS and GMPLS RSVP-TE*, Internet Draft <draft-ietf-ccamp-crankback-05.txt>. Work in Progress. Expiration Date, November 2005.

[FTMP02] F.A. Kuipers, T. Korkmaz, M. Krunz, and P. Van Mieghem, *A Review of Constraint-Based Routing Algorithms*, Tech. report, June 2002.

[FVA06] A. Farrel, Vasseur. J.P., and J. Ash, *Path Computation Element (PCE) Architecture*, Internet Draft <draft-ietf-pce-architecture-01.txt>. Work In Progress. Expiration Date, January 2006.

[FWD+02] F. Le Faucheur, L. Wu, B. Davie, S. Davari, P. Vaananen, R. Krishnan, P. Cheval, and J. Heinanen, *Multi-Protocol Label Switching (MPLS) Support of Differentiated Services*, RFC 3270 (Proposed Standard), May 2002.

[Gao01] Lixin Gao, *On inferring Autonomous System relationships in the Internet*, IEEE/ACM Trans. Netw. **9** (2001), no. 6, 733–745.

[GBS03] E. Grampin, J. Baliosian, and J. Serrat, *Extensible, Transactional Architecture for IP Connectivity Management*, 3rd IEEE Latin American Network Operations and Management Symposium (LANOMS'2003), September 2003.

[GO99] R.A. Guerin and A. Orda, *QoS routing in networks with inaccurate information: theory and algorithms*, IEEE/ACM Transactions on Networking **7** (1999), no. 3, 350–364.

[Gra06] E. Grampin, *PCE Management Interface*, Internet Draft <draft-grampin-pce-mgmnt-if-00.txt>. Work in Progress. Expiration Date, January 2006.

[Hin04] R. Hinden, *Virtual Router Redundancy Protocol (VRRP)*, RFC 3768 (Draft Standard), April 2004.

[How04] M. Howarth, *Initial Specification of Protocols and Algorithms for Interdomain SLS Management and Traffic Engineering for QoS-based IP Service Delivery and their Test Requirements*, Tech. report, Project IST-2001-37961-Mescal D1.2 Deliverable., January 2004.

[HPW02] D. Harrington, R. Presuhn, and B. Wijnen, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*, RFC 3411 (Standard), December 2002.

[IET] IETF IPPM WG, *Internet Protocol Performance Metrics*, http://www.ietf.org/html.charters/ippm-charter.html.

[ITU00] ITU-T Recommendation G.805, *Generic functional architecture of transport networks.*

[ITU01a] ITU-T Recommendation G.807, *Requirements for Automatic Switched Transport Networks (ASTN)*, July 2001.

[ITU01b] ITU-T Recommendation G.8080, *Architecture for the Automatically Switched Optical Network (ASON)*, July 2001.

[ITU01c] ITU-T Recommendation G.872, *Architecture of optical transport networks*, November 2001.

[ITU02a] ITU-T Recommendation G.7715, *Architecture and Requirements for Routing in the Automatic Switched Optical Networks*, June 2002.

[ITU02b] ITU-T Recommendation Y.1711, *OAM mechanism for MPLS networks*, November 2002.

[ITU03a] ITU-T Recommendation G.709, *Interfaces for the Optical Transport Network (OTN)*, March 2003.

[ITU03b] ITU-T Recommendation G.7713.2, *Distributed Call and Connection Management: Signalling mechanism using GMPLS RSVP-TE*, March 2003.

[ITU03c] ITU-T Recommendation G.7713.3, *Distributed Call and Connection Management: Signalling mechanism using GMPLS CR-LDP*, March 2003.

[ITU04] ITU-T Recommendation G.7715.1, *ASON routing architecture and requirements for link state protocols*, February 2004.

[ITU05] ITU-T Recommendation G.7718, *Framework for ASON management*, February 2005.

[JAC+02] B. Jamoussi, L. Andersson, R. Callon, R. Dantu, L. Wu, P. Doolan, T. Worster, N. Feldman, A. Fredette, M. Girish, E. Gray, J. Heinanen, T. Kilty, and A. Malis, *Constraint-Based LSP Setup using LDP*, RFC 3212 (Proposed Standard), January 2002, Updated by RFC 3468.

[KKL] K. Kar, M. Kodialam, and T.V. Lakshman, *Minimum interference routing of bandwidth guaranteed tunnels with MPLS traffic engineering applications*, IEEE Journal on Selected Areas in Communications **18**.

[KKY03] D. Katz, K. Kompella, and D. Yeung, *Traffic Engineering (TE) Extensions to OSPF Version 2*, RFC 3630 (Proposed Standard), September 2003.

[KMF04] T. Karagiannis, M. Molle, and M. Faloutsos, *Long-range dependence ten years of Internet traffic modeling*, IEEE Internet Computing **8** (2004), no. 5, 57–64.

[kom05] *KOM RSVP Engine*, Website: http://www.kom.tu-darmstadt.de/rsvp/, February 2005.

[KR02] K. Kompella and Y. Rekhter, *LSP Hierarchy with Generalized MPLS TE*, Internet Draft <draft-ietf-mpls-lsp-hierarchy-08.txt>. Work in Progress. Expiration Date, March 2002.

[KS05] Kompella K. and Swallow G., *Detecting MPLS Data Plane Failures*, Internet Draft <draft-ietf-mpls-lsp-ping-09.txt>. Work in Progress. Expiration Date, November 2005.

[Lan04] J. Lang, *Link Management Protocol (LMP)*, Internet Draft <draft-ietf-ccamp-lmp-10.txt>. Work in Progress. Expiration Date, April 2004.

[Leu05] J. R. Leu, *MPLS-Linux*, Website: http://mpls-linux.sourceforge.net/ Last visited, July 2005.

[LF05] F. Le Faucheur, *Russian Dolls Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*, RFC 4127 (Experimental), June 2005.

[LKSJK02] Xiao Li, Lui King-Shan, Wang Jun, and Nahrsted Klara, *QoS Extension to BGP*, Proceedings of the 10th IEEE International Conference on Network Protocols, IEEE Computer Society, 2002, pp. 100–109.

[LR06] J.L. Le Roux, *Requirements for Path Computation Element (PCE) Discovery*, Internet Draft <draft-ietf-pce-discovery-reqs-01.txt>. Work in Progress. Expiration Date, January 2006.

[Man04] E. Mannie, *Generalized Multi-Protocol Label Switching (GMPLS) Architecture*, RFC 3945 (Proposed Standard), October 2004.

[MB03] X. Masip-Bruin, *Mechanisms to Reduce the Routing Information Inaccuracy Effects: Application to MPLS and WDM Networks*, Ph.D. thesis, Universitat Politècnica de Catalunya, 2003.

[MCG+03] E. Mykoniati, C. Charalampous, P. Georgatsos, T. Damilatis, D. Goderis, P. Trimintzios, G. Pavlou, and D. Griffin, *Admission control for providing QoS in DiffServ IP networks: the TEQUILA approach*, IEEE Communications Magazine **41** (2003), 38–44.

[Met05] L. Metzger, *Response Time Monitor MIB*, Cisco Systems. ftp://ftp-sj.cisco.com/pub/mibs/v2/CISCO-RTTMON-MIB.my Last visited, July 2005.

[min05] *MINA Research Group*, Website http://www.fing.edu.uy/inco/grupos/mina/ Last visited, July 2005.

[MK00] K. McCloghrie and F. Kastenholz, *The Interfaces Group MIB*, RFC 2863 (Draft Standard), June 2000.

[MLMB05] A. Medina, A. Lakhina, I. Matta, and J. Byers, *BRITE: Boston university Representative Internet Topology gEnerator*, Website: http://www.cs.bu.edu/brite/ Last visited, July 2005.

[MPM+05] R. Munoz, C. Pinart, R. Martinez, J. Sorribes, and G. Junyent, *ADRENALINE Testbed: User Management of Lightpaths over Intelligent Optical WDM Networks through GMPLS and XML*, First International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005., February 2005, pp. 252–261.

[MRG03] Martinez C., Rodriguez F., and Grampin E., *A Real World Example of a Distributed E-Mail System*, Tercer Congreso Iberoamericano de Telematica - CITA'2003, October 2003.

[NC04] T. Nadeau and J. Cucchiara, *Definitions of Textual Conventions (TCs) for Multiprotocol Label Switching (MPLS) Management*, RFC 3811 (Proposed Standard), June 2004.

[ns205] *The Network Simulator - ns-2*, Website: http://www.isi.edu/nsnam/ns/ Last visited, July 2005.

[NSF04] Nadeau T. D., Srinivasan C., and Farrel A., *Multiprotocol Label Switching (MPLS) Management Overview*, Intenet Draft <draft-ietf-mpls-mgmt-overview-09.txt>. Work in Progress. Expiration Date, August 2004.

[NSV04] T. Nadeau, C. Srinivasan, and A. Viswanathan, *Multiprotocol Label Switching (MPLS) Forwarding Equivalence Class To Next Hop Label Forwarding Entry (FEC-To-NHLFE) Management Information Base (MIB)*, RFC 3814 (Proposed Standard), June 2004.

[Oht02] H. Ohta, *Assignment of the 'OAM Alert Label' for Multiprotocol Label Switching Architecture (MPLS) Operation and Maintenance (OAM) Functions*, RFC 3429 (Informational), November 2002.

[OR05] T. Oetiker and D. Rand, *Multi Router Traffic Grapher (MRTG)*, GNU General Public License Software: http://www.mrtg.org/ Last visited, July 2005.

[PB03] C. Pelsser and O. Bonaventure, *Extending RSVP-TE to support inter-AS LSPs*, Workshop on High Performance Switching and Routing (HPSR), no. 24-27, June 2003, pp. 79–84.

[Pel03] C. Pelsser, *Assessment of protocols and algorithms for inter-domain traffic engineering*, Tech. report, Project IST-1999-20675-Atrium D4.2 Deliverable, January 2003.

[PM04] M. P. Pióro and D. Medhi, *Routing, Flow, and Capacity Design in Communication and Computer Networks*, Morgan Kaufman, 2004.

[PSA05] P. Pan, G. Swallow, and A. Atlas, *Fast Reroute Extensions to RSVP-TE for LSP Tunnels*, RFC 4090 (Proposed Standard), May 2005.

[qua05] *Quagga Routing Software Suite*, Website: http://www.quagga.net/ Last visited, July 2005.

[RHK+03] L. Raptis, G. Hatzilias, F. Karayannis, K. Vaxevanakis, and E. Grampin, *An integrated network management approach for managing hybrid IP and WDM networks*, IEEE Network **17** (2003), 37–43.

[RL95] Y. Rekhter and T. Li, *A Border Gateway Protocol 4 (BGP-4)*, RFC 1771 (Draft Standard), March 1995.

[RLA04] B. Rajagopalan, J. Luciani, and D. Awduche, *IP over Optical Networks: A Framework*, RFC 3717 (Informational), March 2004.

[RR01] Y. Rekhter and E. Rosen, *Carrying Label Information in BGP-4*, RFC 3107 (Proposed Standard), May 2001.

[RR05] E. Rosen and Y. Rekhter, *BGP/MPLS IP VPNs*, Internet Draft <draft-ietf-l3vpn-rfc2547bis-03.txt>. Work in Progress. Expiration Date, April 2005.

[RVB05] J.-L. Le Roux, J.-P. Vasseur, and J. Boyle, *Requirements for Inter-Area MPLS Traffic Engineering*, RFC 4105 (Informational), June 2005.

[RVC01] E. Rosen, A. Viswanathan, and R. Callon, *Multiprotocol Label Switching Architecture*, RFC 3031 (Proposed Standard), January 2001.

[SAdO+04] C. Scoglio, T. Anjali, J. C. de Oliveira, I. F. Akyildiz, and G. UhI, *TEAM: A traffic engineering automated manager for DiffServ-based MPLS networks*, Communications Magazine, IEEE **42** (2004), no. 10, 134–145.

[SDIR05] G. Swallow, J. Drake, H. Ishimatsu, and Y. Rekhter, *Generalize Multiprotocol Label Switching (GMPLS) User-Network Interface (UNI): Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Support for the Overlay Model*, Internet Draft <draft-ietf-ccamp-gmpls-overlay-05.txt>. Work in Progress. Expiration Date, March 2005.

[SGD05] A. Sridharan, R. Guerin, and C. Diot, *Achieving near-optimal traffic engineering solutions for current OSPF/IS-IS networks*, IEEE/ACM Transactions on Networking **13** (2005).

[SH03] V. Sharma and F. Hellstrand, *Framework for Multi-Protocol Label Switching (MPLS)-based Recovery*, RFC 3469 (Informational), February 2003.

[SKL+03] J. Song, S. Kim, M. Lee, H. Lee, and T. Suda, *Adaptive load distribution over multipath in MPLS networks*, IEEE International Conference on Communications, 2003. ICC '03., vol. 1, May 2003, pp. 233– 237.

[SL04] H. Smit and T. Li, *Intermediate System to Intermediate System (IS-IS) Extensions for Traffic Engineering (TE)*, RFC 3784 (Informational), June 2004.

[SL05] A. Sayenko and Lahnalampi. T., *COPS (Common Open Policy Service) protocol for the NS-2 simulator*, Website: http://www.cc.jyu.fi/ sayenko/pages/en/projects.htm Last visited, July 2005.

[SNV04] C. Srinivasan, T. Nadeau, and A. Viswanathan, *Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Management Information Base (MIB)*, RFC 3812 (Proposed Standard), June 2004.

[SVN04] C. Srinivasan, A. Viswanathan, and T. Nadeau, *Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base (MIB)*, RFC 3813 (Proposed Standard), June 2004.

[The01] The Optical Internetworking Forum, *User Network Interface (UNI) 1.0 Signaling Specification - Implementation Agreement OIF-UNI-01.0*, October 2001.

[Var96] Y. Vardi, *Network tomography: Estimating source-destination traffic intensities from link data*, Journal of American Statistics Association **91** (1996), no. 433, 365–377.

[Vas05] Vasseur JP., *RSVP Path computation request and reply messages*, Internet Draft <draft-vasseur-mpls-computation-rsvp-05.txt>. Work in Progress. Expiration Date, January 2005.

[Vas06] J.P. Vasseur, *Path Computation Element (PCE) communication Protocol (PCEP) - Version 1*, Internet Draft <draft-vasseur-pce-pcep-01.txt>. Work in Progress. Expiration Date, January 2006.

[Wax88] B. M. Waxman, *Routing of multipoint connections*, IEEE Journal on Selected Areas in Communications **6** (1988), no. 9, 1617–1622.

[XHBN00] X. Xiao, A. Hannan, B. Bailey, and L. M. Ni, *Traffic engineering with MPLS in the internet*, IEEE Network **14** (2000), no. 2TY - JOUR, 28–33.

[YFMB+04] M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sánchez-López, M. Curado, and J. Domingo-Pascual, *A Proposal for Inter-Domain QoS Routing based on Distributed Overlay Entities and QBGP*, First International Workshop on QoS Routing (WQoSR'2004), October 2004.

[ZV05] R. Zhang and J.P. Vasseur, *MPLS Inter-AS Traffic Engineering requirements*, Internet draft <draft-ietf-tewg-interas-mpls-te-req-09.txt>. Work in Progress. Expiration Date, March 2005.