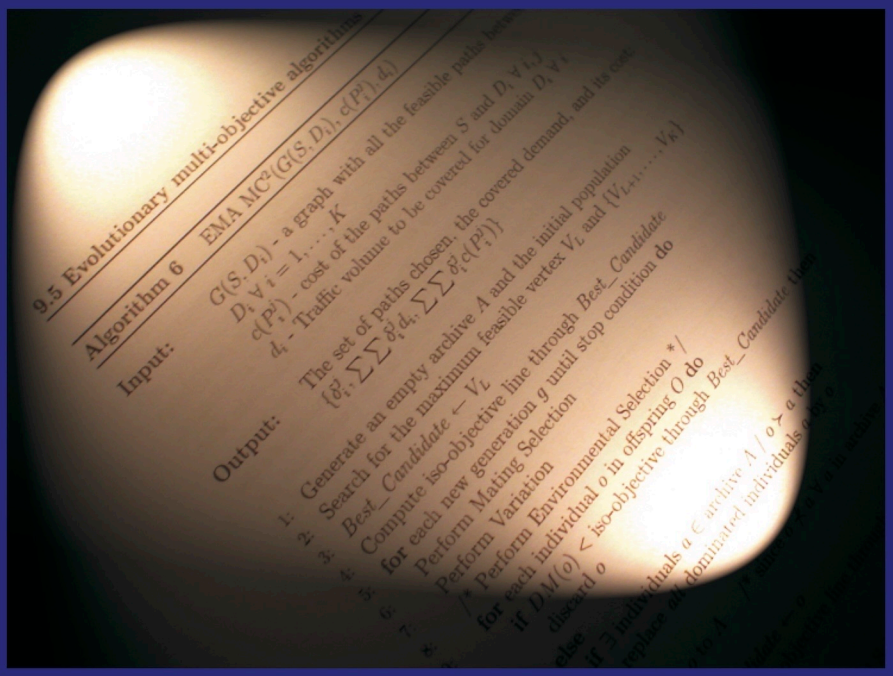
Strategies for Internet Route Control Past, Present, and Future

Author: Marcelo Yannuzzi

Advisor: Dr. Xavier Masip-Bruin
Co-Advisor: Dr. Edmundo Monteiro



Department of Computer Architecture

A THESIS PRESENTED TO
THE TECHNICAL UNIVERSITY OF CATALONIA (UPC) IN FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR
2007

UPC

Strategies for Internet Route Control

Past, Present, and Future

Author: Marcelo Yannuzzi

Advisor: Dr. Xavier Masip-Bruin

Co-advisor: Dr. Edmundo Monteiro

Department of Computer Architecture
Technical University of Catalonia (UPC)

A THESIS PRESENTED TO THE TECHNICAL UNIVERSITY OF CATALONIA IN FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR IN COMPUTER SCIENCE

December 19, 2007

Department of Computer Architecture
Technical University of Catalonia (UPC)
Av. Víctor Balaguer s/n
08800 Vilanova i la Geltrú
Barcelona, Spain

to Tathiana, Camila, and Chani

The grandson asks his grandparent:

– Grandpa, what was the Internet?
– Shhhh! Are you crazy?
– Why?
– Children ... you know ... we used to call it like that, but it doesn't like it.
– You mean the ... – the grandson hesitates – ... oh, I see.
– Yes – said the grandparent.
– But, why? Why doesn't it want to be called like that?
– Because it refers to the most primitive state of its evolution.
– I still don't get what the Internet was, but I understand that – the grandson hesitates again – ... *it* ... has evolved from there.
– Exactly. And stop repeating that name!
– So, if it is not ... *that* ... anymore, then, what is it now?
– A breathing monster. Oh, I shouldn't have said that!

Marcelo Yannuzzi, 2007

Acknowledgments

The development and writing of this thesis could not have been possible without the help of a large number of people and some organizations that – directly or indirectly – supported the different stages of my work.

In particular, I owe an immeasurable debt of gratitude to my supervisor, Dr. Xavier Masip-Bruin, for his impeccable guidance, his clever and always pragmatic advice, and his invaluable support in several facets of my thesis and personal life. It has been a great pleasure and an enriching experience to work under his supervision. Words can hardly express my gratitude to you Xavi, I wish you know that.

I would also like to express my deep gratitude to Prof. Jordi Domingo-Pascual. His support throughout my thesis has been crucial. Our discussions and the experiences I have gained, especially while we were abroad, have always been of great value to me. It has been a privilege to carry out my PhD in the research group and the Department that he leads.

The other person that has played a key role during the development of my thesis is Dr. Sergio Sánchez-López. His help has been evident all the way through; in our discussions, in our joint papers, in counting with my participation in every new and challenging initiative, in easing my duties, and more. Sergi, thank you very much indeed.

I also wish to acknowledge my gratitude to Prof. Josep Solé-Pareta for his permanent support to the extension of my work to the area of optical networks.

Many thanks to the rest of the researchers in our group, and particularly, to Dr. Eva Marin-Tordera, and MSc. René Serral-Gracià, for their help and insights in some of my most recent works. I wish to thank MSc. Guillem Fabregó, who obtained part of the results present in this thesis, and also the support received in countless occasions from MSc. José Nuñez.

To end up with my acknowledgments to members of the Department, I would like to thank Prof. Eduard Ayguadé-Parra, and Dr. Leandro Navarro-Moldes, for their help at the beginning of my PhD. Thanks for making my transition from a faraway country, and from industry to academia, so easy and encouraging.

An important component in the development of this thesis has been my interaction and collaboration with a number of external researchers. I am especially indebted to MSc. Alexandre Fonte, Dr. Edmundo Monteiro, and Dr. Marilia Curado, from the University of Coimbra. A significant part of

the evaluations present in this manuscript were obtained by Alexandre, and they are the result of years of joint work. Our discussions have helped improving and refining my proposals in a substantial way. Up to the writing of this manuscript, the largest simulation framework for competitive intelligent route control that I am aware of, is the one developed by Alexandre.

I would also like to thank Prof. Olivier Bonaventure from the Université catholique de Louvain for some of the most interesting and formative discussions that I have had in inter-domain routing and Traffic Engineering. I have also enjoyed the discussions with some of the researchers in his team, in particular: Dr. Pierre François, Dr. Bruno Quoitin, MSc. Sebastien Tandel, Dr. Cristel Pelsser, and Dr. Cedric de Launois.

Over the last year and a half, I have had the opportunity to met and work with two outstanding researchers, Prof. Ariel Orda from Technion, the Israeli Institute of Technology, and Dr. Alexander Sprintson from Texas A&M University. Their expertise and advice have significantly improved the quality of our joint work, part of which is integrated in this thesis. It has been a pleasant and enriching experience to work with both of them.

I would also like to express my gratitude to my PhD evaluation committee: Prof. Arturo Azcorra, Prof. Josep Solé-Pareta, Dr. Michel Diaz, Prof. Piet Van Mieghem, and Dr. Abdelhamid Mellouk, for reading this manuscript, as well as to the external reviewers for their help in improving different aspects of this thesis.

Some organizations have played a key role supporting the development of this thesis. I would like to especially thank the following European Networks of Excellence: E-NEXT (Emerging Networking Experiments and Technologies: FP6-506869) and CONTENT (Excellence in Content Distribution Network Research: FP6-0384239). I am also indebted to the European Doctoral School of Advanced Topics In Networking (SATIN), the European Integrated Project EuQoS (End-to-end Quality of Service support over heterogeneous networks: FP6-004503), the Spanish research projects contracts nº FEDER-TIC2002-04531-C04-02, TEC2005-08051-C03-01, and FEDER-TSI2005-07520-C03-02, the Department of Computer Architecture and the Technical University of Catalonia (UPC), the Department of Electrical Engineering from the University of the Republic in Uruguay, the National Teleco in Uruguay (A.N.TEL), and OPNET Technologies Inc.

My biggest debt of gratitude, however, is to my wife, Chani. I would have never reached this point without her unwavering support, her strength, and her unfailing love. Words cannot truly express my gratitude for her trust and support. Thank you very much Piri!

Marcelo Yannuzzi, December 2007

Abstract

One of the most complex problems in computer networks is how to provide guaranteed performance and reliability to the communications carried out between nodes located in different domains. This is due to several reasons – which will be analyzed in detail in this thesis – but in brief, this is mostly due to: i) the limited capabilities of the current inter-domain routing model in terms of Traffic Engineering (TE); ii) the distributed and potentially conflicting nature of policy-based routing, where routing policies are managed independently and without coordination among domains; and iii) the clear limitations of the inter-domain routing protocol, namely, the Border Gateway Protocol (BGP).

The goal of this thesis is precisely to study and propose solutions allowing to drastically improve the performance and reliability of inter-domain communications. One of the most important tools to achieve this goal, is to control the routing and TE decisions performed by routing domains. Therefore, this thesis explores different strategies on how to control such decisions in a highly efficient and accurate way.

At present, this control mostly resides in BGP, but as mentioned above, BGP is in fact one of the main causes of the existing limitations. The natural next-step would be to replace BGP, but the large installed base at present together with its recognized effectiveness in other aspects, are clear indicators that its replacement (or its possible evolution) will probably be gradually put into practice.

In this framework, this thesis proposes to to study and contribute with novel strategies to control the routing and TE decisions of domains in three different time frames: i) at present in IP multi-domain networks; ii) in the near-future in IP/MPLS (MultiProtocol Label Switching) multi-domain networks; and iii) in the future optical Internet, modeling in this way a realistic and progressive evolution, facilitating the gradual replacement of BGP.

More specifically, the contributions in this thesis can be summarized as follows.

- We start by proposing incremental strategies based on Intelligent Route Control (IRC) solutions for IP networks. The strategies proposed in this case are incremental in the sense that they interact with BGP, and tackle several of its well-known limitations.

- Then, we propose a set of concurrent route control strategies for MPLS networks, based on broadening the concept of the Path Computation Element (PCE) coming from the IETF (Internet Engineering Task Force). Our strategies are concurrent in the sense that they do not interact directly with BGP, and they can be deployed in parallel. In this case, BGP still controlls the routing and TE actions concerning regular IP-based traffic, but not how IP/MPLS paths are routed and controlled. These are handled independently by the PCEs.

- We end with the proposal of a set of route control strategies for multi-domain optical networks, where BGP has been completely replaced. These strategies are supported by the introduction of a new route control element, which we named Inter-Domain Routing Agent (IDRA). These IDRAs provide a dedicated control plane, i.e., physically independent from the data plane, and with high computational capacity for future optical networks.

The results obtained validate the effectiveness of the strategies proposed here, and confirm that our proposals significantly improve both the conception and performance of the current IRC solutions, the expected PCE in the near-future, as well as the existing proposals about the optical extension of BGP.

Resumen

Uno de los problemas más complejos en redes de computadores es el de proporcionar garantías de calidad y confiabilidad a las comunicaciones de datos entre entidades que se encuentran en dominios distintos. Esto se debe a un amplio conjunto de razones – las cuales serán analizadas en detalle en esta tesis – pero de manera muy breve podemos destacar: i) la limitada flexibilidad que presenta el modelo actual de encaminamiento inter-dominio en materia de ingeniería de tráfico; ii) la naturaleza distribuida y potencialmente antagónica de las políticas de encaminamiento, las cuales son administradas individualmente y sin coordinación por cada dominio en Internet; y iii) las carencias del protocolo de encaminamiento inter-dominio utilizado en Internet, denominado BGP (Border Gateway Protocol).

El objetivo de esta tesis, es precisamente el estudio y propuesta de soluciones que permitan mejorar drásticamente la calidad y confiabilidad de las comunicaciones de datos en redes conformadas por múltiples dominios.

Una de las principales herramientas para lograr este fin, es tomar el control de las decisiones de encaminamiento y las posibles acciones de ingeniería de tráfico llevadas a cabo en cada dominio. Por este motivo, esta tesis explora distintas estrategias de como controlar en forma precisa y eficiente, tanto el encaminamiento como las decisiones de ingeniería de tráfico en Internet.

En la actualidad este control reside principalmente en BGP, el cual como indicamos anteriormente, es uno de los principales responsables de las limitantes existentes. El paso natural sería reemplazar a BGP, pero su despliegue actual y su reconocida operatividad en muchos otros aspectos, resultan claros indicadores de que su sustitución (ó su posible evolución) será probablemente gradual.

En este escenario, esta tesis propone analizar y contribuir con nuevas estrategias en materia de control de encaminamiento e ingeniería de tráfico inter-dominio en tres marcos temporales distintos: i) en la actualidad en redes IP; ii) en un futuro cercano en redes IP/MPLS (MultiProtocol Label Switching); y iii) a largo plazo en redes ópticas, modelando así una evolución progresiva y realista, facilitando el reemplazo gradual de BGP.

Más concretamente, este trabajo analiza y contribuye mediante:

- La propuesta de estrategias incrementales basadas en el Control Inteligente de Rutas (Intelligent Route Control, IRC) para redes IP en la actualidad. Las estrategias propuestas en este caso son de carácter

incremental en el sentido de que interaccionan con BGP, solucionando varias de las carencias que éste presenta sin llegar a proponer aún su reemplazo.

- La propuesta de estrategias concurrentes basadas en extender el concepto del PCE (Path Computation Element) proveniente del IETF (Internet Engineering Task Force) para redes IP/MPLS en un futuro cercano. Las estrategias propuestas en este caso son de carácter concurrente en el sentido de que no interaccionan con BGP y pueden ser desplegadas en forma paralela. En este caso, BGP continúa controlando el encaminamiento y las acciones de ingeniería de tráfico inter-dominio del tráfico IP, pero el control del tráfico IP/MPLS se efectúa en forma independiente de BGP mediante los PCEs.

- La propuesta de estrategias que reemplazan completamente a BGP basadas en la incorporación de un nuevo agente de control, al cual denominamos IDRA (Inter-Domain Routing Agent). Estos agentes proporcionan un plano de control dedicado, físicamente independiente del plano de datos, y con gran capacidad computacional para las futuras redes ópticas multi-dominio.

Los resultados expuestos aquí validan la efectividad de las estrategias propuestas, las cuales mejoran significativamente tanto la concepción como la performance de las actuales soluciones en el área de Control Inteligente de Rutas, del esperado PCE en un futuro cercano, y de las propuestas existentes para extender BGP al área de redes ópticas.

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

AR	Aggregated Representation of the network
AS	Autonomous System
ASBR	Autonomous System Border Router
ASON	Automatically Switched Optical Network
BGP	Border Gateway Protocol
BR	Blocking Ratio
CCAMP	Common Control And Management Plane
CCDF	Complementary CDF
CDF	Cumulative Distribution Function
CRC	Cooperative Route Control
CRCs	Cooperative Route Controllers
CSP	Content Service Provider
CSRC	Cooperative & Social Route Control
CSRCs	Cooperative & Social Route Controllers
DDRP	Domain-to-Domain Routing Protocol
DM	Decision Maker
DNS	Domain Name Service
eBGP	External BGP
EMA	Evolutionary Multi-objective Algorithm
ENAW	Effective Number of Available Wavelengths
E-NNI	External Network-Network Interface
ERO	Explicit Route Object
FIB	Forwarding Information Base
GMPLS	Generalized Multiprotocol Label Switching
iBGP	Internal BGP
IDRA	Inter-Domain Routing Agent
IETF	Internet Engineering Task Force
IGP	Interior Gateway Protocol
I-NNI	Internal Network-Network Interface
IP	Internet Protocol

IRC	Intelligent Route Control
IRCs	Intelligent Route Controllers
IS-IS	Intermediate System-to-Intermediate System Protocol
ISP	Internet Service Provider
ITU	International Telecommunications Union
LOCAL_PREF	Local Preference (BGP attribute)
LSP	Label Switched Path
LSR	Label Switching Router
MC^2	Maximum Coverage at minimum Cost algorithm
MCP	Multi-Constrained Problem
MED	Multi-Exit Discriminator (BGP attribute)
MKP	Multi-dimensional Knapsack Problem
MPLS	Multiprotocol Label Switching
MRAI	Minimum Route Advertisement Interval (BGP timer)
NAT	Network Address Translation
NH	Next-Hop
NRI	Network Reachability Information
NSP	Network Service Provider
OBGP	Optical Border Gateway Protocol
OBGP+	An enhanced version of OBGP
OIF	Optical Internetworking Forum
OSPF	Open Shortest Path First
OXC	Optical Cross-Connect
PCC	Path Computation Client
PCE	Path Computation Element
PSI	Path State Information
QoR	Quality of Resilience
QoS	Quality of Service
QoSR	Quality of Service Routing
RCD	Routing Control Domain

RCN	Root Cause Notification
RFC	Request For Comments
RIP	Routing Information Protocol
RSVP	Resource Reservation Protocol
RWA	Routing and Wavelength Assignment
SPEA2	Strength Pareto Evolutionary Algorithm (version 2)
SRC	Social Route Control
SRCs	Social Route Controllers
TCP	Transmission Control Protocol
TE	Traffic Engineering
TED	Traffic Engineering Database
UNI	User Network Interface
VoD	Video on Demand
VoIP	Voice over IP
VPLS	Virtual Private LAN Service
VPN	Virtual Private Network
WG	IETF's Working Group

Part I

Introduction

Chapter 1

Summary and Road Map

This introductory chapter starts by discussing the motivations and the objectives of this thesis. The rest of the chapter points out the main contributions, and concludes with an overview of the structure of this manuscript.

1.1 Motivations

The Internet is in a permanent state of transition. Quite likely, part of it is being improved and upgraded right now, while this manuscript is being written. Its constant evolution becomes clear by tracing its history, and also, by investigating what experts claim about its future. Even though experts in the field cannot assure how the Internet would look like in the next ten, fifteen or twenty years, they do agree that it will be different than the one we know today [40, 41, 43]. The Internet started almost four decades ago. Throughout these years it has transitioned many states, evolving from the interconnection of a small number of machines – supported by a best-effort paradigm – to the largest and most complex distributed system ever developed by man.

Over the last few years, two of the most noteworthy features of its evolution have been the rapid growth of real-time and multimedia communications across the network, and the growing tendency towards the convergence of digital communications onto IP-based network technologies. A major problem, however, is that the successful support of a growing set of such kinds of communications over the Internet is strongly conditioned by the performance guarantees that can be offered to their traffic (e.g. in terms of delay, jitter, packet loss, available bandwidth, or a combination of them). For this reason, the industry and research community have actively worked in the design of different mechanisms aimed at guaranteeing the performance and reliability of the communications over the Internet. The advances at the intra-domain level are evident. Multiprotocol Label Switching (MPLS) [28], has become the core switching infrastructure inside domains, which is mostly attributable to the undeniable potential of MPLS in terms of Traffic Engineering (TE) [94], Quality of Service (QoS) delivery [31], path protection, fast recovery

from network failures [80, 81] and also due to the flexibility it offers in terms of Virtual Private Networks (VPNs) management [100].

On the contrary, the problem of guaranteeing the performance and reliability of the traffic when the communicating parties are located in different domains results particularly challenging, and remains largely unsolved [138]. The main difficulties reside on the very foundations of the current inter-domain network paradigm. This latter is based on a highly scalable and completely distributed network architecture [50], linked together by a single routing protocol, namely, the Border Gateway Protocol (BGP) [106]. The intrinsic limitations of this model are essentially the following[1]:

- Limited TE capabilities. In practice, there is still neither a model nor a globally useful mechanism for distributing TE information (and TE demands) among domains.
- Distributed and potentially conflicting policy-based routing between domains.
- BGP has no inbuilt QoS capabilities. Indeed, BGP only handles reachability information – for scalability reasons network state information is never exchanged between domains.
- BGP is a slow reacting routing protocol that might require some minutes to recover from a router or a link failure.
- For scalability reasons, each BGP router only advertises the best path it knows to reach a destination. This is also the only path used by BGP to forward traffic to the destination. This behavior drastically reduces the number of alternative paths that a node can use for improving the performance or the reliability of its traffic.

In sum, the current multi-domain network model was designed to manage the most fundamental aspects of reachability and distributed routing on very large scale networks. Issues like fast recovering from network outages, bounding the end-to-end delay, or reducing the packet losses across the Internet for a given block of IP prefixes, are out of the reach of this model. The drawback, however, is that this approach results inadequate to handle most of the emerging demands and expected functionalities over the Internet. It is here, at the inter-domain level, were the majority of the work and research efforts are needed.

[1]Detailed descriptions of these limitations are given in Chapters 3 and 7.

1.2 Objectives

The subject of this thesis is to study the performance and reliability of communications between domains, with special focus on the design of strategies to control the inter-domain routing and TE decisions of a domain. More specifically, the objectives in this thesis are:

1. To analyze why – despite the efforts – it has proven to be so difficult to achieve deterministic end-to-end traffic performance when the control of the routing and TE decisions is performed either by BGP or by extended versions of it.

2. By taking advantage of the lessons learned, to propose a set of alternative inter-domain routing and TE control strategies – including both algorithms and architectures – that could be suitably adopted at different points in time, according to the expected needs and evolution of the Internet. To be precise:

    - At present, in legacy multi-domain IP networks.
    - In the near future, in IP/MPLS multi-domain networks.
    - In the future, in wavelength routed multi-domain optical networks.

The reason for introducing solutions in three different time frames is to leverage the "gradual" transition towards more advanced inter-domain routing and TE control models. In particular, this approach facilitates the gradual replacement of BGP – which is a realistic position given its wide deployment. The transitions proposed in this thesis are the following. First, we start with the current multi-domain network model, where an incremental solution reducing the deficiencies of BGP route control is proposed. Second, in the case of IP/MPLS multi-domain networks, a mixed scenario is studied. In this setting, BGP still controls the routing of legacy IP traffic, but not how IP/MPLS paths are routed (the latter are handled independently from BGP). And third, we propose a multi-domain network paradigm for optical networks, in which BGP has been completely replaced. In what follows, a summary of these strategies and the key contributions at each step of the evolution is given.

1.3 Strategies adopted and contributions

1.3.1 At Present: in multi-domain IP networks

The aims and requirements in terms of routing and TE control of transit domains are quite different from those of stub (i.e. non-transit) domains. On the one hand, transit domains focus on the aggregation and management of large amounts of traffic coming from many different sources. Transit domains are not aware of the end-to-end performance or the reliability features of the communications traversing their networks. On the other hand, most of the inter-domain communications are carried out between stub domains. These latter are the ones that, above all, are presently in need of cost-effective strategies aimed at optimizing the end-to-end performance and reliability of their inter-domain communications[2].

A widespread practice exploited by stub domains is multihoming, which consists of using multiple external links to connect to different Internet Service Providers (ISPs). By increasing their connectivity to the Internet, stub domains can potentially obtain several benefits, especially, in terms of resilience, cost, and traffic performance. These are indeed potential benefits, since multihoming by itself is unable to guarantee the improvement of any of them. Thus, additional mechanisms are needed to accomplish such improvements. In particular, when an online mechanism actively controls how the traffic is distributed and routed among the different links connecting a stub network to the Internet, it is referred to as *intelligent* or *smart route control*.

Several manufacturers are developing and offering Intelligent Route Controllers (IRCs) [10, 22, 58], which are being increasingly adopted by multihomed stub domains. These solutions rely on TE strategies operating in relatively short timescales[3], using measurement-driven dynamic path switching techniques. Without getting here into the details of the operation of IRCs – these are described in Chapter 4 – it is worth highlighting that they offer a powerful way to optimize the cost, reliability, and end-to-end performance of the outbound traffic of multihomed stub domains [4, 44]. IRCs are not intended to replace the existing BGP-based routing and TE model, but rather to complement it, offering an incremental way of tackling some of the key deficiencies affecting inter-domain communications.

Despite these advantages, all the solutions available at present have in common two major drawbacks. The first and most important one is that they behave in a fully selfish way – this thesis shows that their performance

[2] In practice, optimizing only a subset of them is enough, since not all inter-domain communications have strong performance or reliability constraints.

[3] Even reaching the order of a few seconds.

significantly degrades when several of them compete for network resources. Second, all available solutions are standalone, so no routing control interactions exist between the domains exchanging the traffic. The consequences of this lack of interactions are rather coarse route control over the outbound traffic of a domain, and the inability to smartly control how traffic flows into a domain.

Contributions in multi-domain IP networks

Given the limitations exposed above, this thesis discusses the advantages of progressively extending the existing intelligent route control model, first, from standalone and selfish to a standalone and social model, and then, to a more advanced cooperative and social route control model. Two extensions are proposed:

- First, a novel route control algorithm is introduced. The proposal is to endow each controller with a social route control algorithm that adaptively restrains its selfishness. The algorithm proposed is capable of learning from and evolving together with the network dynamics.
- Second, a new protocol is introduced. This protocol leverages the interactions between controllers belonging to a pair of multihomed stub domains that exchange large amounts of traffic.

It shall be shown that with these two extensions it is possible to outperform the existing route control model. Indeed, when several route controllers compete for network resources, the conventional ones are outperformed by those proposed in this thesis and this becomes especially noticeable as the network utilization increases. Extensive simulations reveal that it is possible to reduce the number of path shifts approximately between 40% and 80% on average (depending on the load on the network), and still obtain better end-to-end traffic performance for delay-sensitive applications such as Voice over IP (VoIP) [137]. It will also be shown that the cooperative extension offers significant potential benefits in terms of inbound traffic control.

A key advantage is that the extensions proposed in this thesis can be installed and used today by simply performing software upgrades to any of the existing route control solutions.

1.3.2 Near Future: in IP/MPLS multi-domain networks

MPLS has become one of the most widely deployed technologies at the intra-domain level. At present, service providers have strong incentives to extend the reach of long-lived MPLS paths across domains. Although this has begun to be analyzed by the Internet Engineering Task Force (IETF), more specifically by the Path Computation Element (PCE) Working Group [97], the discussions are still in an early stage. Among the problems that remain unsolved are:

- How to find and establish high quality primary and protection inter-domain MPLS paths for mission-critical services subject to QoS constraints.
- How to solve the trade-off of exploiting as much as possible the advantages of having MPLS coverage for inter-domain traffic aggregates, against the extra cost that this coverage will represent to ISPs. More specifically, to efficiently solve the multi-objective problem of how a domain can maximize the MPLS coverage of its inter-domain traffic demands with minimum cost, subject to a budget and network capacity constraints.

Contributions in IP/MPLS multi-domain networks

This thesis explores the major limitations hindering the solution of these two problems in the context of the current inter-domain routing and TE model [139]. We describe the critical challenges to be faced, and provide solutions to both of them. The proposed solutions are based on the extension of the PCE-based architecture that was recently standardized by the IETF. It shall be shown that by broadening the concept of the PCE-based multi-domain network model, it is possible to overcome several of the existing limitations in terms of inter-domain routing and TE control.

In particular, a solution that we recently proposed in [113] addressing the first of the two problems exposed above will be described. This solution introduces an *Aggregated Representation* (AR) of a multi-domain network that captures the path diversity and part of the internal link state of each domain. The AR of the network is composed and advertised between the PCEs. These latter run a joint distributed routing protocol – decoupled from BGP – that exploits the AR of the network in order to find an optimal pair of link-disjoint paths between the source and destination nodes in

an efficient manner. The problem of optimally finding link-disjoint paths is studied in two different scenarios, namely, in a general multi-domain network setting, and in a multi-domain network subject to the common export policies imposed by customer-provider and peer relationships between routing domains[4].

By taking advantage of this extended PCE-based multi-domain network model, we formulate and efficiently solve the second problem exposed above (i.e. the multi-objective MPLS coverage problem). The main contributions in this thesis regarding this problem are the following:

- We show that based on realistic assumptions and with minor knowledge about the pricing schemes of transit providers, it is possible to steer the search of potential solutions towards specific regions in the objective space.
- We propose a novel Evolutionary Multi-objective Algorithm (EMA) that exploits this fact to find candidate solutions in the Pareto sense, in a fast and efficient way.
- We prove that the search and update strategy of this EMA guarantees the elitism of the candidates monotonically. We also prove that this algorithm converges, and that it is capable of finding an ε-approximate Pareto Set.
- The evaluation results show that the EMA proposed in this thesis is capable of obtaining much better coverages than another powerful and well-known EMA, namely, the Strength Pareto Evolutionary Algorithm 2 (SPEA2) [143], while complying with the network capacity and budget constraints.

A major advantage of the contributions in this part is that they can be easily generalized and applied to other problems where the trade-off of coverage vs. cost is critical.

1.3.3 Future: in multi-domain optical networks

Despite the well-known limitations of BGP, during the past few years some researchers have proposed to adopt an Optical Border Gateway Protocol (OBGP) as the future inter-domain routing protocol for optical networks [16, 39, 130, 131]. These proposals basically attempt to extend BGP so that

[4]Export policies are described in Chapter 2.

it can convey, update, and signal optical path information among OBGP neighbors.

Rather than extending BGP, a sound alternative is to address the known issues in the areas of routing and TE control from its foundations. In fact, future optical networks offer the opportunity to avoid inheriting the problems of the past and evolve towards more advanced multi-domain network models. This is precisely the approach followed in this thesis.

Contributions in multi-domain optical networks

Future on-demand lightpath provisioning networks will demand high reliability, flexible TE capabilities, and high performance for establishing and managing inter-domain lightpaths. A multi-domain network model mostly centered on the exchange of reachability information – as the one we have today – is not going to be enough. This is confirmed by a number of research initiatives recently started, like [40] and [41]. Accordingly, the proposals made in this thesis consider that neighboring domains are able to exchange both reachability and enriched TE information mainly consisting of aggregated path state information.

The main contributions in the areas of routing and TE control for multi-domain optical networks are the following:

- For the information exchange between domains we propose to introduce the concept of the *Inter-Domain Routing Agent* (*IDRA*) [141]. Each Routing Control Domain (RCD) allocates one or more of these agents, which are the ones in charge of computing paths, exchanging routing updates, and exchanging TE information among neighboring RCDs. The IDRAs are devised as standalone devices that control the inter-domain routing and TE decisions of a RCD, and they can be seen as the next-generation version of the extended PCEs proposed for IP/MPLS networks in this thesis. The IDRAs provide a fully distributed and decoupled (physically independent) control-plane paradigm.

- Three different RWA algorithms are proposed and contrasted, namely, OBGP+, Cost, and Cost+Kalman. OBGP+ is our improved version of OBGP. Cost on the other hand, offers a minimum-cost path RWA strategy efficiently exploiting the TE information exchanged between the IDRAs. Finally, Cost+Kalman enhances Cost, and it is based on the stochastic estimation of the Effective Number of Available Wavelengths (ENAW) along inter-domain paths [140]. We propose an approximate model to roughly estimate the ENAW on the paths across

multiple domains, and then refine this estimation by means of observations and an adaptive prediction-correction Kalman filtering process. The performance of these three different RWA strategies is contrasted against OBGP.

The results obtained validate the effectiveness of the algorithms proposed, and confirm that by estimating the wavelength occupancy prior to the RWA decision, the blocking ratio can be drastically reduced compared with the one obtained by OBGP.

Before concluding this section, it is worth highlighting that all the strategies proposed in this thesis have three things in common, which we call the D^3 rule: they are all *Detached, Decoupled and Distributed*.

First, the routing and/or TE control decisions are always taken by independent devices (Intelligent Route Controllers, Path Computation Elements or Inter-Domain Routing Agents) that are *detached* from the devices forwarding the traffic on the network. This approach leverages the evolution towards more advanced and powerful routing and TE control models. The key is to release the traffic forwarders from the burden of exchanging control information and performing complex computations based on it.

The second thing that these strategies have in common is that the performance and reliability of inter-domain communications are always handled by means of a routing and/or a TE control process that is *decoupled* from BGP. In the cases of IP and IP/MPLS networks, BGP is used for exchanging reachability information and updating either topology or policy changes, but the dynamics of the routing and TE decisions are controlled by algorithms that are not embedded in BGP. In the case of optical networks, this is trivial since BGP is not present in the IDRA-based multi-domain network model.

Finally, the third aspect that these strategies have in common is that they are all supported by a fully *distributed* routing and TE control model. This distribution is two-fold: between domains and inside domains, since each domain may allocate several of the proposed devices, which can be arranged and configured in a distributed way as well.

1.4 Thesis structure

The thesis is organized in five parts. In the sequel, we overview the contents of each part and the topics addressed in its corresponding chapters.

PART I

After a brief summary of the aims and contributions in Chapter 1, the subsequent chapters in this part provide the necessary background to understand the issues addressed and the reach of the solutions proposed along this thesis. The readers who are familiar with the current structure of the Internet and with BGP routing can skip Chapter 2. Those who are also familiar with the reasons why the previous attempts to improve the performance and reliability of inter-domain communications by means of BGP have failed, might skip Chapter 3 as well. On the other hand, a comprehensive list of references is provided in this part for the reader who wants to get deeper into the details (and issues) of inter-domain routing and TE.

Chapter 2 introduces the basics of the current inter-domain network model. It explains the fundamental concepts of routing in the Internet, including its policy-based nature and a description of the main features of BGP. In particular, the BGP decision process (i.e. the process controlling the routing of packets though the network) is analyzed.

Chapter 3 exposes the major limitations of the current inter-domain network model, and supports why it is so difficult to improve the performance and reliability of inter-domain communications using BGP-based techniques. This chapter also overviews some of the most compelling proposals made so far to endow BGP with improved TE and routing control capabilities. The limitations exposed along the chapter, should help the reader understand why none of these proposals is widely used in practice.

PART II

This part deals with the problem of inter-domain routing and TE control at present, with focus on the analysis, design, and test of incremental solutions targeting multihomed stub domains. In particular, the strengths and weaknesses of IRC techniques are described here (Chapter 4). Two different solutions tackling the main weaknesses of the current IRC model are proposed in this part. First, a standalone and social route control algorithm

is introduced (Chapter 5). Next, in Chapter 6, a cooperative route control model exploiting the advantages of the social algorithm is proposed.

Chapter 4 exposes why multihomed stub domains are the ones that mainly need of new mechanisms tending to improve the performance and reliability of their inter-domain communications. The basics of IRC are described here. This chapter also reviews related work and remarks the most important deficiencies in the existing IRC model. These deficiencies are progressively tackled along the next two chapters.

Chapter 5 proposes to move from the conventional standalone and selfish IRC model to a standalone and social route control model. To this end, a Social Route Control (SRC) algorithm is introduced. This algorithm is supported by an adaptive cost metric and a two-stage filtering process. The strengths of this approach are evaluated by means of extensive simulations.

Chapter 6 introduces a Cooperative Route Control (CRC) model supported by a CRC protocol. By taking advantage of the social algorithm introduced in Chapter 5, a new Cooperative and Social Route Control (CSRC) algorithm is proposed. The performance evaluations in this chapter confirm the strengths of the CSRC algorithm, and the potential benefits of the CRC model in terms of inbound traffic control for multihomed stub domains.

PART III

This part focuses on the near future of routing and TE control in multi-domain networks, exploring the possibilities offered by IP/MPLS in the areas of performance and reliability for inter-domain communications.

Chapter 7 reviews the main drivers behind the extension of MPLS at the multi-domain level. The chapter deepens in the analysis of some of the key limitations imposed by the current multi-domain routing and TE control model in order to improve the performance and reliability of the communications between domains. Among the problems analyzed are, the scarcity of paths, and the lack of an inter-domain TE information exchange model. The IETF's PCE is introduced here. Finally, the major challenges to be faced by the PCE-based routing and TE control model are examined.

Chapter 8 addresses the problem of finding and establishing high quality primary and backup MPLS paths that need to traverse multiple domains.

The approach proposed in this chapter is to exploit the PCE-based architecture, and introduce a distributed routing algorithm – running between the PCEs – that allows to optimally find two link-disjoint paths directly from the PCE in the source domain. This problem is analyzed in two different contexts, namely, in a general multi-domain network (free from the export-policy constraints), and under the usual export-policies ruling the routing advertisements and forwarding characteristics between domains.

Chapter 9 addresses the trade-off of MPLS coverage vs. cost, subject to a budget and network capacity constraints. The problem is formulated as a multi-objective integer program, and solved by means of a new evolutionary multi-objective algorithm, named MC^2 (*Maximum Coverage at minimum Cost*). The strengths of this latter lie in the search and update engine proposed, which allows to steer the search of potential solutions in the objective space in a highly efficient way.

PART IV

This part addresses the problem of inter-domain routing and TE control in future wavelength routed optical networks. An advanced routing and TE control model tackling some of the major problems in BGP is proposed. Three different RWA algorithms are also introduced and tested in this part.

Chapter 10 describes the strengths and weaknesses of the existing proposals for the optical extension of BGP (i.e. OBGP), as well as others tending to solve the RWA problem between different domains. A discussion about why it is so important to avoid inheriting the problems of the past is also provided in this chapter.

Chapter 11 proposes a novel inter-domain routing and TE control model, supported by a set of agents named IDRAs. A new network architecture, as well as the routing and TE information exchange model are described in this chapter.

Chapter 12 introduces three different RWA algorithms: Cost, Cost+Kalman, and OBGP+. The first two are devised for the IDRA-based architecture, while the third is an enhanced version of OBGP. The chapter also proposes a stochastic estimation method of the wavelength availability along inter-domain paths, which is key in the development of the best of all three RWA algorithms, namely, Cost+Kalman. The performance of these algorithms is

also compared in this chapter by means of extensive simulations using OPNET Modeler [92].

PART V

This last part presents the conclusions that can be drawn from this thesis and analyzes future lines of work.

Chapter 13 highlights the main conclusions of this thesis.

Chapter 14 proposes several ways for extending the reach of the work done in this thesis. The chapter also anticipates that part of these proposals are already in our research agenda.

Chapter 2

Background

2.1 The basics of inter-domain routing

The current Internet is a decentralized collection of computer networks from all around the world. Each of these networks is typically known as a domain or Autonomous System (AS). An AS is in fact a network or a group of networks under a common routing policy, and managed by a single authority. Today, the Internet is basically the interconnection of more than 26000 ASs [21]. Every one of these ASs usually uses one or more Interior Gateway Protocols (IGPs) such as the Intermediate System to Intermediate System (IS-IS) [61] or the Open Shortest Path First (OSPF) [95] for the exchange of routing information within the AS. This is known as intra-domain routing. On the other hand, inter-domain routing focuses on the exchange of routes to allow the transmission of packets between different ASs.

Figure 2.1 illustrates a simplified (but typical) inter-domain scenario depicting the interconnection of several ASs. All the ASs represented in the figure have multiple connections to the network. This is indeed a common practice nowadays, and it is mainly used for resilience, and load balancing reasons. When an AS is connected to multiple different ASs, it is referred to as a multihomed AS. On the other hand, the ASs connected to a single AS are known as single-homed ASs. To fix ideas, all the ASs present in Fig. 2.1 are multihomed except AS3. Even though AS3 is dually-connected to the Internet, both connections are with the same AS (AS31).

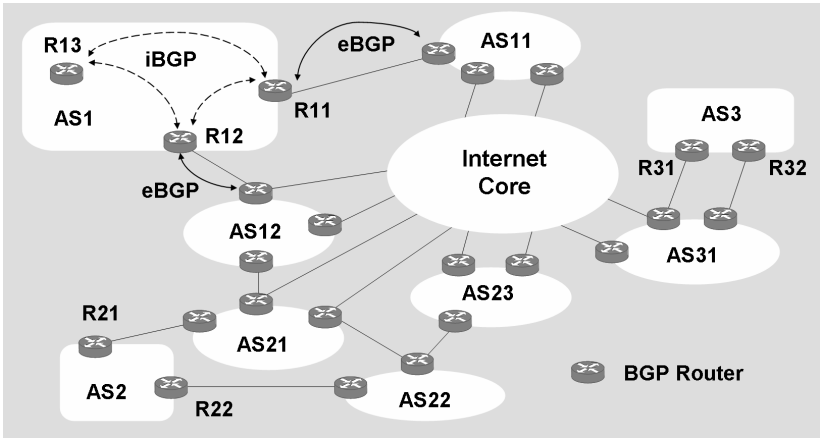


Figure 2.1: A simplified inter-domain scenario.

The Internet is composed by three different types of ASs, namely:

(i) Single-homed stub ASs, such as AS3 in Fig. 2.1.

(ii) Multihomed stub ASs, such as AS1 and AS2 in Fig. 2.1.

(iii) Transit ASs, which can be classified into very large transit ASs composing what is usually referred to as the Internet core, and smaller-sized transit ASs such as AS11, AS12, AS21-AS23, and AS31 in Fig. 2.1.

The two types of stub ASs crowd together mostly enterprise customers, universities, Content Service Providers (CSPs), and small Network Service Providers (NSPs). These two groups of stub domains represent approximately 85% of the ASs present in the Internet [101]. The third type gathers most of the Internet Service Providers (ISPs).

In today's Internet, there is a hierarchy of transit ASs [116]. This hierarchical structure is rooted in the two different types of relationships that could exist between ASs, i.e., a customer-provider or a peer-to-peer relationship. Thus, for each transit AS any directly connected AS is either a customer or a peer. At the top of this hierarchy we found the largest ISPs, which are usually referred to as Tier-1 ISPs. There are about 20 Tier-1s at present [116], which represents less than 0.1% of the total number of ASs in the Internet [21]. These Tier-1s are directly interconnected in almost a full-mesh, and compose the Internet core. In the core, all relationships between Tier-1s are peer-to-peer, so a Tier-1 is any ISP lacking of an upstream provider. The second level of the hierarchy is composed by Tier-2 ISPs. A Tier-2 is any transit AS which is a customer of one or more Tier-1 ISPs. A representative

example of a Tier-2 ISP is a national service provider. Tier-2 ISPs tend to establish peer-to-peer relationships with other neighboring Tier-2s, for both economical and performance reasons. This is typically the case for geographically close Tier-2 ISPs that exchange large amounts of traffic. There are also Tier-3 ISPs, which are those transit ASs in the hierarchy that are customers of one or more Tier-2 ISP, such as regional ISPs within a country. Stub ASs are non-transit ASs which are customers of any ISP (Tier-1, Tier-2 or Tier-3). In Fig. 2.1 ISPs such as AS11, AS12, AS21, AS23 and AS31 would be classified as Tier2 ISPs, while AS22 represents a Tier-3 ISP. An important corollary of this hierarchical structure is that the diameter of the Internet is very small in terms of AS hops.

2.2 BGP

The Border Gateway Protocol (BGP) is currently the de facto standard inter-domain routing protocol in the Internet. Its current official release is BGP-4, which was specified in [106] on March of 1995[1]. BGP is used to exchange reachability information throughout the Internet and it is mainly an inter-AS routing protocol. However, the reachability information that an AS learns from the exterior needs to be distributed within the AS so that every router in the AS could properly reach destinations outside the AS. When reachability information is exchanged between two BGP routers located in different ASs the protocol is referred to as external BGP (eBGP). On the other hand, when reachability information is exchanged between BGP routers located inside the same AS, the protocol is referred to as internal BGP (iBGP).

For instance, in AS1 in Fig. 2.1, the reachability information that R11 learns from AS11 is received over eBGP. This information is passed from R11 to the routers inside AS1 (i.e., R12 and R13) so that they could be able to reach the routes advertised by AS11. This exchange of reachability information between R11 and the internal routers in AS1 is done by means of iBGP. The same occurs for the external routes that R12 learns from AS12.

For scalability reasons BGP does not try to keep track of the entire Internet's topology. Instead, it only manages the end-to-end AS-path of one route in the form of an ordered sequence of AS numbers. For this reason BGP is known as a path vector routing protocol, to reflect the fact that it is essentially a modified distance vector protocol. While a typical distance vector protocol like the Routing Information Protocol (RIP) [52, 76] chooses a route according to the least number of routers traversed (router hops),

[1]The Inter-Domain Routing (IDR) working group of the IETF has finalized the revision of [106]. This revision documents the currently deployed code.

BGP "generally" chooses the route that traverses the least number of ASs (AS hops). For example, the BGP process running in router R21 will typically choose to reach AS1 via the ASs AS21 and AS12. Thus, the AS-path chosen by R21 is {AS21, AS12, AS1} (please notice that the Internet core accounts for at least one AS hop more in the AS-path if only one Tier-1 ISP is traversed while reaching AS1).

The term "generally" mentioned before is due to the fact that the AS-path length is one of the steps of the BGP decision process, but not the only one. This decision process is used for route selection each time a BGP router has at least two different routes for the same destination. Thus, BGP routing is more complex than simply minimizing the number of AS hops. BGP routers have inbuilt features to override the AS hop count, and to tiebreak if two or more routes have the same AS-path length. The details of the BGP decision process are described in next section.

2.2.1 The BGP decision process

Most ASs have chosen to increment their connectivity to the Internet, mainly, for resilience and load balancing reasons. As mentioned before, this practice of connecting to multiple ISPs is known as multihoming. Due to this practice, BGP routers typically have multiple candidate paths to reach a destination. In such cases, BGP will need to choose the best path among the candidate set of routes. The algorithm that a BGP router runs to make such selection is referred to as the BGP decision process. The sequence of steps in Algorithm 1 represents a simplified version of the BGP decision process.

In this process each subsequent step is used to break ties when the routes being compared were equally good in the previous step. The Local Preference (LOCAL_PREF) in step 1 and the Multi-Exit Discriminator (MED) in step 3 are two BGP attributes, which are used by BGP routers for controlling how traffic flows from and into an AS, respectively. In the sequel, we will describe these BGP attributes by means of the example in Fig. 2.2. We will illustrate their role, and particularly, their use in practice during the BGP decision process. The reader who wants to go deeper into the details of the BGP decision process and its attributes can consult [50, 106].

Let us now consider the example in Fig. 2.2. We are interested in analyzing the routing of packets sourced by AS2 and AS3 towards the multihomed stub domain AS1. Our focus is on the BGP decision process running on the border routers of AS2 and AS3. This simple example will help to illustrate how these two ASs can choose the best route to reach AS1, and also, the mechanisms that AS1 has in order to load-balance its inbound traffic.

Algorithm 1 Simplified version of the BGP decision process

Input: A set of candidate routes to reach a destination d
Output: The best route to destination d

1: Choose the route with the highest local preference (LOCAL_PREF)
2: If the LOCAL_PREFs are equal choose the route with the shortest AS-path
3: If the AS-path lengths are equal choose the route with the lowest MED
4: If the MEDs are equal prefer external routes over internal routes (i.e. eBGP over iBGP)
5: If the routes are still equal prefer the one with the lowest IGP metric to the next-hop router
6: If more than one route is still available run tie-breaking rules

Assume that AS1 originates two IP prefixes 194.100.80.0/20 (obtained from AS2's block of IP prefixes) and 200.2.160.0/20 (obtained from AS5). In order to load balance its inbound traffic and to count with a fault-tolerant routing scheme, AS1 seeks the following goals:

(i) Traffic targeting 194.100.80.0/20 should primarily enter AS1 via AS2 and use AS5 as a backup path.

(ii) Traffic targeting 200.2.160.0/20 should primarily enter AS1 via AS5 and use AS2 as a backup path.

(iii) Traffic targeting 194.100.80.0/20 coming from AS2, should primarily enter AS1 via router R12 and use the route via R11 as a backup route.

(iv) Traffic targeting 200.2.160.0/20 coming from AS2, should primarily enter AS1 via router R11 and use the route via R12 as a backup route.

Figure 2.2 shows the BGP advertisements sent by AS1. To accomplish goals (i) and (ii), AS1 selectively prepends its own AS number in its BGP advertisements. The aim is to increase the AS-path length for the specific prefixes, and hence, influence the selection of the best route in upstream ASs[2]. On the other hand, to achieve goals (iii) and (iv) AS1 sets different values of the MED attribute in its BGP advertisements to AS2. The MED

[2]It is worth mentioning that even though prepending is widely used in operational networks to influence how traffic enters an AS, for several reasons, it does not always work. These reasons will be addressed in the rest of the example and also in Chapter 3.

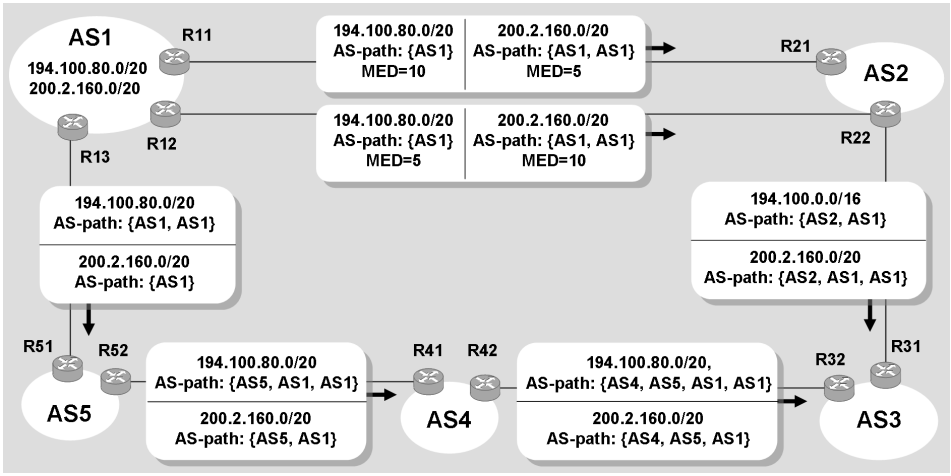


Figure 2.2: Example of the BGP decision process.

attribute offers AS1 a way to influence a neighboring AS (AS2 in this example) regarding the preferred route into AS1. During the BGP decision process (see Algorithm 1), if all other factors are equal in steps 1 and 2, the route with the lowest MED should be preferred. According to the BGP advertisements sent by AS1, AS2 should prefer to reach 194.100.80.0/20 via R12, but for 200.2.160.0/20 it should prefer entering AS1 via R11. The MED is an optional non-transitive attribute, which means that not all implementations of BGP must support it (the optional part), and also that it must never be propagated to other neighboring domains (the non-transitive part).

In practice, the role of the MED attribute is to suggest a neighboring domain which is the preferred ingress link to the AS, when more than one option exists. It is important to notice that this is in fact a suggestion, given that the neighboring AS might not always choose that route. To clarify this point, let us analyze the BGP decision process for the egress routers in AS2. The MED attribute in the BGP advertisements received from AS1 will not be assessed until step 3 in Algorithm 1. If more than one candidate route is available, the first attribute assessed by the BGP routers in AS2 is the LOCAL_PREF. The role of this latter is to choose the egress point from an AS. To be precise, the egress point with the highest configured LOCAL_PREF is the one preferred, and it can be independently configured for different destination prefixes.

An issue at present is that there is no mechanism in practice to coordinate the ingress point suggested by AS1 and the egress point chosen by AS2. Given that the LOCAL_PREFs are assessed in the first step of Algorithm 1, if the configured Local Preferences in R21 and R22 for the prefix 194.100.80.0/20

are such that:

$$LOCAL\_PREF(R22) \geq LOCAL\_PREF(R21) \tag{2.1}$$

then AS1 will be able to accomplish goal (iii). If on the contrary:

$$LOCAL\_PREF(R22) < LOCAL\_PREF(R21) \tag{2.2}$$

then AS1 will not be able to achieve goal (iii). The same reasoning can be applied to the case of prefix 200.2.160.0/20 and goal number (iv) of AS1.

Let us now focus on the AS-path length and the analysis of the second step in Algorithm 1. To this end, we will examine the BGP decision process of the border routers in AS3, namely, R31 and R32. We assume that the configured LOCAL_PREFs in R31 and R32 are the same for AS1's prefixes, so the second step in Algorithm 1 is always reached by R31 and R32 while sending traffic to AS1. We also assume that AS2 and AS5 are configured differently. Whereas AS5 simply propagates the two BGP advertisements received from AS1, AS2 sends an aggregate advertisement for 194.100.0.0/16. As this prefix includes 194.100.80.0/20, the advertisement received from AS1 is not propagated. This is typically the case when a customer advertises a prefix that belongs to one of its ISP's block of prefixes. In such a case the ISP could aggregate the customer advertisement into a shorter prefix when advertising the prefix to other customers or peers.

As shown in Fig. 2.2, even though AS1 originates only two prefixes AS3 receives four routes for three different prefixes: 194.100.80.0/20 (from AS4), 194.100.0.0/16 (from AS2) and 200.2.160.0/20 (from both AS4 and AS2). The border routers R31 and R32 exchange the external routes received by means of iBGP. After R31 and R32 have learned these candidate routes, they run Algorithm 1 in order to choose the best path to reach the prefixes in AS1. Despite the prepending operation performed by AS1, all traffic from AS3 towards 194.100.80.0/20 in AS1 will be routed via AS4. This is because a BGP router always prefers the most specific (i.e. the longest) prefix when forwarding packets – independently of the AS-path length. In such conditions, AS2 will usually stop aggregating AS1's prefixes so that AS1 could start receiving traffic for 194.100.80.0/20 via AS2. This disaggregation causes that AS2 advertises to AS4 two prefixes, the customer's prefix 194.100.80.0/20 and the aggregate 194.100.80.0/16 with an additional increment in the size of the BGP routing tables.

On the other hand, the decision of how to reach prefix 200.2.160.0/20 from AS3 is slightly more complex. For this prefix, the AS-path length

observed by R31 and R32 is the same – they both observe three AS-hops due to the prepending operation made by AS1 – so the BGP decision process reaches step 4 for both of them in Algorithm 1. Moreover, the BGP routers inside AS3 (not depicted in the figure) will receive two advertisements, one from R31 and another from R32. Since both of these advertisements are sent via iBGP, the routers inside AS3 will reach step 5 in Algorithm 1 for the considered prefix. Steps 4 and 5 in Algorithm 1 basically mean that the routing criterion is to try to get rid of the packets from the AS as fast as possible, which for transit ASs this is typically determined by the intra-domain routing protocol running on the AS (step 5 in Algorithm 1). This routing practice is commonly used by transit providers in the Internet, and is known as hot potato routing [2]

Overall, even from the simple example shown in Fig. 2.2 it can be concluded that the selection of a route by means of BGP is a rather complex process. And most important of all, BGP only offers very limited ways of controlling and influencing this process. This is especially evident in the case of inbound traffic control.

2.3 Export-Policies

In order to understand the way inter-domain routing information flows in the Internet, as well as the basic content of this information, it is mandatory to introduce first the business relationships between ASs. There are two major types of business relationships, i.e., customer-provider and peer-to-peer, which correspond to the two different traffic exchange agreements between neighboring domains. The former applies when a domain buys Internet connectivity from an ISP. The latter typically applies when two providers that exchange a significant amount of traffic, agree to connect directly to each other to avoid transiting through a third-party provider. Peering domains share the costs of the connection between them, so there is no customer-provider relationship in this case. These two types of relationships imply the following usual export policies of the ASs [135].

Customer-Provider Advertisements:

- Each AS advertises to its providers all its allocated IP prefixes and those learned from its own customers, but never those learned from its peers or from other providers.
- Each AS advertises to its customers all the reachable IP prefixes it knows (or sometimes only a default route).

Peer-to-Peer Advertisements:

- Each AS advertises to its peers its own IP prefixes as well as those learned from its customers, but never those learned from its providers or other peers.

These export policies determine the inter-domain routing preferences of a provider as follows. A provider prefers customer routes over peer routes or higher hierarchy provider routes, independently of the AS-path length. Moreover, a provider always prefers peer routes over higher hierarchy provider routes. Clearly, the routing across a multi-domain network is governed by the export policies. To better understand these export policies let us illustrate their effects on the way domains forward traffic to other domains [113]. For any two neighboring routing domains D_i and D_j, one of the three following cases hold: (i) D_i is a provider of D_j and D_j is a customer of D_i; (ii) D_j is a provider of D_i and D_i is a customer of D_j; (iii) D_i and D_j are peers.

The export policies impose the following constraints on the forwarding policy.

- Suppose that D_i is a customer of D_j. Then, D_i can forward packets received from D_j to its customers, but never to its peers or other providers.

- Suppose that D_i is a provider of D_j. Then, D_i can forward packets received from D_j to its customers, providers and peers.

- Suppose that D_i is a peer of D_j. Then, D_i can forward packets received from D_j to its customers but never to its providers or peers.

Let D_i, D_j, and D_k be three domains such that D_i is connected to D_j and D_j is connected to D_k. Table 2.1 summarizes the conditions under which D_j can forward the traffic received from D_i to D_k.

	D_j is a customer of D_k	D_j is a provider of D_k	D_j is a peer of D_k
D_i is a customer of D_j	Yes	Yes	Yes
D_i is a provider of D_j	No	Yes	No
D_i is a peer of D_j	No	Yes	No

Table 2.1: The usual export policies.

After this short description of the main components and their roles in inter-domain routing, we follow in next chapter with some of the major limitations imposed by the current inter-domain network model in terms of routing and TE control.

Chapter 3

Internet Route Control: Past

The performance and reliability of the communications over the Internet strongly depend on the routing process that determines how packets are treated and forwarded through network. The accurate control of this process in a multi-domain network is considered a challenging research area [9, 82, 138]. This is mainly rooted in the following two facts:

(i) The inter-domain routing protocol currently used in the Internet has several limitations, but its replacement is not a realistic option due to its worldwide deployment. These limitations are becoming especially noticeable given the explosive growth that the network has experienced in these last few years [57, 21]. For instance, Fig. 3.1 illustrates the growth in terms of the number of ASs composing the network (data obtained from [21] showing the evolution from 1996 up to the present). Similarly, Fig. 3.2 shows the number of entries in the Forwarding Information Base (FIB) of a typical BGP router owned by a Tier-1 ISP (data obtained from [21] showing the evolution from 1989 up to the present). The "explosive" growth not only refers to the size of the network, but also to the amount of and variety of the applications actually available on the Internet. This growth tendency is placing significant stress on the capabilities of the inter-domain routing protocol.

(ii) As its name indicates, inter-domain routing denotes routing among distinct domains or networks. These domains are completely autonomous entities, which perform their own routing management based on policies that only have local significance. In this scenario, conditions such as business and competition between domains, along with fully independent management using potentially conflicting policies, makes the problem of accurately controlling inter-domain routing and the effects of TE even harder.

The goals of this chapter are, first to present an up-to-date inspection of some of the main issues in the areas of inter-domain routing and TE control. Second, we intend to survey the state of the art and briefly describe some of

the most relevant proposals made in these areas. Third, we seek to point out why these issues are so difficult to solve at present, and succinctly explain why most of the existing proposals have never moved into a deployment stage. Our aim is to put things into perspective and summarize here the lessons learned so far.

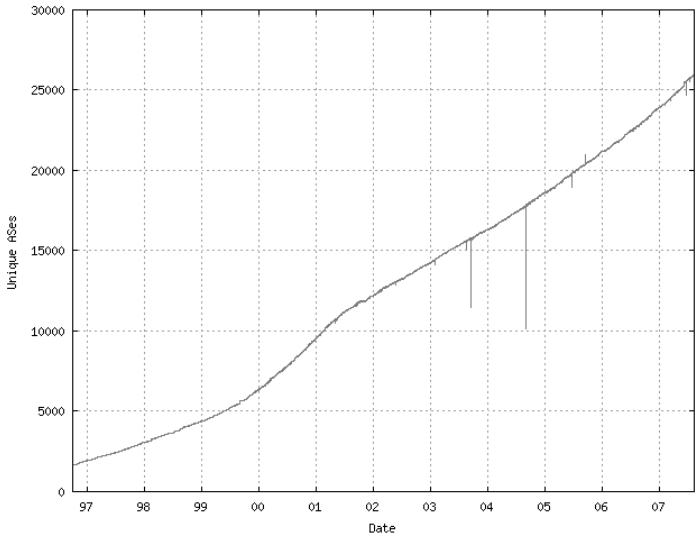


Figure 3.1: Growth in the number of ASs.

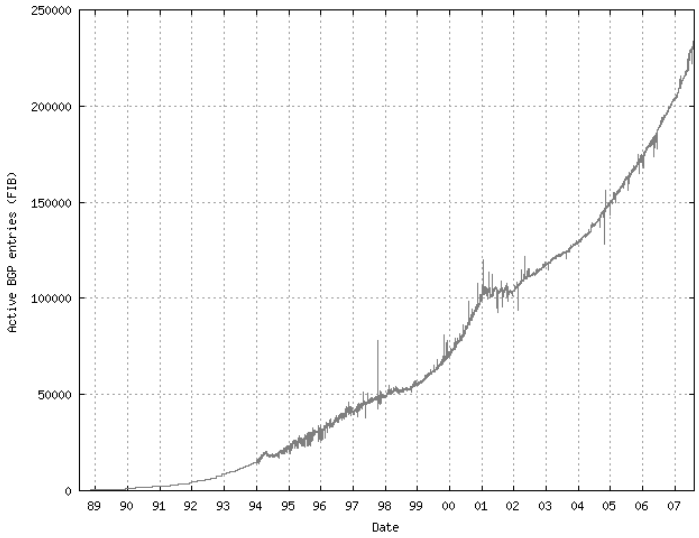


Figure 3.2: Growth in the number of entries in the FIB.

3.1 Major limitations

The current inter-domain routing structure is not prepared to handle the service characteristics that several applications are demanding from the network. In effect, the end-to-end performance of these applications is not only affected by the limitations of BGP, but also by the diversity of interests and lack of cooperation between the ASs composing the Internet. Therefore, several issues remain to be solved in the areas of inter-domain routing and TE control. This chapter analyzes some of the most important challenges faced by researchers in these areas today. The methodology that we follow is first to introduce the problem. Next, we survey several proposals addressing the issue, and try to discriminate which are in fact operational palliatives. After that, we discuss why despite many efforts each of the issues exposed remains largely unsolved.

The order in which the issues are presented is chosen so as to gradually introduce the different aspects of BGP and the inter-domain network paradigm, as well as to link how the initial set of issues influences the subsequent ones.

3.1.1 Slow Convergence and Chattiness of BGP

In order to exchange reachability information two BGP routers must establish a BGP session. This session is supported by a TCP connection through which the peers exchange four different types of messages, specifically [106]:

(i) OPEN message: to open a BGP session between two peers.

(ii) UPDATE message: to transfer reachability information among the peers. This message is used either to advertise a feasible route to a peer or to withdraw unfeasible routes. The UPDATE message is usually referred to as a BGP advertisement.

(iii) NOTIFICATION message: sent when an error condition is detected. The BGP session is immediately shutdown after this message is sent.

(iv) KEEPALIVE message: periodically exchanged to verify that the peer is still reachable.

Each peer is able to determine if the BGP session corresponds to an iBGP or an eBGP session from the content of the OPEN message. When a BGP session starts, each peer advertises its entire set of routes. After that, only incremental updates and KEEPALIVE messages are exchanged.

An important performance metric for a routing protocol is its convergence time, i.e. the time required to reroute packets around a failure. The

first significant studies of the convergence of BGP were carried out using measurements in the Internet [68]. These studies showed that the convergence of BGP was rather slow, often measured in tens of seconds. This slow convergence is caused by several factors, some of which are inherent to the utilization of path vectors by BGP, while others are due to implementation choices. In short, this slow convergence is mainly rooted in the fact that in the global Internet a single link failure can force all BGP routers to exchange large amounts of BGP advertisements, while exploring for alternative paths toward the affected destinations. This process is referred to as path exploration.

During a BGP convergence, routers may need to exchange several advertisements concerning the same prefix. To avoid storms of BGP advertisements, most BGP routers use a timer called Minimum Route Advertisement Interval (MRAI), with a recommended default value of 30 seconds. This timer prevents BGP routers from sending a new advertisement for one prefix, if the previous advertisement for the prefix was sent less than 30 seconds earlier [106]. This reduces the number of BGP advertisements exchanged, but may cause important BGP advertisements to be unnecessarily delayed. Griffin and Presmore showed in [45] that this arbitrary 30 seconds value has a huge impact on BGP convergence time. They observed that for each network topology and for a particular set of experiments there is an optimal value of the MRAI timer. This optimal value can significantly reduce the convergence time of BGP. Unfortunately, this might be extremely hard to find in practice since it varies from network to network.

To cope with flapping routers that regularly advertise and shortly after withdraw their routes, many routers implement BGP route flap damping [127]. This technique works by ignoring routes that change too often. This is necessary to avoid storms of advertisements due to flapping routers, but unfortunately this increases the BGP convergence time [77].

Several authors have proposed modifications to reduce the BGP convergence time in case of failures. The ghost-flushing approach, proposed in [18] improves the BGP convergence by ensuring that the messages indicating bad news are distributed quickly by the BGP routers, while good news propagate slower. The downside of ghost-flushing is that it does not tackle the root of the problem, i.e., path exploration. Instead, it only tries to speed up the convergence of BGP.

Other solutions such as BGP-RCN [98] and EPIC [19] improve the convergence of BGP and also reduce the number of BGP messages exchanged during the convergence by adding to each BGP message an identifier (root-cause) indicating the cause of the BGP message. With this additional information, when a failure occurs on one link, distant routers can avoid to select as their

alternate path a path that is also affected by the failure – but for which they have not received up-to-date information yet.

The good news is that these proposals significantly limit path exploration. The bad news is that accurately identifying the root-cause of a failure still represents a challenging problem. This is first because root-cause approaches require modifying BGP to add information in the BGP advertisements, but ISPs are cautious about upgrading BGP[1]. Second, they only introduce significant improvements under the assumption of extensive deployment. And most important of all, the additional information needed to identify the root-cause of a failure strikes against the scalability of BGP.

The explanation for this latter is that for scalability reasons the BGP advertisements spawned by ISPs are often aggregated. Two levels of aggregation exist in these advertisements. Firstly, the set of the destinations advertised by BGP routers are composed by IP prefixes which aggregate several routes into a single route[2]. Secondly, the AS-paths carried in the BGP advertisements intrinsically represent highly aggregated information, since they do not reveal any clue about the internal details of the ASs in the path (e.g. topology, state of connectivity, etc). While the first level of aggregation reduces the size of the BGP routing tables, the second tremendously reduces the amount of details exchanged between BGP routers. The downside is the loss of granularity in the reachability information that each BGP router manages. In this framework, pinpointing the source of a failure is almost impossible, given that different failures will produce the same BGP UPDATE message [19]. To cope with this, the BGP advertisements from ISPs should be somehow disaggregated, which unfortunately has a direct impact on BGP's scalability.

Clearly, two trade-offs exist: i) how to, and how much should the reachability information be disaggregated in the BGP advertisements so as to accurately identify the source of a failure; and more general ii) how much could the BGP convergence time be reduced while keeping the overall routing system scalable.

An interesting alternative to pinpoint the source of a failure without needing to modify BGP was proposed in [36]. Feldmann et al. propose to infer the precise location of a failure by analyzing its effects, i.e., by observing the flow of BGP UPDATE messages during a convergence process. This is achieved by using multiple observation points (known as vantage points) and correlating the data observed along three dimensions: time, vantage point, and prefixes. However, this work proposes an offline methodology to pinpoint

[1] A clear incentive to explore solutions decoupled from BGP.
[2] An example of this aggregation was shown in Chapter 2 (Fig. 2.2).

the source of a failure, so it was not devised as a mechanism to reduce the BGP convergence time.

Figure 3.3 depicts three major inter-domain routing objectives as well as how the set of mechanisms described before strengthen or weaken the accomplishment of these objectives. The figure shows that unfortunately none of the existing mechanisms is able to strengthen the accomplishment of some of the objectives without weakening the accomplishment of some other. From our perspective the issue remains largely unsolved, and it will remain in this state unless we thoroughly understand the intrinsic trade-offs between some of the objectives in Fig. 3.3, and based on this understanding we succeed in developing novel mechanisms that could timely balance the accomplishment of all the objectives at the same time.

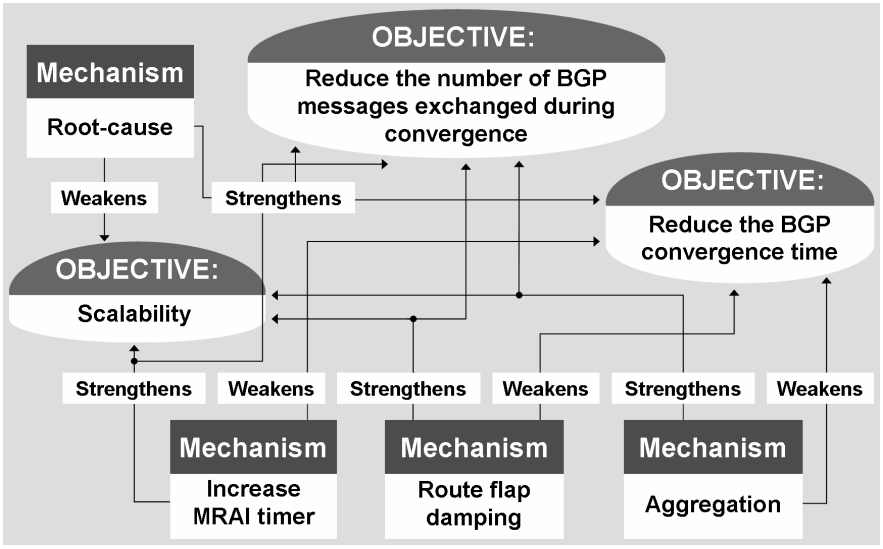


Figure 3.3: Unsolved balance between different objectives.

3.1.2 Expressiveness and Safety of Policies

Each AS in the Internet administrates its traffic in a completely autonomous way based on a set of policies which only have local significance to the AS. In other words, the way in which BGP routes are advertised through the global Internet, and the way in which routing is finally performed are the result of the application of several independently configured policies. This lack of global coordination between the policies used in the different domains is a major weakness of the current inter-domain routing paradigm.

Several studies such as [46, 47] have demonstrated that without coordination the interaction between independent policies may lead to global routing anomalies, such as inconsistent recovery from link failures or even route oscillations. Figure 3.4 depicts one of these routing anomalies. This particular configuration is known as "the bad gadget" [47], and illustrates how the policy based nature of BGP may lead to configurations that are guaranteed to diverge (i.e. BGP does not converge). In this configuration, the routing policies are such that each AS prefers the counterclockwise route to reach AS0, instead of the direct route. For example, AS2 prefers the route {AS2, AS1, AS0} over the route {AS2, AS0}. Given that AS1 and AS3 have analogous preferences, this configuration clearly causes the divergence of the BGP protocol.

In Section 3.1.1 we assumed the convergence of BGP as a fact, and based on it we exposed that the speed of this convergence is affected not only by the intrinsic properties of path vector routing protocols, but also by implementation decisions of BGP. The previous example shows that the convergence of BGP is indeed a much more complex and open problem, since managing routing based on independent policies causes that convergence cannot be assumed as a fact.

The main reasons for the absence of cooperation between domains are:

(i) The characteristics of the BGP policy expressiveness.

(ii) The ASs are not willing to disclose the details about their internal configuration and policies.

The expressiveness of policies is particularly tricky. On the one hand this expressiveness is rich enough to construct intricate local routing policies. Unfortunately, these policies may conflict with policies from other domains leading to the global routing problems described before. On the other hand, this expressiveness is not enough to attach information to a route so that it could be straightforwardly shared and used throughout the network.

It should become clear that both, the expressiveness of policies and the basis for autonomic management of policies able to guarantee robust convergence of the inter-domain routing protocol are in a very early stage of development. We need to thoroughly understand these two central aspects of distributed policies in order to balance the complex trade-off between allowing the ASs to disclose only the set of details they are willing to disclose, and guaranteeing robust convergence of BGP. A further discussion of these issues can be found in [63].

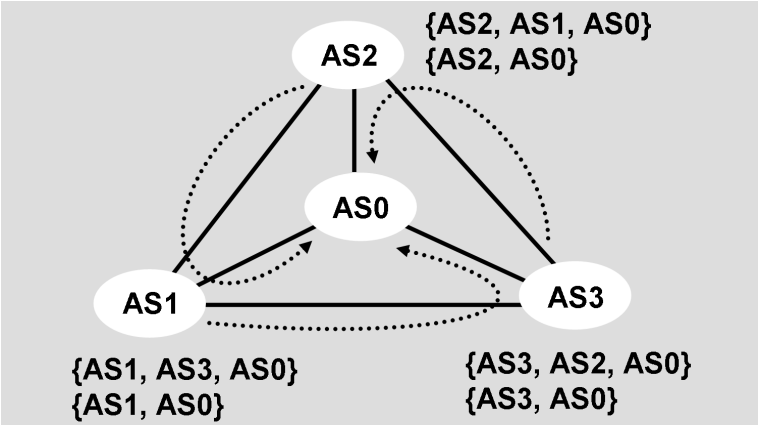


Figure 3.4: The bad gadget example.

3.1.3 Robustness of BGP Sessions

The exchange of messages among two BGP routers is supported by a TCP connection, which supplies a reliable transport layer for the communication between the routers. Despite this reliability, some previous studies showed that the resilience of BGP sessions was formerly affected by congestion. In 1999, Labovitz et al observed that KEEPALIVE messages were delayed during periods of peak network usage [69]. This led BGP sessions to fail when KEEPALIVE messages were delayed beyond the BGP hold timer [106]. Another previous study concerning the resilience of BGP sessions to congestion was presented in [111]. This study showed that increased queuing and delays had negative effects in the resilience of BGP. One of the main conclusions of [111] was the requisite to differentiate somehow the routing protocol messages from normal data traffic. For this reason, an operational palliative that several operators use at present is to prioritize BGP messages by setting their IP precedence to 7.

More recent work such as [134] shows that the conservative behavior of TCP retransmissions actually aggravates the instability of BGP sessions when network failures occur. The authors analyze the case of iBGP sessions, and propose a simple modification of TCP to increase the robustness of these sessions. However, the community remains cautious about upgrading TCP.

Furthermore, the robustness of BGP sessions is an important issue at present for security reasons. This is because a BGP session will fail if the TCP connection fails due to an attack (see [138] and the references therein).

In sum, it seems mandatory to find secure ways of guaranteeing the robustness and reliability of the exchange of routing control information between domains. And clearly, this issue should be addressed before developing

BGP-based control strategies aimed at improving the performance and reliability of inter-domain communications.

3.1.4 Lack of Multipath Routing

A BGP router could receive multiple advertisements for the same route from multiple sources. For instance, in Fig. 2.2 router R32 receives two advertisements for the prefix 200.2.160.0/20, and hence it will need to run its BGP decision process (Algorithm 1 in Chapter 2) to select the best path to reach this destination. In its current release BGP selects only one route as the best path, and this is the path that it places in the forwarding table. In addition, each BGP router only advertises to its peers the best route it knows to any given destination. Thus, in Fig. 2.2 router R32 will install in its forwarding table only one path for the prefix 200.2.160.0/20 (the path via AS4 in this example, step 4 of the BGP decision process), and this is the path it will advertise to its peers.

This behavior introduces mainly two important limitations. First, since the routing protocol only uses one best route, load balancing is not feasible even between paths presenting the same AS-path length. For this reason some vendors have developed and actually support multipath extensions in their BGP implementations. Despite this fact, only the best route is still advertised to other peers in all implementations. This is precisely the second and most important limitation. Given that a BGP router only advertises the best route it knows many alternative paths that could have been potentially used by any source of traffic will be unknown.

For example, a peer of R32 will receive an advertisement that the network 200.2.160.0/20 is reachable via {AS3, AS4, AS5, AS1}, but it will not know that the prefix is also reachable via the path {AS3, AS2, AS1, AS1}. This causes that the BGP messages received in an AS contain only a subset of all the available paths to the destination. This pruning behavior inherent to BGP introduces several limitations to the current inter-domain routing paradigm, especially, from the end-to-end QoS and TE viewpoints[3].

At present, efforts are being carried out so that a BGP router could be able to advertise to its peers multiple routes for the same destination. One of the most recent proposals can be found in [128].

Despite the limitations described before, it is very unclear how to endow BGP with multipath routing capabilities without deeply impacting on its scalability. If more routes are selected and advertised by BGP routers, then

[3]It is important to highlight that the shortest AS-path does not necessarily supply the best end-to-end traffic performance [55].

more entries will exist in the BGP routing tables increasing the problems exposed in Section 3.1.1.

3.1.5 Transit through an AS: iBGP issues

BGP is an inter-domain routing protocol, and as such, its main concern is the transmission of routes and packets between ASs. However, as an AS may contain thousands of routers, it is necessary to specify how the inter-domain routes and packets can transit an AS. When a border router learns a new inter-domain route, it needs to distribute this route to other routers inside its AS. This is done by sending the inter-domain routes over iBGP sessions inside the AS. If the AS is small, a full-mesh of iBGP sessions can be established between the BGP routers. If the AS is larger, route reflectors [12] or confederations [121] can be used to replace the unscalable iBGP full-mesh.

When a border router of a transit AS receives a packet whose destination is not local, it will consult its BGP routing table to determine the BGP next-hop. This latter is typically another border router inside the AS. Clearly, there can be several intermediate routers between the ingress border router and the egress border router. To ensure that an inter-domain packet will reach the BGP next-hop selected by the ingress border router, the transit AS must ensure that all intermediate routers will also select this next-hop.

This problem was discussed early during the development of BGP [105] and two techniques have emerged. The first solution, proposed in 1990, is to use encapsulation, i.e. the ingress border router encapsulates the inter-domain packets inside a tunnel towards the egress border router chosen by its BGP decision process. At that time, encapsulation suffered from a major performance drawback given the difficulty of performing encapsulation on the available routers. Today, high-end routers are capable of performing encapsulation and decapsulation at line rate when using MPLS or IP-based tunnels. The main advantage of using encapsulation is that the BGP forwarding table is consulted only once (by the ingress border router) per inter-domain packet inside each AS.

Unfortunately, this is not often a common practice in pure IP-based transit networks. This type of networks typically uses another technique called Pervasive BGP, which is to run BGP on all (border and non-border) routers inside the transit AS. As all intermediate routers must consult their BGP forwarding table for each inter-domain packet, there is a risk of deflection, or worse, routing loops, when the forwarding tables are not perfectly synchronized such as during a BGP convergence [114] or when route reflectors or confederations are used [48].

The main issue at present is that route reflectors and/or confederations have become absolutely necessary given the tremendous scalability they have supplied to large transit ASs. However, anomalies such as the ones described above can occur, especially in the event of a link or a router failure with Pervasive BGP. The question that arises is then: how can it be guaranteed that iBGP configurations remain highly scalable and anomaly-free at the same time?

3.2 BGP-based Traffic Engineering

The current inter-domain network model offers scarce TE capabilities for several reasons. First, BGP was designed to distribute mainly reachability information. Second, as exposed in Section 3.1.4 the inability of BGP to advertise multiple routes for the same destination limits the number and quality of the alternative paths that could be used to reroute packets around a failure. In addition, the limitations of BGP in terms of multipath routing restrict the possibilities of balancing traffic across domains to certain setups and vendor specific implementations.

On the other hand, in Section 3.1.2 we showed that the autonomic management of policies and the limitations in the expressiveness of these latter impose strong restrictions on how the ASs are able to control and manage the flow of their inter-domain traffic. For instance, even though BGP allows an AS to flexibly manage its outbound traffic, it exhibits a scarce degree of control in order to manage and balance how traffic enters an AS across multiple paths. In other words, accurately controlling inbound traffic with BGP is a very complex task and it is still unclear how this can be optimally accomplished. The reason for this lies in the lack of global coordination between the policies used in the different domains. This causes that each AS in any given path may apply its own local policies and route its outbound traffic as desired, overriding any routing advertisement and requirement from downstream ASs (an example of this problem was shown in Section 2.2.1).

To cope with the problem of controlling the inbound traffic of an AS several operational palliatives are possible. The most common palliatives based on BGP rely on the utilization of AS-path prepending [20], the BGP communities [104], or tuning the MED attribute [50].

However, all of them have strong limitations. A corollary of what we exposed in sections 2.2.1 and 3.1.2 is that AS-path prepending might not always work. The BGP communities, on the other hand, provide more control than AS-Path prepending. Recently, an interesting solution based on BGP communities called Virtual Peerings was proposed in [102]. The approach

there, is that a pair of ASs could cooperate and set up a unidirectional IP tunnel between their border routers to manage the traffic between them. The problem, however, is that this solution requires to slightly modify BGP, so that it can convey additional information by means of an optional and transitive extended community value. Unfortunately, ISPs remain cautious about modifying BGP. In general terms, the main problems of BGP communities are that they are not perfect and they are not always supported. We have also shown in Section 2.2.1 that the MED attribute only offers a way to "suggest" a neighboring AS about the preferred ingress point to the AS, so it cannot guarantee the expected result.

Due to these limitations in BGP, alternative – non-BGP-based – techniques have arisen. Some of these techniques rely on Network Address Translation (NAT) [49, 86]. However, controlling the traffic by means of NAT is simply unfeasible for medium and large ASs.

After examining the problems and the existing solutions, the overall result is that in practice inbound traffic is manually configured and tuned on a trial-error basis, and hence, remains as an open problem in the area of inter-domain TE.

Another important topic is that the objectives of inter-domain TE drastically vary depending on the type of AS. The classification of the three different types of ASs made in Section 2.1 is pertinent, since the requirements and the problems faced by each of these ASs are quite different. For instance, the current trend for multihomed stub ASs is to deploy selfish TE techniques able to operate in short timescales [5][4]. These techniques typically try to exploit the multi-connectivity of the AS, with the aim of improving the performance and reduce the monetary costs. The main problem behind this TE model, is that if more and more ASs keep on using such selfish techniques, this could place significant stress on the scalability and reliability of the entire inter-domain routing system.

On the other hand, TE mechanisms developed for transit multihomed ASs such as large ISPs are designed to operate in large timescales (typically in the order of weeks or months). These ASs usually use a routing practice known as hot potato routing [2]. In this practice a BGP router within the AS will be able to reach a certain destination by multiple exit points of the AS, so the router needs to run the BGP decision process (see Algorithm 1 in Section 2.2.1). Typically, a subset of those multiple exit points will supply the same AS-path length toward the destination, so the BGP decision process usually reaches steps 4 or 5 in Algorithm 1 in Section 2.2.1. These steps indicate that the AS tries to get rid of the packets as fast as possible (that is why is called

[4]These TE techniques will be thoroughly analyzed in Chapter 4.

hot potato routing). One of the main problems that transit ASs are facing in terms of TE is that usually the attempts to improve their hot potato routing have a profound impact on their inter-domain traffic (and reciprocally) [2]. This causes that traffic patterns change across the boundaries of the transit AS affecting other ASs. These latter may now run their own TE policies, which in turn may negatively impact back again on the original AS. This brings back the problem of routing instabilities due to poor or no coordination between the policies used in the different domains.

In sum, controlling and communicating the inter-domain TE decisions of an AS by means of BGP has demonstrated to be not only inaccurate but also poorly effective in practice. Novel inter-domain TE models, and particularly, highly effective ways of controlling and signaling TE decisions between domains will be necessary in the coming years. A significant part of this thesis is devoted to this end. To conclude with this overview of BGP-based inter-domain TE techniques and their limitations, we recommend the following references for the reader who wants to get deeper into the subject [101, 103, 122].

3.3 BGP-based QoS routing

Services such as VoIP or VPNs have strong requirements in terms of QoS. To fulfill those requirements, many ISPs have deployed mechanisms to provide differentiated treatment to part of the traffic inside their networks. Customers are now requiring similar levels of QoS for network services than span across the domain boundaries [87]. To accomplish this goal, it is necessary to count with a mechanism capable of finding paths between a source and a destination satisfying one or more QoS constraints. This is precisely the function of QoS Routing (QoSR). Unfortunately, BGP has no inbuilt QoSR capabilities, since it was designed as a protocol to distribute essentially reachability information [82]. The inability of BGP to supply and distribute QoS information was recognized as a missing piece by the IETF in mid-1998 [26].

This issue has received a significant amount of attention during the last years. We cannot review here the entire literature, but some appealing proposals can be found in [27, 96, 133], and recently, an extended version of BGP named EQ-BGP was proposed in [83].

Despite the efforts and over a decade of work, the astonishing outcome is none the proposals has turned out to be sufficiently appealing to become deployed in practice. This is because ISPs have preferred to overprovision their networks rather than delivering and managing QoS. The debate about overprovision vs. QoS is still open. Leaving aside issues like the monetary

cost to deploy and maintain QoS, or the development of possible businesses leading to tangible sources of profit for ISPs, an undeniable fact is that the issue remains unsolved mainly because "*all*" the issues presented before in this chapter are actually strong limitations for BGP-based QoS at the inter-domain level. The inter-domain routing paradigm itself is in fact a major cause for this lack of QoS support.

An alternative could be to change the paradigm, but at present only incrementally deployable approaches seem realistic. In sum, efficient mechanisms allowing domains to improve their end-to-end performance while demanding minimal efforts to support and maintain are still missing. This is precisely the subject addressed in the next part of this thesis.

Part II

Route Control: Present

Chapter 4

Intelligent Route Control (IRC)

At present, most of the inter-domain communications are carried out between nodes located in non-transit (a.k.a stub) domains. This particular fraction of domains represents nearly 85% of the more than 25000 ASs that currently compose the Internet [101], and crowds together primarily medium and large enterprise customers, public administrations, Content Service Providers (CSPs), universities, and small Network Service Providers (NSPs).

By having multiple connections to the Internet, these domains can potentially obtain a number of benefits, especially, in terms of resilience and traffic performance [3], so the large majority of stub domains in the Internet are multihomed. It is worth highlighting that these are actually potential benefits, since multihoming per se is unable to guarantee the improvement of any of them. Therefore, multihomed stub domains need of additional mechanisms in order to improve the performance and reliability of their inter-domain communications. In particular, when an online mechanism actively controls how traffic is distributed and routed among the different links connecting a multihomed stub network to the Internet, it is referred to as *Intelligent Route Control* (IRC)[1].

In light of this, IRC has gained significant interest in both the research and commercial fields during the last few years. In what follows we first provide the necessary background and describe the basics of the IRC model. Then, we examine the key deficiencies in the current IRC model, and finally, we review the most relevant related work. The contributions in this thesis to the IRC area will be introduced in the following two chapters.

4.1 The facts supporting the IRC model

An important aspect of inter-domain routing is that recent studies reveal that the topological characteristics of inter-domain paths show large variations over time. Indeed, large fractions of AS-paths are only present in the BGP routing tables for a few minutes [123]. This behavior is increasing the

[1]In the literature IRC is sometimes also referred to as *smart route control*.

number of BGP messages traversing the network. Despite this variability, four important results support the current IRC model at the inter-domain level:

(i) Measurement studies show that, in one AS, a small fraction of the destination prefixes are responsible for a large fraction of the inter-domain traffic [107, 123]. While this applies at the prefix-level, clearly, a correlation exists at the AS path-level, so a similar conclusion can be drawn. For instance, the measurements conducted in [123] reveal that only six AS paths carried about 36% of the one-month total traffic of a multihomed stub AS.

(ii) Regardless of the large number of BGP update messages, "popular" destination prefixes represent stable entries in the BGP forwarding tables for weeks or even months [107, 123].

(iii) The majority of the BGP update events correspond to prefixes that do not receive much traffic [123].

(iv) Actively tweaking BGP so as to improve the performance of the traffic that flows "from" an AS is perfectly feasible, even, in short timescales. This is because the outbound traffic of an AS can be dynamically altered by means of BGP without needing to advertise the changes to the global Internet, i.e., without affecting any BGP router outside the local AS. Conversely, actively tuning BGP in order to improve the performance of the inbound traffic of an AS is unfeasible in rather short timescales. Controlling the flow of the inbound traffic of an AS implies to modify how upstream domains select their best path toward that AS. Unfortunately, this requires to advertise and propagate outside the AS every single tuning made in the AS, which normally affects the routing tables of a large fraction of BGP routers across the Internet. Clearly, it is not recommended to follow this approach too often. In addition, the effectiveness of controlling the inbound traffic of an AS is rather unpredictable, since it depends on the willingness of upstream domains to honor the advertisements of the AS [138].

These facts have important repercussions, since researchers can focus on devising novel TE and route control mechanisms that multihomed networks would apply to a significant fraction of their outbound traffic – whose routes are typically stable.

4.2 The basics of IRC

Several manufacturers are developing and offering IRC solutions targeting multihomed stub domains [10, 22, 58, 86]. All these solutions follow the D^3 rule introduced at the end of Section 1.3.3, i.e., IRC offers a detached, decoupled, and distributed route control model. IRC solutions are detached because they are deployed as independent devices – intelligent route controllers are always detached from the rest of the routers in the multihomed stub AS (see AS1 in Fig. 4.1). IRC offers a decoupled solution in the sense that the routing and TE decisions of the multihomed stub domain are controlled by a process that is not embedded in BGP. In addition, the IRC model is inherently distributed, given that each multihomed stub domain can intelligently distribute and route its traffic independently.

Based on the facts listed in Section 4.1, all available IRC solutions follow the same principle, that is, they actively improve – in short timescales – the end-to-end performance and reliability of the traffic that flows from a multihomed stub domain towards a reduced set of "popular" destination prefixes. To accomplish such improvements, the IRCs[2] are capable of performing a series of tasks, which basically include discovery and monitoring of popular destination prefixes by means of passive and active measurements, and dynamically routing the traffic towards them depending on the outcome of their measurements. In brief, IRC solutions have the ability to influence how traffic is routed among non-directly connected multihomed stub domains by means of measurement-driven dynamic path switching techniques performed at the source domain[3].

IRC proposes an incremental approach, by exploiting the BGP infrastructure while complementing at the same time some of the most important deficiencies of the BGP-based route control model. In IRC, the set of candidate routes to be probed by the route controllers is determined by BGP. In opposition to overlay networks [4], or inter-domain tunnels [102], intelligent route controllers never circumvent BGP. Instead, they select on-the-fly the egress link from the AS for each popular destination prefix based on the result of their measurements. The effectiveness of this approach is confirmed not only by recent studies like [4], but also by the increased trend in the deployment of these solutions.

Since the existing IRC solutions only operate over the outbound traffic

[2]Note that the singular form of the acronym IRC represents Intelligent Route Control, whereas the plural form represents Intelligent Route Controllers.

[3]Clearly, IRC solutions are not applicable to large transit ASs, such as Tier1 and Tier2 ISPs, given that the effects of switching large amounts of inter-domain traffic in short timescales are unpredictable.

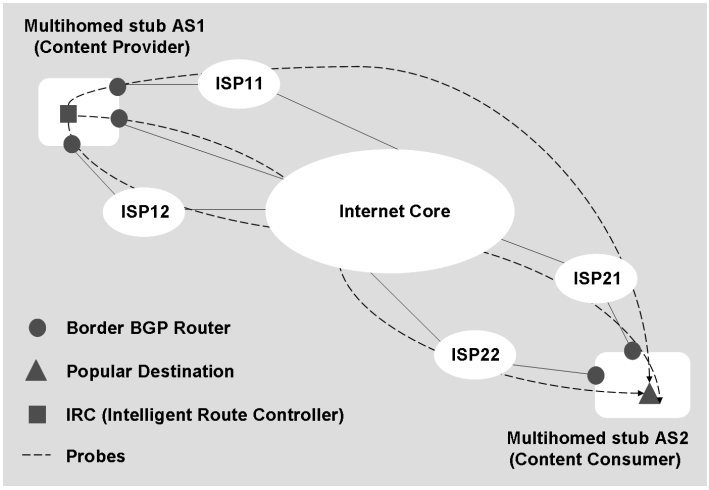


Figure 4.1: The IRC model.

from a multihomed stub domain, the IRC model is applicable to domains that serve traffic to the greater Internet (e.g. a content provider), but not for those that mainly receive traffic from this latter (i.e. domains that are essentially content consumers). Figure 4.1 shows a content provider (AS1) as a multihomed source domain allocating one controller, and a major content consumer (AS2) where one of the popular destinations of AS1 is located. All available solutions [10, 22, 58, 86] operate in the same way. The route controller in AS1 sends probes towards a reduced set of popular destinations through all the egress links of AS1 from which those destinations are reachable (see Fig. 4.1). The probe replies received by the route controller allow this latter to gather measurements of end-to-end parameters, such as RTTs and packet losses, along the candidate paths provided by BGP between AS1 and its popular destinations. Based on these measurements, the route controller in AS1 is capable of taking rapid routing decisions to bypass network problems, such as distant link/node failures[4] or performance degradation[5], for a particular set of popular or even pre-configured destination prefixes.

A central issue, however, is that multi-connectivity to the Internet together with an IRC tool does not guarantee an increase in the end-to-end path diversity. In fact, part – or even all – of the available paths between a source domain and one of it's popular destinations might have some overlapping segments. This can be clearly observed in Fig. 4.1. The source AS1 has three egress links, whereas the destination AS2 has only two ingress links, so

[4]The timescale needed by IRC tools to detect and react to a distant link/node failure is very small compared with that of BGP [68, 10, 22, 58].

[5]Which is not possible at present with BGP – at least in operative networks.

at least two of the three paths probed by the route controller in AS1 overlap over one of the ingress links of AS2. Overall, this is due, first, to the fact that BGP only advertises the best path it knows, so BGP considerably hides the total number of available paths between distant ASs, and second, to the topological characteristics of the Internet at the AS-level. Despite several studies have addressed these scarce path diversity issues [3, 70, 139], recent studies like [4] demonstrate that, in practice, multihoming in combination with IRC are powerful techniques to improve the end-to-end performance of inter-domain communications – though clearly they cannot guarantee to speed up the recovery from a distant link/node failure due to the potential overlap of the paths.

The strengths of IRC can be summarized as follows. First, the IRC model can be effectively adopted by multihomed stub domains without needing any kind of cooperation or interactions with transit domains, since IRC operates transparently to these latter. Second, IRC offers a straightforward and cost-effective way of improving the end-to-end performance and reliability of the outbound traffic by exploiting the multi-connectivity of stub domains. And third, IRC does not introduce any kind of modification or extension to BGP, but rather, it simply exploits the existing BGP infrastructure.

Despite these strengths, all IRC solutions available at present have in common two major weaknesses, which have motivated the contributions in this part of the thesis. We examine these issues in the next section.

4.3 Deficiencies in the current IRC model

The first issue is that all IRC solutions available at present are standalone, so no cooperation exists between the ASs sourcing and sinking the traffic – clearly, no cooperation exists with the ASs providing transit to the traffic. The main consequences of this lack of cooperation are: i) coarse route control over the outbound traffic of the ASs; and ii) the inability to intelligently control the inbound traffic.

The second issue is that all available solutions behave in a fully selfish way, that is, they operate without considering the effects of their decisions in the performance of the network. Therefore, it becomes unclear if these route controllers could still perform so well if several of them compete for the same network resources.

Conversely to a previous work which argues that the interference between multiple competing standalone route controllers causes only minor performance penalties [44], it will be shown in Chapters 5 and 6 that in practice the penalties can be large, especially, when the network utilization increases.

In that work, the performance penalty considered was the average latency, and it was evaluated at traffic equilibrium. Unfortunately, the available route control solutions at the AS-level are not precisely focused on seeking such kind of theoretical equilibrium. In addition, other performance penalties must be considered in practice, such as the implications associated with the number of traffic relocations needed to obtain a certain latency. For the route control solutions operating at the AS-level there are two major implications. First, each traffic relocation causes the flood of iBGP messages, so that all the BGP routers inside the AS learn about the new egress point from the AS to reach the popular destination. Second, it was recently found that bounces of traffic relocations and even oscillations might occur [42].

4.4 Related work

Several research efforts are being carried out in order to improve the performance and reliability of inter-domain communications. As shown in Fig. 4.2, these efforts have contributed with solutions that can be divided into two different groups. The first group gathers solutions trying to enhance BGP with new capabilities, such as TE and QoS extensions. Some of the most relevant proposals in this area were cited in Sections 3.2 and 3.3. All these in-band solutions, i.e., solutions intrinsically supported and signaled using BGP, are able to supply improvements over rather large timescales. Unfortunately, they are inadequate to manage and distribute inter-domain traffic in short timescales. The reasons for this are that BGP is a slow reacting protocol [68], and also that tweaking BGP too often would significantly increase the number of messages exchanged between BGP routers, which may lead to network instabilities [77, 127]. Overall, in-band solutions are not able to cope with the current and the expected demands of multihomed stub ASs.

The second group shown in Fig. 4.2 is composed by solutions tending to decouple the route control and TE provisioning tasks from BGP devices. These out-of-band solutions, i.e., solutions supported and signaled without using BGP, are able to operate in shorter timescales, even reaching timescales in the order of a few seconds. These out-of-band solutions can be in turn divided into two different groups (see Fig. 4.2).

In the first group we found overlay networks. The main idea behind the overlay concept is to entirely decouple the routing process from BGP devices. Overlay networks circumvent BGP, so that the applications can get the desired end-to-end performance from the network. As a result of the strengths of this approach, the overlay scheme has gained its position becoming a solid research area. Among the most relevant proposals are the

following.

In [1], the authors introduce an overlay model allowing an AS to request for a route change to another AS in the overlay architecture. In this proposal, a server named Relationship Mapper (RMAP) acts as a repository of the inter-AS relationships and Internet hierarchy. These relationships are deduced from multiple BGP routing tables by applying heuristics. Then, each time an AS needs to reroute part of its incoming traffic it consults the RMAP server to know exactly which AS in the overlay network needs to be contacted, so that this latter performs the appropriate route advertisements.

In [117] the authors present an approach for providing Internet QoS using overlay networks. The proposal consists of using a Controlled-Loss Virtual Link (CLVL) abstraction that bounds the loss rate experienced by the overlay traffic. This abstraction is used basically to provide statistical bandwidth and loss assurances. In simple terms, the approach consists of trading throughput for loss-rate, wherein the problem is reduced to find a minimum redundancy factor such that the desired loss-rate can be achieved.

The Detour framework is also an interesting proposal based on providing an overlay solution to avoid some of the main issues in inter-domain routing [23, 110]. Detour was based on a virtual network allowing users' traffic to be routed around failures and heavily congested paths.

In [6] the authors propose RON (Resilient Overlay Networks), as a way to improve the robustness and availability of paths between hosts separated across a wide-area routing infrastructure. RON provides a framework in which a small group of distributed Internet applications could detect and recover from path outages or service level degradation within tens of seconds, highly improving the timescale needed by BGP, which might need even minutes to recover. RON hosts measure QoS parameters among themselves, and

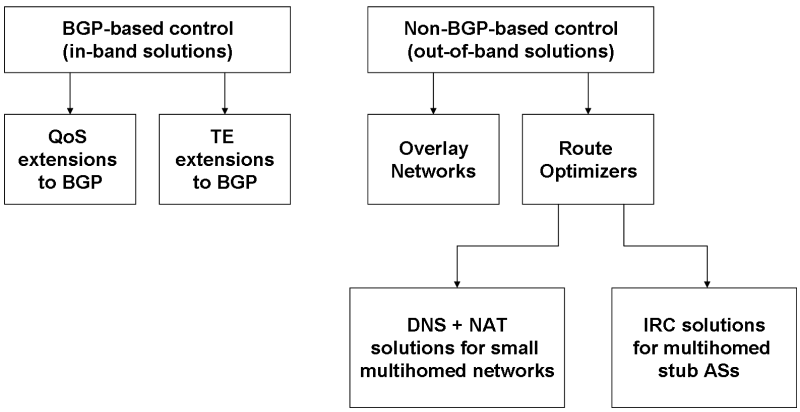


Figure 4.2: Current strategies for inter-domain route control.

use this information to decide whether to route packets directly over the Internet, or indirectly by way of other RON nodes, optimizing application-specific routing metrics.

Recently, an extension called QRON (QoS-aware Routing in Overlay Networks) was proposed in [73]. This proposal presents the concept of Overlay Brokers (OBs), in which every AS within the Internet should have one or more OBs. These OBs collaborate with each other composing an Overlay Service Network (OSN), providing overlay services such as resource allocation, routing, and topology discovery.

An alternative approach is presented in [29], in which a Service Overlay Network (SON) purchases bandwidth from the underlying network to provide end-to-end value-added QoS sensitive services, such as VoIP or Video-on-Demand (VoD).

In [136] we proposed an incremental approach, where an overlay architecture together with a new routing layer are used for dynamic QoS provisioning, and an enhanced version of BGP (QBGP) is used for static QoS provisioning. The focus on that work was mainly to influence how traffic was routed among non-directly connected multihomed stub domains based on specific QoS parameters.

Even though all the aforementioned overlay proposals enhance in one way or another the end-to-end performance and reliability of inter-domain traffic, none of them has been adopted at the AS-level. While some of these proposals are centralized and rely on rather complex – and sometimes inaccurate [1] – heuristics, others are distributed but definitively not scalable, or need massive deployment in order to be able to operate (i.e. at least one overlay node per-AS in every AS). Another issue is that cooperation is needed between the ASs participating in the overlay. For instance, in some proposals domains should be willing to support the transit though them, by forwarding traffic towards other ASs in the overlay. In practice, overlay networks are mainly utilized by end-users for file sharing, distributing content, and private networks purposes. The overlay model does not seem applicable to entire ASs yet – or at least is has not gain popularity in this area.

In the second group of the out-of-band solutions (see Fig. 4.2), we found route optimizing tools. Two types of route optimizers are commercially available at present, namely, DNS-based optimizers [5, 86], and IRCs [10, 22, 58, 86].

The DNS-based solutions rely on NAT (Network Address Translation) to manage how traffic flows from/into the AS, which can add its own problems for some applications. Additionally, these solutions are based on intensive end-to-end active probing, and the advertisement of DNS responses with a very low Time To Live (TTL). This latter forces the end user's DNS server

to request an updated IP address every few seconds. Thus, their endeavors may be useless if for instance a DNS server does not honor a very low TTL, or if the end user's application caches DNS requests. DNS-based solutions are not scalable, as they are addressed to small organizations that are not willing to deal with the difficulties of BGP peering and management. Thus, they are out of the scope of this thesis.

IRC on the other hand, provides a much more scalable solution since it is addressed to larger organizations. Among the current top line products are [10, 22, 58]. Despite most commercially available IRC solutions do not reveal in depth the technical details of their internal operation and route control decisions, the behavior of one particular controller is described in detail in [49]. That work also contributes with measurements evaluating the effectiveness of different design decisions and load balancing algorithms. Akella et al have also provided rather detailed descriptions and experimental evaluations of multihoming in combination with IRC tools, like [3], [4], and [5]. These research publications, along with the documentation provided by vendors, allowed us capturing and modelling the key features of conventional IRC techniques[6]. A similar approach was followed by the authors in [42].

In [44], the authors simultaneously optimize the cost and performance for multihomed stub networks by introducing a series of new IRC algorithms. However, the contributions in that work are mostly theoretical. For instance, the authors claim that an intelligent route controller can improve its own performance without adversely affecting other IRCs in a competitive environment, but the conclusions are drawn at traffic equilibria[7]. As mentioned in Section 4.3, after examining and modelling the key features of conventional IRC, it becomes clear that the available IRC tools neither seek nor try to operate subject to such kind of traffic equilibria.

Indeed, as a recent work shows [42], conventional IRC solutions can actually cause significant performance degradation instead of improvements. The key contribution in [42] is to show that in a competitive environment, persistent oscillations can occur when independent controllers get synchronized due to a considerable overlap in their measurement time windows. To avoid the synchronization issues, the authors propose simple randomized IRC algorithms, and empirically show that the oscillations disappear after applying their randomized route control strategies.

In the next two chapters, we will show that it is possible to devise novel intelligent route control strategies outperforming conventional IRC. More

[6]By conventional we mean state-of-art or existing IRC techniques.

[7]A traffic equilibrium is defined by the authors in [44], as a state in which no traffic can improve its latency by unilaterally changing its link assignment.

precisely, the IRC algorithms proposed here are able to reduce the number of path shifts approximately between 40% and 80% on average in a competitive environment, while even getting better end-to-end performance than conventional IRC. It is worth highlighting that our algorithms also avoid the synchronization issues found in [42], by means of a randomized technique that we developed in a previous work [136].

Chapter 5

Social Route Control (SRC)

Multihoming in combination with IRC solutions are becoming a common practice in order to improve the end-to-end performance of the communications sourced at stub domains. As described in Chapter 4, IRC allows to actively exploit the multi-connectivity of stub domains to the Internet, by leveraging the relocation of part of their outbound traffic in short timescales. The existing IRC practices were not considered adverse, but recent studies like [42] show that IRC can actually cause significant performance degradation rather than improvements in a competitive environment.

In light of this, it becomes necessary to explore alternative route control strategies under the current trend, in which completely independent and uncoordinated multihomed stub domains can simultaneously tweak their inter-domain traffic distributions seeking only for the best of their own purposes in short timescales. These new route control strategies should:

- Always improve the performance and reliability of inter-domain communications, hence avoiding adverse effects.

- Drastically reduce the penalties associated with frequent traffic relocations, such as packet losses [42], floods of iBGP messages [137], and persistent oscillations [42].

Accordingly, in this chapter we propose, design, and test a Social Route Control (SRC) model for multihomed stub domains in a competitive environment. In the SRC model, each controller remains independent, so it does not need any kind of coordination or interactions with the rest of the competing controllers in the network – our SRCs operate in a standalone fashion, just as conventional IRCs do. The key is that each controller is endowed with a social route control algorithm that adaptively restrains its intrinsic selfishness by learning from and evolving together with the network dynamics.

More specifically, the route control decisions made by the SRCs not only depend on the current state of the network, but also on the dynamic of these states, since the route control decisions are able to evolve and adapt jointly

with the network dynamics. Under changing network conditions, it is imperative that each route controller counts with a social mechanism allowing it to adapt by diminishing or even preventing the path shifts until the network conditions become once again stable. Such network conditions might occur when a significant number of conventional IRCs compete for the same network resources, during link flaps, or even routing misconfigurations.

This thesis makes the following contributions:

(i) This is the first study providing a thorough and highly detailed step-by-step design of an IRC strategy endowed with social behavior.

(ii) In [42] the authors propose randomized techniques to avoid IRC oscillations. In this study we show that randomization only offers a way to de-synchronize competing route controllers, but it still leads to a large number of unnecessary path shifts. These latter can be easily avoided by "socializing" the decisions of the route controllers. By blending randomization techniques with a social route control algorithm it is possible to outperform the current IRC model. With this novel approach it is possible to avoid oscillations, to obtain significant improvements in terms of the end-to-end performance, while drastically reducing the path shifts needed to achieve the desired performance.

(iii) As far as our knowledge, we have carried out the largest tests made so far to assess the performance of different IRC strategies in a competitive environment.

(iv) The social extension proposed in this chapter can be easily integrated and used today, since the only thing actually needed is a software upgrade of the available route controllers.

5.1 The network model

A typical IRC scenario is shown in Fig. 5.1. The multi-domain network is composed by the source domain S, the transit domains, and a set of popular destination prefixes $\{p\}$, with cardinality $\|p\| = P$. The source domain S has a set of egress links $\{e\}$, with $\|e\| = E$.

In order to dynamically decide the best egress link to reach a popular destination p, the social controller probes all the candidate paths through the egress links e of S using the same techniques and the same measurement platform that conventional IRCs use today. It is important to highlight that the SRC model does not introduce any change in the way that measurements

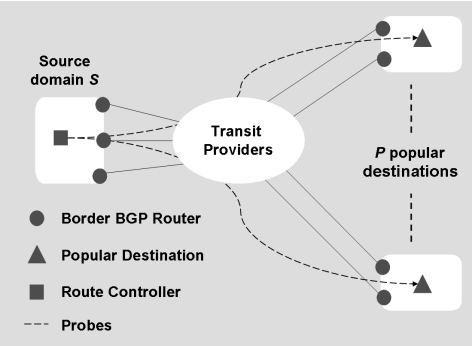


Figure 5.1: The network model.

are conducted and computed today by conventional IRC solutions. The changes proposed here apply to the route control algorithm, i.e., after the measurements have been collected and processed.

The collected measurements are utilized to compute a cost $C_e^{(p,t)}$ at time t, associated with each of the available paths towards a popular destination p of S. Let $\widehat{N}^{(p)}$ denote the number of available paths to reach p. Since $\widehat{N}^{(p)}$ represents the number of candidate paths in the FIBs of the BGP border routers of S, $\widehat{N}^{(p)} \leq E \; \forall \; p$.

We now proceed to describe the details of the SRC strategy, and particularly, how we propose to compute and use the $min\{C_e^{(p,t)}\}$ towards each p in the SRC algorithm.

5.2 The SRC strategy

Two candidate approaches can be adopted for the social route control algorithm running on the SRCs. On the one hand, the algorithm could follow a reactive approach, i.e., switch traffic only when a pre-established bound is not fulfilled. An alternative is to follow a proactive approach by switching traffic as soon as the performance becomes degraded up to some extent. After extensive evaluations, we have confirmed that controlled proactive approaches perform much better than reactive ones. This claim applies not only in terms of end-to-end performance, but also in terms of the penalties associated with frequent traffic relocations. The reason for this is that proactive approaches are able to anticipate network congestion situations, which in the reactive case, typically demand several traffic relocations when congestion has already been reached. Accordingly, the social route control algorithm proposed here is based on a proactive approach.

5.2.1 An adaptive integer cost

Equation (5.1) presents the integer cost used by each social controller. The collected measurements allow to compute the two terms of the additive cost $C_e^{(p,t)}$, $\forall\ p, e$. The first term consists of end-to-end delay information[1], which is based on processing and filtering the RTTs inferred from the probes. We name this first component *Smoothed RTT* (SRTT)[2], and it is denoted as $S_e^{(p,t)}$. The second term of the cost consists of local information that is based on collecting the *Available Bandwidth* (AB) from the egress links e of S, and it is denoted as $AB_e^{(t)}$.

In addition, the non-negative parameters $\alpha^{(p,t)}$ and $\beta^{(p,t)}$ are the adaptive weights that endow the SRCs with the desired social behavior – their role and the way we propose to let these weights evolve in time will become clear in the rest of this section.

The bound $\overline{D^{(p)}}$ represents the maximum tolerable RTT to reach a popular destination prefix p, which can be a pre-configured value depending on the kind of traffic sent to that prefix. On the other hand, the bound B_e represents the minimum acceptable bandwidth in the e_{th} egress link of S. This constraint supplies a minimum bandwidth guarantee per-egress link of S, and it can be configured by the network administrator in S according to the domain's resilience and performance policies. It basically offers a safe margin, so that in case of an inter-domain link failure, the affected traffic can be distributed along the available egress links of S.

$$C_e^{(p,t)} = \begin{cases} \left\lfloor \alpha^{(p,t)} S_e^{(p,t)} + \dfrac{\beta^{(p,t)}}{AB_e^{(t)}} \right\rfloor & \text{if } S_e^{(p,t)} \leq \overline{D^{(p)}} \ \wedge \ AB_e^{(t)} \geq B_e \\ \infty & \text{if } S_e^{(p,t)} > \overline{D^{(p)}} \ \vee \ AB_e^{(t)} < B_e \end{cases} \tag{5.1}$$

If any constraint is violated the cost $C_e^{(p,t)}$ is set to infinite. This means that the e_{th} egress link of S should be removed from the list of available links to reach p as long as a violation exists. The main motivations for selecting this particular additive cost can be summarized as follows:

[1]The SRC strategy proposed here is especially designed to improve the end-to-end performance of delay-sensitive applications, such as VoIP.

[2]Our aim is to avoid unnecessarily changing the cost too often, so instead of using instantaneous values of the collected RTTs, we use filtered RTTs for the computation of $C_e^{(p,t)}$.

(i) $C_e^{(p,t)}$ is simple, and it is easy to compute.

(ii) As we will show $C_e^{(p,t)}$ effectively captures the dynamics of the delay along the candidate paths.

(iii) The weights $\alpha^{(p,t)}$ and $\beta^{(p,t)}$ in (5.1) facilitate to prioritize the relevance of the RTTs over the local AB. The additive term of AB in $C_e^{(p,t)}$ indicates that the social controller will prefer to route traffic over egress links with more AB when the RTT conditions are similar along two or more candidate paths. This allows an overall better traffic distribution for the outbound traffic from S.

5.2.2 A two-stage filtering process of the RTTs

In order to compute the cost in (5.1) the SRCs smooth the RTT samples gathered from the probes using two filters in cascade. The first filter corresponds to the median RTT, since it is widely accepted as an excellent estimator of the delay that the users' applications are currently experiencing in the network. As shown in (5.2), the median is computed through a sliding window of size W probes. The index $n_e^{(p,t)}$ in (5.2) simply represents the sequence number of the instantaneous samples of RTTs to detect the occurrence of losses in the probes.

$$\mathcal{M}_e^{(p,t)} = Median(RTT_e^{(p,k)}), \quad k \in \left[n_e^{(p,t')} - W + 1, n_e^{(p,t)}\right] \tag{5.2}$$

The computation of the mean or the median for a set of measurements through a sliding window is a usual practice. In our case, we have tuned W in order to get a good trade-off between the responsiveness of the filter, and a strong correlation between the measurements and the applications under control. The median has two important advantages compared to the mean. First, the mean is much more biased by outliers than the median. Second, computing the mean RTT through a sliding window needs a special treatment in case of losses, whereas lost RTT samples can be set to infinite without problem while computing the median. In sum, the first filter is utilized by the route controllers as an estimate of the delay experienced between the source S and its popular destinations p.

The second filter is what actually endows the route controllers with the social behavior. Before getting deeper into the details of the design of this filter and the SRC algorithm, we provide a high-level description so as to help understanding the general aspects of the proposed SRC strategy.

High-level description of the SRC strategy

Our goal is that the social controller in S becomes capable of adaptively adjusting its proactivity depending on the RTT conditions. To be precise, the social controller analyzes the evolution of the RTT, and depending on it's dynamics, the controller can adaptively restrain its traffic reassignments (i.e., its selfishness).

To accomplish this, the social controller performs the following tasks. First, it classifies and groups the probe replies received from the P destinations according to which particular flow of probes they belong to (recall that conventional IRC uses one flow of probes for each available egress link at S, $\forall\ p$). From these groups of replies, the social controller obtains the evolution of the median RTT, $\mathcal{M}_e^{(p,t)}$, for each available path at S, $\forall\ p$.

The evolutions of these medians are precisely the inputs to the second filter, where the social nature of the algorithm covers two different facets: i) controlled proactivity; and ii) socialized route control.

Controlled Proactivity: On the one hand, the proactivity of S is controlled so as to avoid that minor changes in the medians trigger traffic relocations at S. The advantages of this approach are twofold. First, it reduces the performance penalties associated with each traffic reassignment. And second, it avoids interfering too often with competing route controllers. For this reason, the social controllers filter the evolution of the medians.

This second filter works like an A/D converter, and its outcome is the the SRTT, i.e., the term $S_e^{(p,t)}$ in (5.1). An example of the two-stage filtering process for one of the available paths at S is depicted in Fig. 5.2. The instantaneous samples of RTT are filtered to obtain the median RTT, and the evolution of this latter is filtered to obtain the SRTT.

The social route control algorithm only takes into account and compares SRTTs. Thus, domain S may relocate certain traffic towards p only when a variation in the SRTT along one of the available paths, produces a change in the best past selection at S (see Fig. 5.2). The number of paces that the SRTT needs to change so as to trigger a traffic reassignment can be configured by the administrator of S. We foresee hence different and configurable proactive strategies for the SRCs. The first advantage of this filter is that it produces the desired effect, that is, it prevents that minor changes in the medians trigger unnecessary traffic relocations at S.

Socialized Route Control: The second facet of the social behavior of the algorithm has to do with the dynamics of the median RTTs, to be precise, with how rapid are the variations in the evolution of the median values. The motivation for this is that when the median values start to show rather quick variations, the algorithm must react so as to avoid a large number of traffic reassignments in a short timescale. Such RTT dynamics typically occur when several smart route controllers compete for the same resources, leading to situations where their traffic reassignments interfere between each other.

To cope with this problem, our heuristic is to turn the second filter in Fig. 5.2 into an adaptive filter. This filter is endowed with an adaptive pace of conversion, which is automatically adjusted by the algorithm according to the evolution of the median RTTs. If the RTT conditions are smooth the pace is small, and more proactivity is allowed at S. However, if the RTT conditions may lead to instability the pace increases and the number of changes in the SRTT is diminished or even stopped until the network conditions become smooth once again. This has the effect of de-synchronizing only the competing route controllers. Therefore, the second advantage of the filter is that it can be exploited by SRCs to "socially" decide whether to reassign the traffic to an alternative egress link or not, and the degree of "sociability" of S is constantly adjusted by the adaptive nature of the second filter.

In the rest of this section we describe in detail the design of the second filter and its relation with the weights $\alpha^{(p,t)}$ and $\beta^{(p,t)}$ in (5.1). Figures 5.2 and 5.3 will help to understand how we propose to actively adapt this two-stage filtering process.

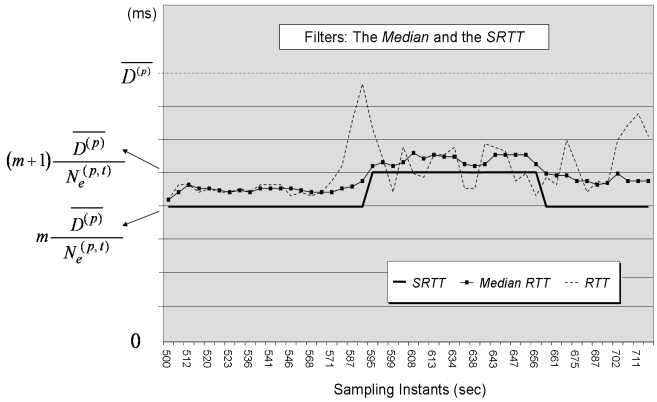


Figure 5.2: The two-stage filtering process.

Designing the adaptive filter

Our goal is to design this second filter with adaptive capabilities, particularly, depending on varying RTT conditions on the network.

First Design Decision: To this end, and as shown in Fig. 5.2, the interval $[0, \overline{D^{(p)}}]$ is initially divided in $N_e^{(p,0)}$ subintervals or paces, i.e.:

$$\left[m\frac{\overline{D^{(p)}}}{N_e^{(p,0)}}, (m+1)\frac{\overline{D^{(p)}}}{N_e^{(p,0)}}\right], \qquad 0 \leq m \leq (N_e^{(p,0)} - 1) \tag{5.3}$$

defining an initial set of grids $\forall\ e, p$. In order to design the filter we define the following parameters using the first W instantaneous samples of the RTTs:

$$\begin{cases} \overline{RTT_e^{(p,0)}} = \max\limits_k \{RTT_e^{(p,k)}\} & \forall k = 1, \ldots, W \\ \\ \underline{RTT_e^{(p,0)}} = \min\limits_k \{RTT_e^{(p,k)}\} & \forall k = 1, \ldots, W \end{cases} \tag{5.4}$$

Then, the interval $[\underline{RTT_e^{(p,0)}}, \overline{RTT_e^{(p,0)}}]$ defines our first estimation of the range of variation of the instantaneous samples of RTT (see Fig. 5.3). Our aim is to prevent unnecessary variations of the SRTT $S_e^{(p,t)}$ in the cost $C_e^{(p,t)}$, so the main idea behind the adaptive filter is that moderate variations of the median $\mathcal{M}_e^{(p,t)}$ generate the same numerical value of $S_e^{(p,t)}$, and thus the same cost $C_e^{(p,t)}$.

Second Design Decision: Our second decision while designing the filter is that the maximum variation after collecting the first W samples, i.e., $(\overline{RTT_e^{(p,0)}} - \underline{RTT_e^{(p,0)}})$, fits into one subinterval of the grid (see Fig. 5.3). Moreover, we introduce an adjustable coefficient $\Delta^{(p)} \in \Re\ /\ \Delta^{(p)} \geq 1\ \forall\ p$, which assures at least a percentage of separation between the grid lines and the parameters defined in (5.4) given by $(\Delta^{(p)} - 1)\text{x}10^2$. The coefficient $\Delta^{(p)}$ basically reflects the degree of conservativeness while defining the initial grid. In addition, $\Delta^{(p)}$ will also play a fundamental role when adding adaptive capabilities to the filter. Figure 5.3 shows the design approach.

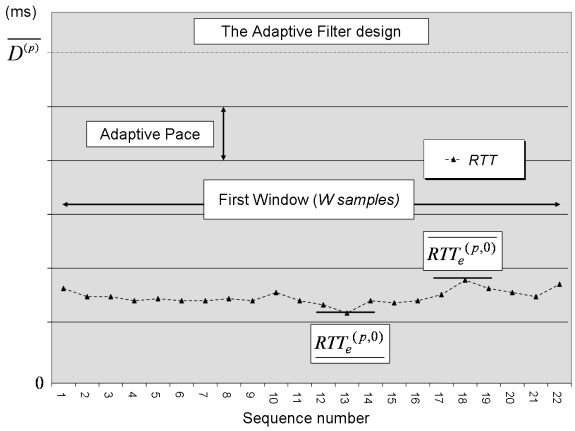


Figure 5.3: The adaptive filter design.

Accordingly, $N_e^{(p,0)}$ is bounded by:

$$\left.\begin{aligned} m\frac{\overline{D^{(p)}}}{N_e^{(p,0)}} &\leq \left(\Delta^{(p)}\right)^{-1}\underline{RTT_e^{(p,0)}} \\ (m+1)\frac{\overline{D^{(p)}}}{N_e^{(p,0)}} &\geq \Delta^{(p)}\overline{RTT_e^{(p,0)}} \end{aligned}\right\} \Rightarrow$$

$$N_e^{(p,0)} \leq \frac{\overline{D^{(p)}}}{\left(\Delta^{(p)}\overline{RTT_e^{(p,0)}} - \left(\Delta^{(p)}\right)^{-1}\underline{RTT_e^{(p,0)}}\right)} \tag{5.5}$$

In order that the route controller can compare the costs $C_e^{(p,t)}$ and $C_{e'}^{(p,t)}$, with $e \neq e'$, it is necessary to use the same pace of conversion, hence: $N_e^{(p,0)} = N_{e'}^{(p,0)}\ \forall\ e \neq e' \Rightarrow N^{(p,0)} = N_e^{(p,0)}\ \forall\ e$. We anticipate that a trade-off exists between the granularity of the pace in the grid, and how proactively the traffic will be switched by the social controller.

Third Design Decision: Following a conservative approach, our third decision is to use the minimum number of paces while generating the initial grid of the filter. Thus:

$$N^{(p,0)} = \left\lfloor \frac{\overline{D^{(p)}}}{min\left\{\overline{D^{(p)}}, \max_e\left(\Delta^{(p)}\overline{RTT_e^{(p,0)}} - \left(\Delta^{(p)}\right)^{-1}\underline{RTT_e^{(p,0)}}\right)\right\}} \right\rfloor \tag{5.6}$$

where (5.6) satisfies the restriction in (5.5) and provides the initial grid $\mathcal{G}^{(p,0)}$ $\forall$ egress link e through which the popular destination p is reachable. Then, the initial pace of the grid is: $G^{(p,0)} = \frac{\overline{D^{(p)}}}{N^{(p,0)}}$.

Once the pace of the grid is determined, it can be easily shown that $\forall$ $t > 0$, the SRTT in (5.1) can be expressed as:

$$S_e^{(p,t)} = \begin{cases} \mathcal{M}_e^{(p,t)} & \text{if } N^{(p,t)} = 1 \\ G^{(p,t)} \left\lfloor \dfrac{\mathcal{M}_e^{(p,t)}}{G^{(p,t)}} \right\rfloor & \text{if } N^{(p,t)} > 1 \end{cases} \tag{5.7}$$

Equation (5.7) is the most general expression for $S_e^{(p,t)}$, and reflects the adaptability of the social approach, that is, the route control decisions can automatically evolve in time, and actually toggle between two different modalities depending on the variations of the RTT, namely: (i) reactively, in the extreme case that $N^{(p,t)} = 1$; or (ii) proactively, whenever $N^{(p,t)} > 1$.

In the reactive case the grid has only one pace ($N^{(p,t)} = 1$), so the SRC algorithm will not allow to switch traffic unless a violation to the bound $\overline{D^{(p)}}$ occurs (the details of the SRC algorithm are described in Section 5.2.4). Conversely, in the usual proactive case, the grid has at least two paces, and the number of paces is dynamically adapted by the SRCs depending on the RTT conditions. In this case, the SRC algorithm allows to proactively switch traffic before the bound $\overline{D^{(p)}}$ is reached.

So far we have described in detail the design criteria for the computation of the initial grid $\mathcal{G}^{(p,0)}$ of the adaptive filter. In the sequel, we will provide the details on how we propose to dynamically adapt the grid $\mathcal{G}^{(p,t)}$ depending on the RTT conditions.

Fourth Design Decision: Our approach is to avoid frequent recalculations of the grid, especially, during potentially unstable RTT conditions, so we propose that each time a new grid is computed, this is maintained for a superset of several windows, $\mathcal{W}$, with $\mathcal{W} = nW$, $n \in \mathbb{N}$. Then, we trigger the recalculation of the grid whenever:

$$\begin{cases} G^{(p,t)} < \max\limits_e \left(\overline{RTT_e^{(p,t)}} - \underline{RTT_e^{(p,t)}} \right) & \vee \\ \min\limits_e \left(\Delta^{(p)} \overline{RTT_e^{(p,t)}} - \left(\Delta^{(p)}\right)^{-1} \underline{RTT_e^{(p,t)}} \right) < \overline{RTT^{(p,t)}} - \underline{RTT^{(p,t)}} \end{cases} \tag{5.8}$$

where now the samples to be considered are those inside $\mathcal{W}$:

$$\begin{cases} \overline{RTT_e^{(p,t)}} = \max\limits_k \{RTT_e^{(p,k)}\} & \forall k \in \mathcal{W}^{(t)} \\ \underline{RTT_e^{(p,t)}} = \min\limits_k \{RTT_e^{(p,k)}\} & \forall k \in \mathcal{W}^{(t)} \end{cases} \tag{5.9}$$

and $\left[\underline{RTT^{(p,t)}}, \overline{RTT^{(p,t)}}\right] = \max\limits_e \left[\underline{RTT_e^{(p,t-1)}}, \overline{RTT_e^{(p,t-1)}} \right]$ in the previous grid $\mathcal{G}^{(p,t-1)}$.

The inequality at the top of (5.8) reflects that the RTT conditions have become less steady, so the pace $G^{(p,t)}$ of the grid needs to be increased, while the inequality at the bottom of (5.8) indicates that the conditions have become even steadier, so the pace could be diminished. It is possible that while a candidate path towards p through an egress link e satisfies the inequality at the top of (5.8), another egress link e' satisfies the one at the bottom of (5.8). In such a case, our design decision is to follow a conservative approach and always increase the pace of the grid.

Therefore, a new grid $\mathcal{G}^{(p,t+1)}$ will be obtained whenever one of the inequalities in (5.8) is fulfilled, and the substitution of:

$$\max_e \left(\Delta^{(p)} \overline{RTT_e^{(p,t)}} - \left(\Delta^{(p)}\right)^{-1} \underline{RTT_e^{(p,t)}} \right) \tag{5.10}$$

in (5.6) for $t > 0$, yields a number of paces $N^{(p,t+1)}$ such that $N^{(p,t+1)} \neq N^{(p,t)}$.

Finally, if the grid was not recomputed after $\mathcal{W}^{(t)}$, instead of setting the current grid for a whole new window $\mathcal{W}^{(t+1)}$, the SRCs began to search for either of the conditions in (5.8) through a sliding window of size $\mathcal{W}$. This approach improves the responsiveness of the social controllers.

5.2.3 Linking the adaptive filter and the additive cost

The next step in the design is to set the weights $\alpha^{(p,t)}$ and $\beta^{(p,t)}$ in (5.1), and link them to the adaptive filter. As mentioned in Section 5.2.1, our goal is to prioritize the role of the RTTs over the local AB in (5.1), and use this latter to tiebreak so as to get a better outbound traffic distribution from S when two or more candidate paths offer similar RTTs. Such prioritization can be easily achieved by properly adjusting the sensitivity of $C_e^{(p,t)}$ with respect to $S_e^{(p,t)}$ and $AB_e^{(t)}$. To this end, we use the following criterion.

Fifth Design Decision: Let $\widetilde{C}_e^{(p,t)}$ denote $C_e^{(p,t)} < \infty$, before the floor operation in (5.1). Then, we choose:

$$\left|\frac{\partial \widetilde{C}_e^{(p,t)}}{\partial S_e^{(p,t)}}\right|_{AB_e^{(t)}} = \max_e \left|\frac{\partial \widetilde{C}_e^{(p,t)}}{\partial AB_e^{(t)}}\right|_{S_e^{(p,t)}} \quad \forall\, t \;\Rightarrow\; \alpha^{(p,t)} = \frac{\beta^{(p,t)}}{\min_e \left(B_e{}^2\right)} \tag{5.11}$$

Sixth Design Decision: The key of the social algorithm is that it is capable to evolve in time and adapt to varying conditions in the observed RTTs. This study proposes that the cost $C_e^{(p,t)}$ absorbs the fluctuations that the RTTs might show during certain intervals, when several independent and selfish IRCs compete for the same network resources. The major advantage of this approach is that when such fluctuations occur, the cost $C_e^{(p,t)}$ will mascarade them, and hide them from the route control decision algorithm. A straightforward way to achieve this is to let the weights $\alpha^{(p,t)}$ and $\beta^{(p,t)}$ evolve together with the adaptive filter and its grid $\mathcal{G}^{(p,t)}$ in the following way:

$$\widetilde{C}_e^{(p,t)}\left(S_e^{(p,t)} + \Delta S_e^{(p,t)}, AB_e^{(t)}\right) - \widetilde{C}_e^{(p,t)}\left(S_e^{(p,t)}, AB_e^{(t)}\right) = \alpha^{(p,t)} \Delta S_e^{(p,t)} \tag{5.12}$$

and since the variations $\Delta S_e^{(p,t)}$ are discretized by the paces $G^{(p,t)}$ of the grid (see (5.7) for $N^{(p,t)} > 1$) $\Rightarrow \Delta S_e^{(p,t)} \propto G^{(p,t)}$, so substituting in (5.12) yields:

$$\Delta \widetilde{C}_e^{(p,t)} \propto \alpha^{(p,t)} G^{(p,t)} \tag{5.13}$$

Now, by choosing:

$$\alpha^{(p,t)} = \frac{1}{G^{(p,t)}} \;\Rightarrow\; \Delta \widetilde{C}_e^{(p,t)} = K \text{ paces, with } K \in \mathbb{N} \tag{5.14}$$

Therefore:

$$\beta^{(p,t)} = \frac{\min_e \left(B_e{}^2\right)}{G^{(p,t)}} \tag{5.15}$$

Table 5.1 summarizes the key advantages of this last design decision, especially, on how the cost $C_e^{(p,t)}$ is able to capture and adapt in the event of fluctuations on the values of the RTTs.

When fluctuations in RTT	The pace $G^{(p,t)}$ of the grid	$\Delta\widetilde{C}_e^{(p,t)}$
Increase	Increases	Only a significant change in the median $\Delta\mathcal{M}_e^{(p,t)}$ can make $\Delta\widetilde{C}_e^{(p,t)}$ change by K
Decrease	Decreases	Small variations in the median $\Delta\mathcal{M}_e^{(p,t)}$ can make $\Delta\widetilde{C}_e^{(p,t)}$ change by K

Table 5.1: Social Route Control Strategy.

This concludes the design of the adaptive filtering process and the cost $C_e^{(p,t)}$ that will be exploited by the SRC algorithm. The details of this latter are presented in the next section.

5.2.4 The SRC algorithm

The social route control algorithm is described below in Algorithm 2. As in the case of conventional IRC solutions, our SRC algorithm works in a proactive way, leveraging the relocation of traffic even when the desired performance bounds are fulfilled. Its proactivity is automatically adjusted along its operation according to the adaptive processes described above. For the sake of simplicity, Algorithm 2 only describes the stationary operation of the social route control strategy.

Steps 6–8 in Algorithm 2, show that the SRCs always estimate the effects of switching from one egress link to another before shifting the traffic. It is worth highlighting that all existing IRC solutions are able to perform this estimation thanks to their measurement platform.

5.3 Simulation set-up

This section presents the simulation set-up developed to assess the advantages of the social route control model. The performance of our SRCs is compared against that obtained with the following two alternative models: i) the conventional standalone and selfish IRC model; and ii) default BGP routing.

Algorithm 2 SRC($\{p, e, C_e^{(p,t)}\}$)

Input: $\{p\}$ - set of popular destination prefixes of domain S
$\{e\}$ - set of egress links of domain S
$C_e^{(p,t)}$ - Cost of reaching p through e at time t

Output: $e^{best}\ \forall\ p$ - The best egress link to reach every p

1: $R_{th} \leftarrow K$ /* Configurable threshold to trigger the path switch*/
2: *Wait* for changes in the costs $C_e^{(p,t)}$ /* Equation (5.1) */
3: /* Egress links selection process */
4: **if** $\exists\ e'' \neq e\ /\ (C_e^{(p,t)})_{Best} > C_{e''}^{(p,t)}$ **then**
5: Choose $e' = min\{C_{e''}^{(p,t)}\}\ \forall\ e''$
6: Estimate the amount of bandwidth $b_e^{(p,t)}$ to be switched from e to e'
7: /* Estimate if after switching the traffic the cost would still be better in terms of bandwidth */;
Compute $(C_{e'}^{(p,t)})_{Estimate} = C_{e'}^{(p,t)}(S_{e'}^{(p,t)}, AB_{e'}^{(t)} - b_e^{(p,t)})$
8: **if** $(C_e^{(p,t)})_{Best} - (C_{e'}^{(p,t)})_{Estimate} > R_{th}$ **then**
9: Switch traffic towards p from e to e'
10: $e \leftarrow e'$
11: $(C_e^{(p,t)})_{Best} \leftarrow (C_{e'}^{(p,t)})$
12: **end if**
13: **end if**
14: /* End of egress links selection process */
15: Go to Step 2

5.3.1 Evaluation methodology

The simulation tests were carried out using the event-driven simulator J-Sim [64]. All the functionalities of the route controllers were developed on top of the BGP implementation available in this platform, i.e., the BGP Infonet suite.

AS-level Topology: For our simulation tests, the AS-level topology was built using the BRITE topology generator [85]. The topology was generated using the Waxman model with (α, β) set to (0.15, 0.2) [132], and it was

composed of 100 ASs with a ratio of ASs to links of 1:3. This simulated network aims at representing an Internet core composed by ASs of ISPs able to provide connectivity and reachability to stub ASs. We assume that all ISPs operate Points of Presence (PoPs) through which the stub ASs are connected. To emulate the stub ASs sourcing traffic towards popular destinations, we considered 12 ASs uniformly distributed across the AS-level topology. These stub ASs are connected to the routers located at the PoPs of three different ISPs. We considered triple-homed stub ASs because significant performance improvements are not expected from higher degrees of multihoming [3]. To emulate the stub ASs containing popular destinations we considered 25 ASs uniformly distributed across the AS-level topology. This gives an emulation of 12x25=300 IRC flows competing for the same network resources during the simulation runtime.

It is worth highlighting that the size of the AS-level topology used during our evaluations is small compared to the size of the Internet. However, to the best of our knowledge, this is the largest test made to assess the performance of different IRC strategies in a competitive environment. Furthermore, given that smart route controllers operate in short timescales, we assumed that the AS-level topology remains invariant during the simulation runtime.

Simulation Scenarios: In our experiments we run the same simulations separately using three different scenarios:

(a) Default defined BGP routing, i.e., BGP routers choose their best routes based on the shortest AS-path length.

(b) BGP combined with the social route control model at the 12 source domains.

(c) BGP combined with the conventional (i.e. standalone and selfish) IRC model at the 12 source domains.

For a more comprehensive comparison between the different models, we performed the simulations for three different network loads. We considered the following load scale factors (f):

(i) $f = 1.0$, low load corresponding to a traffic occupancy of 45% of the egress links capacity.

(ii) $f = 1.5$, medium load corresponding to a traffic occupancy of 67.5% of the egress links capacity.

(iii) $f = 2.0$, high load corresponding to a traffic occupancy of 90% of the egress links capacity.

Synthetic Traffic and Simulation Conditions: The simulation tests were conducted using traffic aggregates sent from the source domains to each popular destination p. These traffic aggregates are composed by a variable number of multiplexed Pareto flows, as a way to generate synthetic traffic demands, as well as to control the network load during the experiments. The flow arrivals are independently and uniformly distributed during the simulation runtime (i.e., the arrivals are described by a Poisson process). This approach aims at generating sufficient traffic variability supporting the assessment of the different route control strategies.

In addition, we used the following way to generate synthetic traffic demands for the remaining Internet traffic – usually referred to as background traffic. We start by randomly picking four nodes in the network. The first one chosen acts as the origin (O) node, and the remaining three nodes act as destinations (D) of the background traffic. We assigned one Pareto flow for each O-D pair. Next, this process continues until all the nodes are assigned with three outgoing flows (including those in the multihomed stub ASs and those in the ISPs). All background connections were active during the simulation runtime.

Furthermore, the frequency and size of the probes sent by the route controllers were correlated with the outbound traffic being controlled (just as conventional route controllers do [10, 22, 58]).

Finally, we assume that the route controllers have pre-established performance bounds for the traffic under control, i.e. $\overline{D^{(p)}}$ and B_e in our case. For instance, the recommendation G.114 of the International Telecommunication Union – Telecommunication Standardization Sector – (ITU-T) suggests a One-Way-Delay (OWD) bound of 150 milliseconds to maintain a high quality VoIP communication over the Internet. Thus, for VoIP traffic the maximum RTT tolerated ($\overline{D^{(p)}}$) was chosen as twice this OWD bound, that is, 300ms.

5.3.2 Objectives of the performance evaluation

The performance evaluations carried out in Section 5.4 have two main objectives.

Performance Penalties: The first objective of the simulation study is to demonstrate how the social nature of the SRC contributes to reduce the performance penalties associated with frequent traffic relocations. To achieve

this goal, we compared the number of path shifts occurred during the simulation runtime for the 300 competing IRC flows, for the scenarios (b) and (c) in Section 5.3.1. The number of path shifts is obtained by adding the number of route changes that are needed to meet the target RTT bound $\overline{D^{(p)}}$ for each popular destination p.

It is worth highlighting that in both IRC and SRC, the route controllers operate independently and compete for the same network resources. This allows us to evaluate the overall impact on the traffic caused by the interference between several standalone route controllers running at different stub ASs. Thus, while analyzing the results for the different route control models in Section 5.4, it is important to keep in mind that we will be taking into account all the competing route controllers present in the network.

In order to contrast the performance penalties under fair conditions, we made two important decisions. First, we have endowed the conventional IRC controllers with the same randomized control approach used by our SRCs – which was developed in one of our previous works [136]. This approach avoids the appearance of persistent oscillations that might lead to a large number of path shifts in the case of conventional IRC [42]. And second, we have conducted the simulations modeling the same triggering condition for the relocation of traffic in both the IRC and SRC models. The main difference is that in the latter, the social adaptability of the controllers can make that the trigger is reached more often, or less often, depending on variability of the RTTs on the network (see Steps 1 and 8 in Algorithm 2).

End-to-end traffic performance: The second objective of the simulation study is to assess how the different route control strategies aid to improve the end-to-end traffic performance. To achieve this goal, we compared the $\langle RTTs \rangle$ obtained for the 300 flows in the three different scenarios, namely, default BGP, SRC, and conventional IRC.

5.4 Performance evaluation

The left-hand side of figures 5.4, 5.5, and 5.6 illustrate the number of path shifts performed both by conventional IRC and the SRC models in all the stub ASs for the three different load scale factors, f=1.0, f=1.5, and f=2.0, respectively. The number of path shifts is contrasted for different triggering conditions, i.e., for different values of the threshold R_{th} (shown on a logarithmic scale).

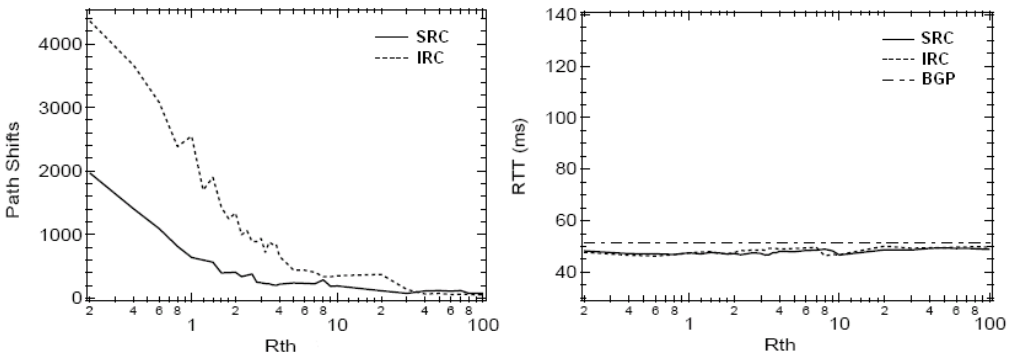


Figure 5.4: Number of path shifts and $\langle RTTs \rangle$ for $f = 1.0$.

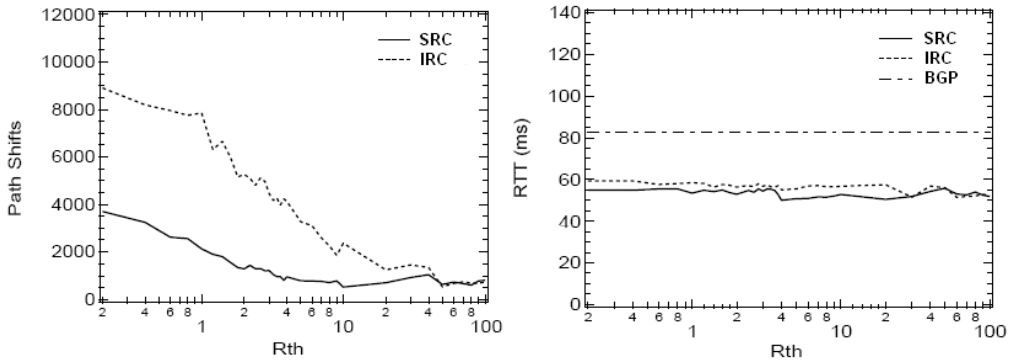


Figure 5.5: Number of path shifts and $\langle RTTs \rangle$ for $f = 1.5$.

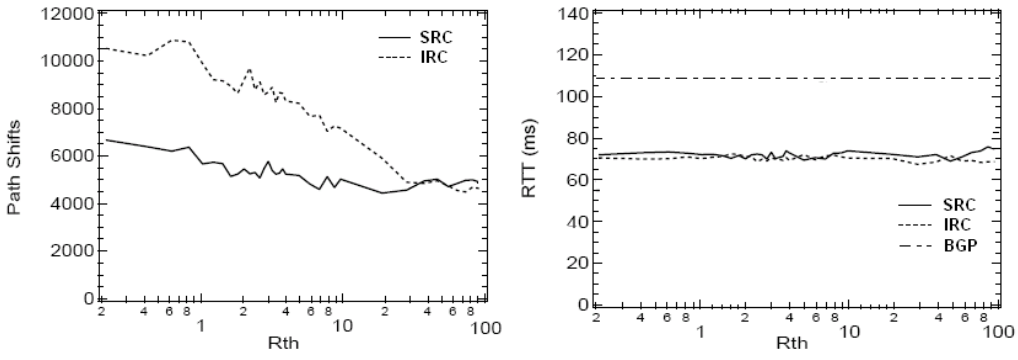


Figure 5.6: Number of path shifts and $\langle RTTs \rangle$ for $f = 2.0$.

Several conclusions can be drawn from the results shown in the left-hand side of figures 5.4, 5.5, and 5.6. In the first place, the results confirm that the SRC model drastically reduces the number of path shifts compared to the existing IRC model[3]. An important result is that the reductions are significant for "all" the load scale factors assessed.

When compared with the conventional IRC model, the SRC model contributes for instance to reductions of up to:

- 77% for R_{th}=1.0 and 71% for R_{th}=2.0 when f=1.0 (see Fig. 5.4)
- 75% for R_{th}=1.0 and 74% for R_{th}=2.0 when f=1.5 (see Fig. 5.5)
- 43% for R_{th}=0.65 and 44% for R_{th}=2.3 when f=2.0 (see Fig. 5.6)

The second major observation is that the reductions in the number of path shifts offered by the SRC strategy become more and more evident as the proactivity of controllers increases, i.e. for low values of R_{th}, which is precisely the region where IRC solutions operate today. It is worth recalling that these results were obtained when both route control strategies were complemented by the same randomized decisions [42, 136]. This confirms that randomized route control decisions help to avoid the potential synchronization of some route controllers, but not in reducing the global number of path shifts in a competitive environment.

On the other hand, when the controllers become less proactive, i.e. for higher values of R_{th}, IRC and SRC tend to behave comparatively the same. Our results reveal that when the route control strategies become excessively reactive – rather than proactive – a social approach does not actually introduce any benefit over a simple randomized technique.

Another important aspect of the results shown on the left-hand side of figures 5.4, 5.5, and 5.6, is that independently of the threshold condition set, and the load scale factor f, a minimum number of path shift is always needed to guarantee the targeted performance.

In order to assess the effectiveness of the SRC model, it is mandatory to confirm that the reductions obtained in the number of path shifts are not excessive, resulting on a negative impact on the end-to-end traffic performance. To this end, we will first analyze the performance of conventional IRC and our SRC "globally", i.e., by averaging over the RTTs obtained by "all" competing route controllers. This is shown from figures 5.4 up to 5.9. The end-to-end performance obtained by "each" route controller individually, is then later in Fig. 5.10.

[3]Clearly, no results are shown for the default BGP routing scenario here, since BGP does not actively perform path switching.

The right-hand side of figures 5.4, 5.5, and 5.6 reveal that – as expected – both SRC and IRC perform much better than BGP $\forall$ f, R_{th}, and the improvements in the achieved performance become more evident as the network utilization increases. In particular, SRC is capable of improving the $\langle RTTs \rangle$[4] by more than 40% for $f = 1.5$, and by more than 35% for $f = 2.0$, when compared with BGP.

Moreover, the $\langle RTTs \rangle$ obtained by SRC and IRC are comparatively the same, and particularly, for $f = 1.5$, SRC not only drastically reduces the number of path shifts, but also improves the end-to-end performance for almost all the triggering conditions assessed. It is worth highlighting that a low value of R_{th} together with a load scale factor of $f = 1.5$, reasonably reflect the conditions in which IRC operates in today's Internet.

Our results also confirm that by allowing more path shifts, some route controllers can slightly improve their end-to-end performance, but such actions have no major effect on the overall $\langle RTTs \rangle$. As mentioned before, a certain number of path shifts is always required, and this amount of path shifts is what actually assures the average performance observed in the right-hand side of the figures 5.4, 5.5, and 5.6.

By analyzing figures 5.4, 5.5, and 5.6 as a whole, it becomes evident that the selection of the best triggering condition actually depends on the load present on the network. The best trade-offs are $R_{th} = 30$ for $f = 1.0$, $R_{th} = 10$ for $f = 1.5$, and $R_{th} = 7$ for $f = 2.0$, which is a reasonable progression to lower values of R_{th}, since the route controllers need less proactivity when the network utilization is low. The corollary of this is that the triggering condition should be adaptively adjusted as well, depending on amount of traffic carried through the egress links of the domain. We plan to address this issue as future work.

Figures 5.7, 5.8, and 5.9, compare the probability distribution of the average RTTs obtained by BGP, SRC, and IRC for the three different load scale factors assessed. To facilitate the interpretation of the results, we use the Complementary Cumulative Distribution Function (CCDF).

An important observation is that under high egress link utilization, i.e., $f = 2$, there is a fraction of $\langle RTTs \rangle$ for which the bound $\overline{D^{(p)}}$ of 300ms cannot be achieved in the case of BGP, whereas both SRC and IRC fulfill the targeted bound.

To complete the analysis, Fig. 5.10 shows the CCDFs of the RTTs for the 12 source domains containing the competing route controllers. The figure shows the results for the three studied scenarios, for all the load scale factors

[4]As mentioned before, this average is computed over the RTTs obtained by all competing route controllers in the network.

assessed, and for $R_{th} = 1$, which as mentioned above, is in the range of operation of the IRC solutions presently deployed in the Internet.

Our results show that the targeted bound of 300 ms is satisfied on average by both SRC and IRC in all cases, and for all domains. Figure 5.10 also shows that IRC generally achieves slightly better performance than SRC, but at the price of a much larger number of traffic relocations: i) $\approx$ 435% larger for f=1.0; ii) $\approx$ 400% larger for f=1.5; and iii) $\approx$ 80% larger for f=2.0, when R_{th}=1.0.

Overall, we conclude that these results support the accomplishment of the two evaluation objectives listed in Section 5.3.2.

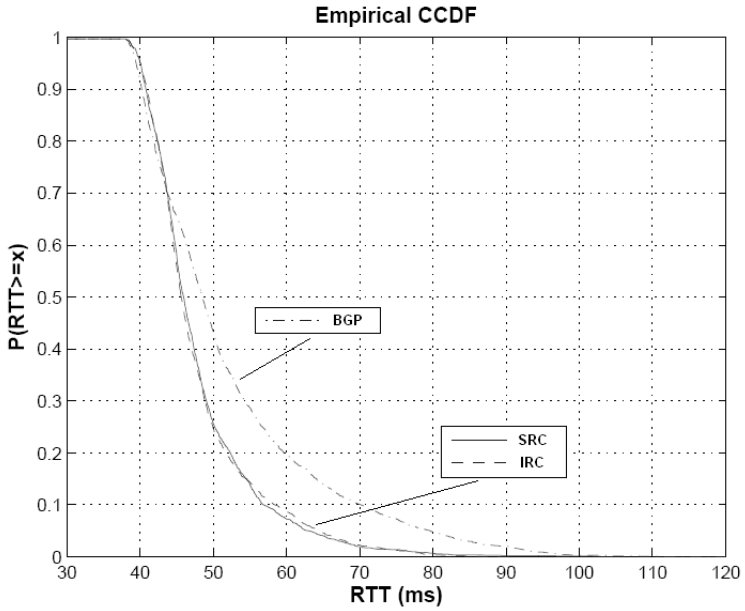


Figure 5.7: Complementary CDF for $f = 1.0$.

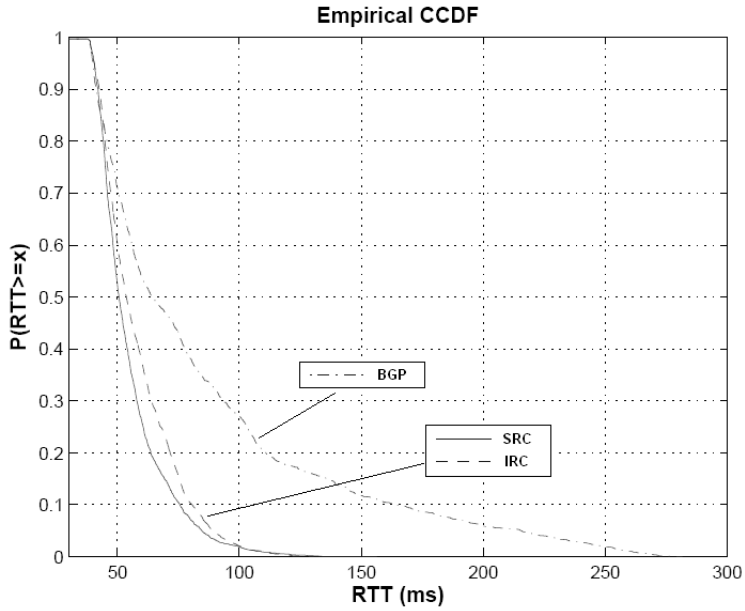


Figure 5.8: Complementary CDF for $f = 1.5$.

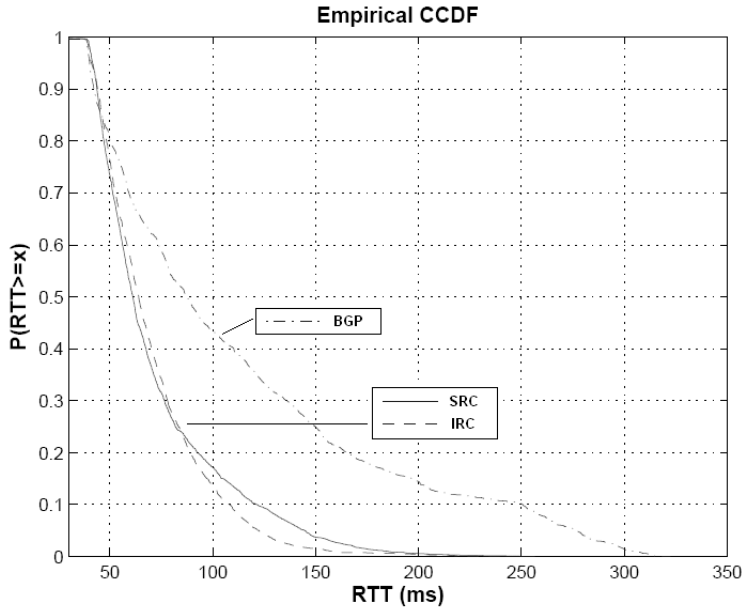


Figure 5.9: Complementary CDF for $f = 2.0$.

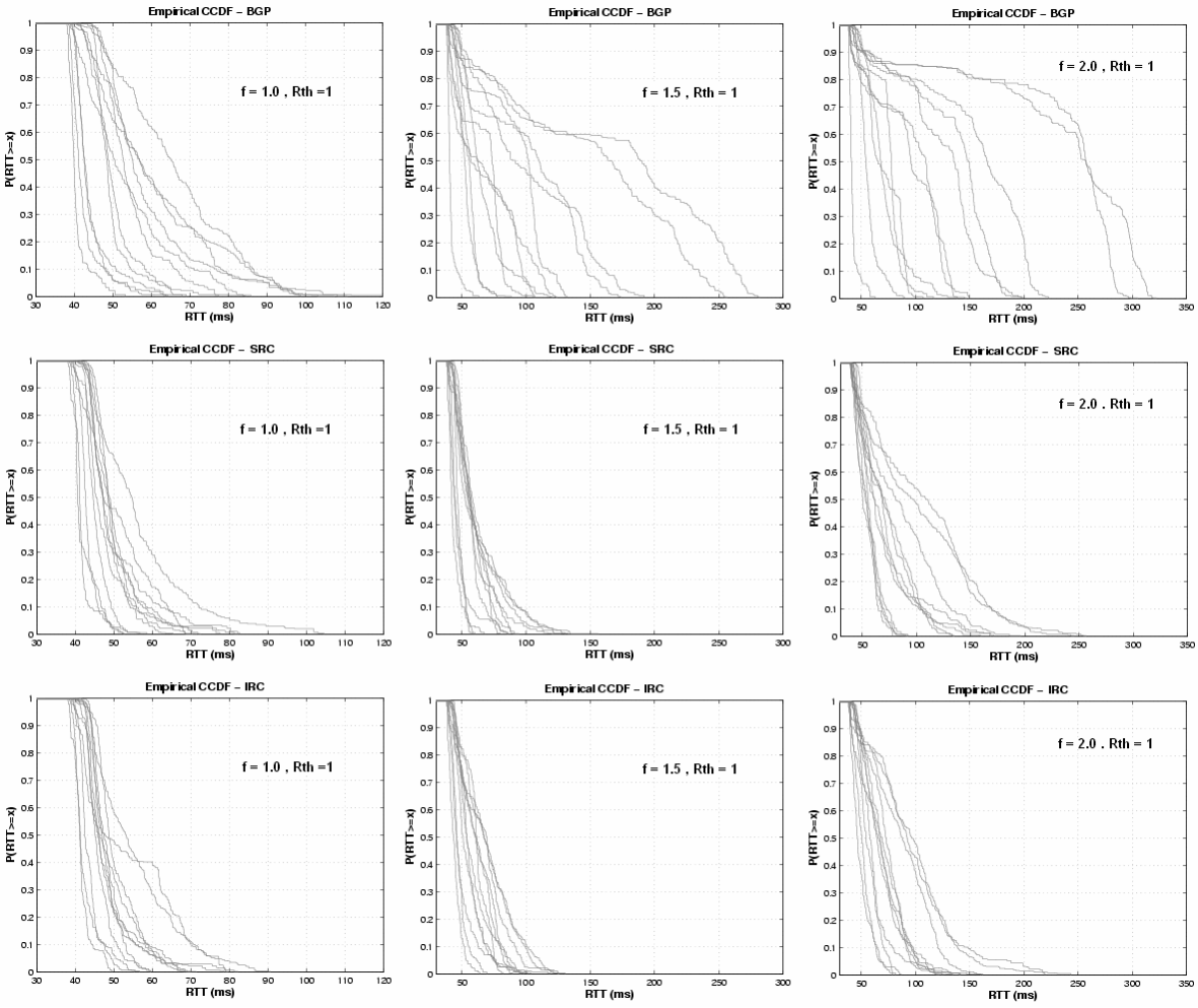


Figure 5.10: CCDFs for BGP, SRC, and IRC.

Chapter 6

Cooperative and Social Route Control (CSRC)

In Chapter 5 we proposed to extend the current IRC model from a standalone and selfish to a standalone and social route control model. In this chapter we propose to move one step further, and extend it to a Cooperative and Social Route Control (CSRC) model.

Our aim is that route controllers belonging to a pair of multihomed stub domains that exchange large amounts of traffic become capable of communicating and cooperating between each other. This cooperation will allow such domains to improve the end-to-end performance of the traffic they exchange either in a one-way or a two-way fashion, depending on their specific needs and the way in which the majority of their traffic flows. An appealing advantage is that either of the two ASs can challenge the other to start the cooperation, which can be exploited by an AS to smartly control part of its inbound traffic, something which is unfeasible with standalone route control solutions.

The cooperative route control model proposed here takes full advantage of the social behavior developed in Chapter 5, so our cooperative controllers are endowed with the SRC algorithm. Our main contribution in this chapter is to show that with this new extension it is possible to outperform the existing standalone route control model, and the only thing needed is a software upgrade of the available route controllers. Figures 6.3, 6.4, and 6.5 support this claim.

We proceed now to introduce the cooperative route control model, as well as the communication protocol proposed supporting the cooperation between the route controllers.

6.1 The Cooperative Route Control (CRC) model

A major incentive for cooperation is to improve the way in which conventional route controllers monitor the network. In addition to passive measurements, all the route controllers available today, perform active probing for a set of popular destinations through all the available egress links of the AS. A route controller constantly evaluates the end-to-end performance of these probes and selects the best path to route the traffic (and hence the best egress link), based on the lowest latency measured. Unfortunately, the current standalone controllers only consider the RTT latency, so they can only take coarse-grained routing decisions given that they decide how to route outbound without taking into account the potential asymmetry of the paths in the Internet.

Furthermore, conventional route controllers perform the active measurements directly against the end-systems, so their success and precision actually depends on the willingness of these latter to accept and reply ICMP, UDP, and TCP probes. In a cooperative framework, the route controllers can exploit the benefits of one-way measurements, such as One-Way Delays (OWDs), which can be performed directly between the route controllers, so the success and precision of the measurements becomes independent of the end-systems[1].

Another key incentive for cooperation between domains are the potential benefits in terms of inbound traffic control. Standalone solutions are only able to control outbound traffic, which results appropriate for domains that serve data to the greater Internet, but not for those that mainly receive data from this latter. However, if both the source (S) and destination (D) domains could count with a Cooperative Route Controller (CRC), then the CRC in D could challenge the one in S to monitor and control the performance of the traffic flowing from S to D (see Fig. 6.1). The CRC in S could either accept or refuse to carry out such task depending on its own policies, its current load, and its particular needs.

With this in mind, we define "*cooperation between two distant domains*" as an association by which two peering CRCs can agree upon a set of performance bounds, carry out one-way measurements, and exchange notification messages, either in a one-way or a two-way fashion.

[1]We assume that the distant domains sourcing and sinking the traffic are such that the difference between the exact end-to-end OWD of the traffic, and that measured between the route controllers belonging to the ASs is negligible.

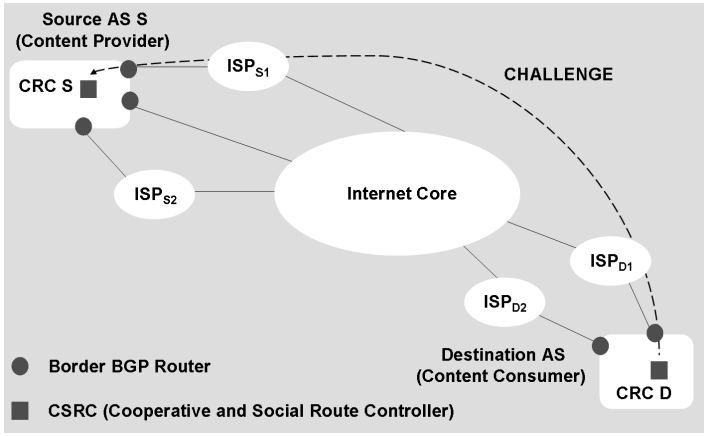


Figure 6.1: Cooperation between two distant CRCs.

This cooperation is supported by a reliable communication protocol between peering CRCs, which we describe next.

6.1.1 The CRC protocol

Given that the measurements require the accurate computation of the OWD, we assume that the CRCs are properly synchronized (e.g., by means of GPS) and the details concerning synchronization are out of the scope of this thesis.

DISCOVERY: A mechanism is needed to locate distant CRCs before the cooperation actually starts. An appealing option is to rely on the extensible nature of the DNS, and follow a similar approach to the one proposed by Bonaventure et al in [17]. With this approach, a new Resource Record (RR) called CONTROLLER can be added in the reverse DNS, as a pointer to a CRC. When a CRC wants to locate the CRC hosting a given prefix (either a popular source or destination prefix) it only needs to perform a reverse DNS lookup for the prefix, and ask for the CONTROLLER's address.

HANDSHAKING: Once the distant CRC is located the handshaking process shown in Fig. 6.2 starts. As mentioned above, this process can be started by initiative of either the source AS or by the destination AS. The maximum tolerated one-way performance parameters of the traffic to be monitored and controlled are negotiated and exchanged during the handshake. The outcome of a successful negotiation could be either that both CRCs are going to send probes to each other or that only one of them will do. This depends on the asymmetry of their traffic exchange, their local policies, and

their particular needs. After this negotiation, both CRCs have synchronized their clocks so as to perform the one-way measurements.

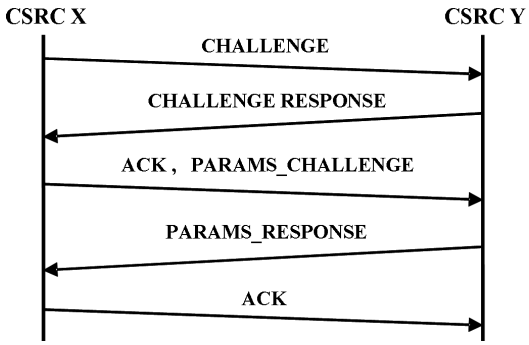


Figure 6.2: Handshake between two distant CSRCs.

KEEPALIVE: These messages are needed because the CRC sending the probes needs to be sure that the CRC receiving the probes is actually performing the OWD measurements and remains alive. Thus, the CRC receiving the probes is the one that sends the KEEPALIVEs.

MEASUREMENTS and NOTIFICATIONS: Once the initial negotiation has finished the communication between the CRCs in Fig. 6.1 continues as follows – for simplicity we assume that the interest is just to monitor and control the traffic flow from S to D. S sends probes to D through all the available egress links at S from which D is reachable, just as conventional standalone route controllers do today[2]. This means that D is receiving a set of probe flows – one for each egress link at S as determined by the BGP routes available towards D. D performs the one-way measurements and, instead of computing the median RTTs as it was the case in Chapter 5, it computes the median OWD for each of these probe flows. If no performance changes are detected by D, i.e., all the medians remain unchanged, only KEEPALIVE messages are sent back from D to S. Since in this example we assumed only one-way control between S and D, D does not probe S in this case. As in Chapter 5, the median values are computed through a sliding window so as to leverage the notification reactivity of D. In case that any of

[2]We recall that the route control strategies proposed in this thesis do not introduce any changes to the measurement systems of conventional IRC.

the median OWD changes, or any other relevant event, D notifies S, so that the adaptive and social route control algorithm running on S can decide if the corresponding traffic needs to be switched or not to an alternative egress link of S. A high-level description of the CSRC algorithm is provided in the next section.

6.1.2 The CSRC algorithm

From a functional standpoint, once the cooperation is established and the measurements are being taken, the CSRC algorithm works exactly in the same way as the SRC algorithm described in Chapter 5. The only relevant difference between CSRC and SRC, is that in the cooperative case the adaptive filter and the grid operate using OWDs instead of RTTs.

Figure 6.3 provides a high-level view of the interactions between a pair of cooperative route controllers.

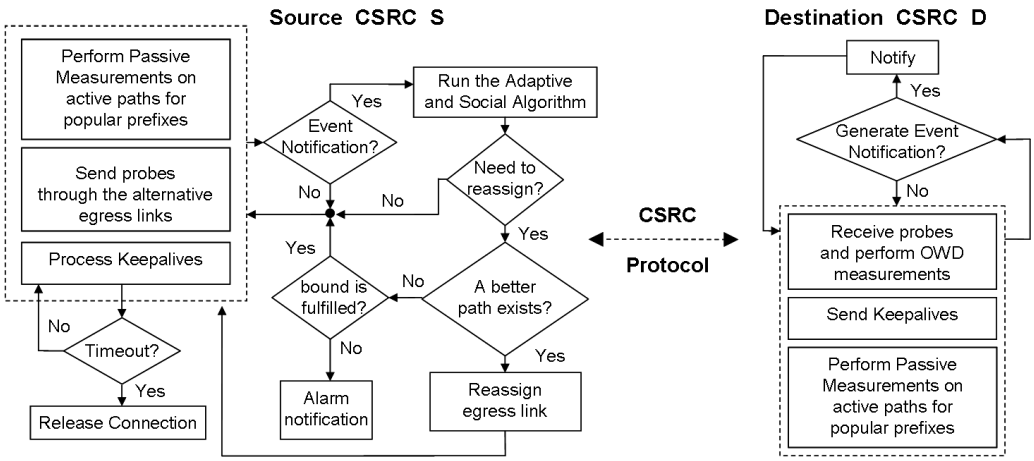


Figure 6.3: High-level description of the CSRC interactions.

6.2 Performance evaluation

The simulation set-up as well as the evaluation methodology that we follow here are the same used in Chapter 5. Please refer to Section 5.3 for the details (e.g. we consider once again 12 source ASs, 25 popular destinations, etc.). The following scenarios are tested in this case:

(a) Default defined BGP routing, i.e., BGP routers choose their best routes based on the shortest AS-path length.

(b) BGP combined with the conventional standalone and selfish route control model at the 12 source domains.

(c) BGP combined with a cooperative route control model at the 12 source domains, but without running the social algorithm.

(d) BGP combined with a cooperative and social route control model at the 12 source domains.

Once again, we performed the simulations for the three previous load scale factors f. The objectives of these tests are essentially the same as those in Chapter 5, i.e., to indirectly assess the performance penalties associated with too frequent path switches – by accounting, precisely, the frequency of the path shifts – and, to analyze the end-to-end performance of the traffic during the simulation runtime.

6.2.1 Outbound traffic improvement

Figure 6.4.a illustrates the average frequency of path shifts performed in all the stub ASs for the three different load scale factors. Our results reveal that the cooperative and social route control model drastically reduces the frequency of path shifts compared to both the conventional model and a cooperative model without exploiting the strengths of the social route control algorithm. An important result is that the average reductions are significant for all the load scale factors assessed. When compared with the conventional route control model, the cooperative and social model contributes to reductions that vary between 50% for f = 2, up to 73% for f = 1.5 (for simplicity all the results shown here are for $R_{th} = 1$.

Figure 6.4.b on the other hand, allows us to observe the average frequency of path shifts on a per-domain basis. This is shown only for the highest load scale factor, i.e. f = 2, since the results obtained for the other two load scale factors are similar and do not supply additional information. The most important things to notice from Fig. 6.4.b are:

(i) When contrasting the conventional route control model against the cooperative and social route control model, all the competing ASs are able to reduce the frequency of their path shifts, and hence reduce the associated performance penalties.

ii) These reductions are indeed significant for all the stub ASs, except for AS10, which only obtains a marginal improvement.

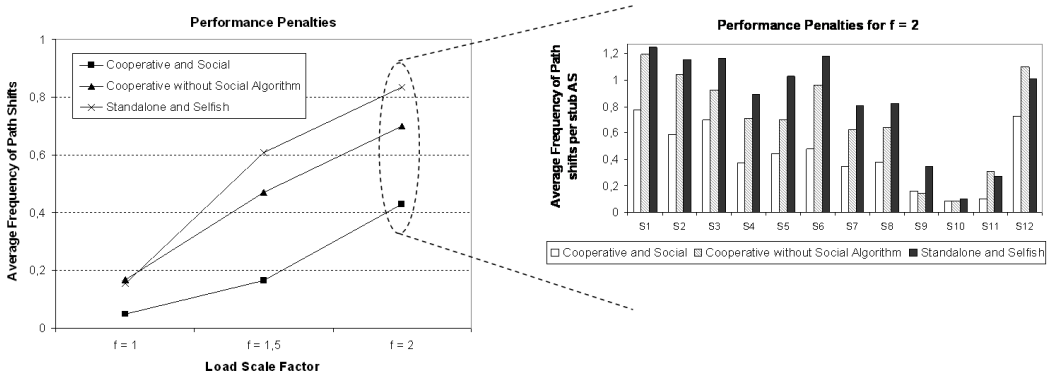


Figure 6.4: Evaluation of the performance penalties.
(a) Average frequency of path shifts for the three different load scale factors, and for $R_{th} = 1$. (b) Average frequency of path shifts per-source AS, for $f = 2$.

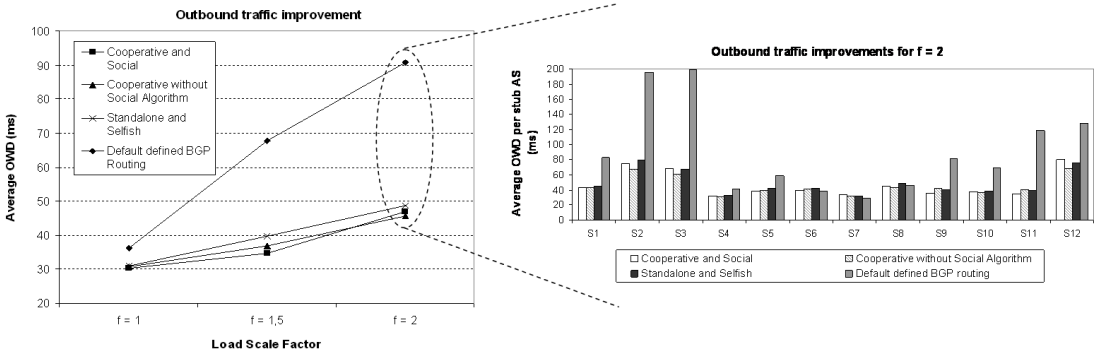


Figure 6.5: Evaluation of end-to-end traffic performance.
(a) Average one-way latency for the three different load scale factors. (b) Average one-way latency per-source AS, for $f = 2$.

Similarly as we did in Chapter 5, in order to assess the effectiveness of the cooperative and social route control model, it is mandatory to confirm that the reductions obtained in the performance penalties are not excessive.

The results in 6.5.a show that the default defined BGP routing scheme has the worst average one-way latency for all the load scale factors considered.

This was naturally expected, since it is a well known fact that the best paths chosen by BGP are usually not correlated with the paths exhibiting the best end-to-end performance for the users' traffic.

Figure 6.5.a shows that the average OWD is drastically reduced when any of the intelligent route control solutions is used. The figure reveals that the three route control models assessed show almost the same end-to-end performance when the network load is rather low ($f = 1$). When the network load gets higher ($f = 1.5$), the two cooperative route control models are able to improve the average OWD when compared with the conventional standalone and selfish route control model. The relative improvements against this latter are, 7.5% for the cooperative model without exploiting the social route control algorithm, and 12.5% for the cooperative and social model. For the highest load scale factor ($f = 2$) the cooperative models still perform better than the conventional route control model, but the relative improvements are less than for $f = 1.5$. The relative improvements are now 6% for the cooperative model without exploiting the social route control algorithm, and 4% for the cooperative and social model.

Social improvements are usually not for free, and in our case this is confirmed in Fig.6.5.b. This figure allows us to observe the average OWDs obtained on a per-domain basis. Once again, this is only shown for the highest load scale factor, i.e. for $f = 2$. The results obtained for the other two load scale factors are consistent with these. Figure 6.5.b shows that for two ASs, namely, AS7 and AS12 the cooperative and social model performs slightly worse than the conventional route control model. Nevertheless, the average OWD penalties are only about a few milliseconds, and Fig.6.4.b confirms that both ASs achieve significant reductions in terms of path shifts.

6.2.2 Inbound traffic improvement

Finally, we have compared the inbound traffic improvements relative to BGP-based route control. Figure 6.6 shows the average OWD reductions obtained for the different load scale factors. Clearly, the conventional standalone and selfish route control model is not assessed in this case, since it cannot be exploited for inbound traffic control. The results in Fig. 6.6 reveal that the improvements in terms of one-way latency are large, and as expected, the improvements are especially noticeable for higher load scale factors. The results in Fig. 6.6 were obtained when all the sources accepted the challenges from the destination domains. Thus, depending on the local policies of the source domains, the average improvements can be less than the ones shown here, especially when one or more sources start to reject the challenges received.

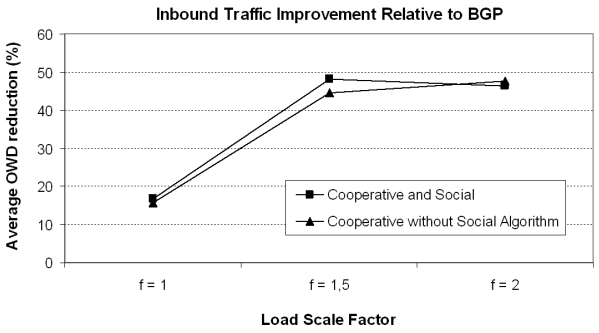


Figure 6.6: Potential inbound traffic improvement.

6.3 Conclusions on IRC

The most important conclusions that can be extracted from the studies in this part of the thesis are:

Cooperative and/or social route control models not only drastically reduce the penalties associated with frequent traffic relocations, but also supply almost the same – and in several cases even better – end-to-end traffic performance for all the load scale factors assessed. This suggests that a large number of the path shifts performed by conventional route controllers are actually unnecessary in competitive environments.

Two simple extensions, such as the introduction of a route control protocol and a modified route control algorithm, are enough to outperform the existing route control model. These extensions can be incrementally introduced and used today as software upgrades, leveraging the cooperation and social behavior of the existing route controllers. Our first contribution has been to show that with these two extensions, the performance penalties can be drastically reduced on average and still obtain globally better end-to-end traffic performance. Our second contribution has been to show the potential benefits of these extensions in terms of inbound traffic control.

It is important to highlight that the extensions proposed here do not compromise the scalability of the current route controllers. The existing route control solutions are able to monitor and control more than 100 popular destination prefixes along all available egress links of the source domain [10,

22, 58]. Our extensions neither modify the core of the monitoring system nor intend to increase the scale of these solutions. On the contrary, our aim is to endow the route controllers with mechanisms that allow them to "socially" deal with their intrinsic selfishness.

Part III

Route Control: Near Future

Chapter 7

IP/MPLS Multi-Domain Networks

MPLS is being actively adopted as the core switching infrastructure at the intra-domain level. This trend is mainly due to its strengths in terms of VPNs management, TE, QoS delivery, path protection and fast recovery from network failures.

In this chapter we review the drivers behind the expected extension of MPLS Label Switched Paths (LSPs) across domain boundaries. We also deepen in the analysis of the limitations imposed by the current multi-domain routing and TE control model in order to improve the performance and reliability of the inter-domain communications by means of MPLS LSPs. Among the problems analyzed here are the lack of a TE information exchange model between domains, the issues associated with policy-based routing, and the scarce number of inter-domain paths available in practice.

The IETF has recently standardized an architecture [34, 97][1], which offers a suitable MPLS framework that can potentially tackle most of these problems. However, the advances made so far at the IETF are still centered on intra-domain issues. Among the problems that remain unsolved are:

(i) How to exploit the model proposed by the IETF to efficiently find and establish optimal – or near-optimal – primary and protection inter-domain LSPs subject to QoS constraints.

(ii) If we could count with a solution tackling the previous item, further research is needed to provide strategies on how operators could exploit as much as possible the advantages of having long-lived inter-domain MPLS coverage, against the extra cost that such coverage would represent.

The goal in this chapter is to explore the major limitations hindering the deployment of primary and protection LSPs across multiple domains, in the

[1]This architecture is described in Section 7.3.

context of the current inter-domain network model. We describe the critical problems faced by the research community today, and discuss about how to overcome the problems exposed. Detailed solutions to items (i) and (ii) are presented later in Chapters 8 and 9, respectively.

7.1 Drivers

Many research efforts have been – and are still – devoted to improve different facets of MPLS in the context of a single domain. At present, a significant part of these efforts are expected to move into the inter-domain area. On the one hand, customers are requiring from their ISPs the capability to extend their MPLS-based Layer 2 and Layer 3 VPN services across domains. Such services typically support some mission-critical applications and IP telephony, demanding hard QoS guarantees and fast restoration capabilities from the network. On the other hand, ISPs are eager to offer these services.

Another clear incentive is for content providers, since they can exploit aggregate MPLS paths to reach geographically spread groups of consumers, without maintaining a large number of distributed replicas of their delivery platforms; it is also easier for them to adapt when the distribution of major consumers changes.

In addition, providers are trying to offer new and value added services, which will require some means of guaranteeing the quality and reliability of their customers' communications – even when the other end-point of the communication is outside their administrative domain [83]. Guaranteed quality and high reliability are features that providers are also seeking for the expansion of their VPN offer and content distribution models. To achieve this, two service providers can negotiate a peering agreement [54, 83], and now use long-lived MPLS paths to supply QoS support and high reliability to a set of aggregate traffic flows between them. It is worth noticing that this MPLS-based peering model applies even when the domains are not directly connected, supporting the idea of having transparent shortcuts between distant providers. A limiting fact, however, is that end-to-end MPLS connectivity is only feasible if intermediate transit providers support the establishment of MPLS paths through them.

A recently chartered Working Group (WG) by the IETF has started to address the issue. Their first contribution is the introduction of a new network component inside each domain called the *Path Computation Element* (*PCE*). The WG has already standardized a PCE-based architecture and the requirements for the communication protocol between PCEs [34, 8]. The PCE WG is expected to draft solutions and provide guidelines for a wide

range of unsolved problems, including:

(i) The extension of MPLS Traffic Engineering (MPLS-TE) capabilities across domains.

(ii) The design of novel communication protocols to handle requests for the computation of paths subject to multiple constraints, within, and between domains [126].

(iii) The definition of the extensions needed for some of the existing routing and signaling protocols.

From this range of open problems, in this chapter we focus on exploring the major limitations hindering the deployment of primary and protection inter-domain LSPs for mission-critical services subject to given QoS constraints. The interest here, is in advance path protection strategies, i.e., backup paths need to be established jointly with the primary LSPs. The rationale for this approach is that in many practical settings, it might not be possible to restore all QoS protected paths after a failure. This typically depends on the type of failure, and the amount of traffic that needs to be restored. Furthermore, restoring inter-domain QoS LSPs after a failure might take an unacceptably long time for a number of mission-critical applications. Thus, for this kind of applications, switching promptly from a primary to a backup path in the event of a failure can be guaranteed by provisioning two disjoint QoS paths between the source and destination nodes. The subject of this chapter is to explore the challenges in doing so at the inter-domain level. As we will show, the problem of finding two disjoint QoS paths in the context of the current inter-domain network model yields solutions that are far from optimal.

7.2 Existing limitations

The current inter-domain network model introduces a series of limitations that hinder the computation and establishment of high quality disjoint (primary and protection) LSPs across domains. These limitations can be grouped into three categories:

(A) Lack of a model for TE information exchange between domains.

(B) Policy-based routing.

(C) Scarce path diversity.

(A) Lack of a model for TE information exchange between domains

At present, the information exchange between domains at the control plane level is conveyed by the inter-domain routing protocol, i.e., BGP. Although BGP supports the distribution of some limited TE information (see Section 3.2), in practice, BGP only advertises reachability information between domains. BGP routers never exchange network "state" information, such as path bandwidth utilization, or path delays, which are essential for TE purposes. Furthermore, BGP routers are completely unaware of the topology of the Internet. A BGP router handles destination prefixes, and the next-hop to reach each destination. This approach has proven to supply a scalable inter-domain control plane. Unfortunately, it hinders the deployment of TE mechanisms capable of coping with the existing QoS and resilience demands at the multi-domain level. Overall, at present there is neither a model nor a valuable mechanism for distributing TE information (or TE demands) among domains.

(B) Policy-based routing

As described in Section 2.3 , there are two types of business relationships between domains, i.e., customer-provider and peer-to-peer, which correspond to the two different traffic exchange agreements between neighboring domains. These relationships determine the export policies of the ASs indicated in Section 2.3.

The top of Fig. 7.1 illustrates the effect of the export policies. The figure shows six interconnected ASs. Let us suppose that AS1 is a customer of AS2 and AS3, which are in turn peers of AS4. Let us also suppose that AS2 and AS3 are peers. In addition, AS5 is a customer of AS4, and AS6 is a customer of both, AS3 and AS4. The arrows in the figure represent the flow of BGP advertisements for the set of prefixes owned by AS4, according to the export policies. At a pure AS-graph level, AS3 has four possible paths to reach AS4, i.e., one through AS1, one through AS2, one through AS6, and the one directly linked to AS4. However, the export policies determine that the path directly connecting AS3 and AS4 is actually the only one available for AS3.

The overall effect of the export policies is two-fold. First, inter-domain routes cannot be inferred from the topology. These set of rules turn inter-domain routing into policy driven rather than topology driven or network state driven, so finding disjoint paths across domains is – at least at present – strictly limited by these rules. Second, the algorithms for finding optimal disjoint QoS paths typically rely on a directed graph abstracting the network

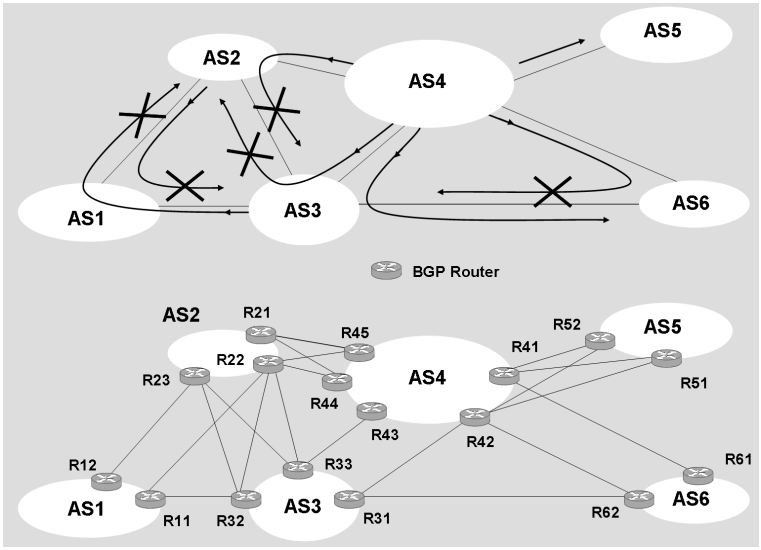


Figure 7.1: Export policies and scarce path diversity issues.

topology. However, in [113] we show that a multi-domain network cannot be abstracted as a directed graph in the presence of the export policies. Thus, efficient intra-domain algorithms such as the ones proposed in [93] cannot be simply extended for AS-diverse routing.

(C) Scarce path diversity

In addition to the reduction in the candidate paths due to the export policies, other factors contribute to the problem of the scarce path diversity between nodes located in distant ASs. The power-law relationship of the Internet topology, which was first reported in [32], is one of the main contributors to the problem. It reveals the hierarchical nature of the Internet and exposes the issue that only a very few highly connected transit ASs keep the Internet as a whole. At present, around only twenty of these large transit ASs exist [116], which means that, at the AS-level, the core of the Internet is very small. It also means that the ASs located at the edge of the Internet tend to connect to this highly connected group of ASs, which translates into very few AS-paths between distant ASs.

Another main contributor to the scarcity of paths is BGP. BGP introduces two major limitations. As described in Section 3.1.4, while a BGP routing table typically contains more than one candidate route towards a destination prefix, BGP routers allocate only one route (the best route) in the forwarding table. BGP routers typically select the shortest AS-path as

the best route. This route is the one they use to forward packets and the only one advertised to other BGP peers. This reduces the number of routes handled by upstream domains, supplying a scalable routing approach, but unfortunately it drastically reduces the path availability information flowing upstream.

The second limitation introduced by BGP in terms of path diversity is that, for the sake of scalability, BGP handles and advertises highly aggregated information. To be precise, the reachability information advertised by BGP routers only contains AS-path information, that is, a set of destination prefixes and the list of AS hops that need to be traversed to reach those destinations. Such a list of AS hops offers highly aggregated information by completely hiding the internal structure of the ASs. The advantage of this lack of internal visibility is that it makes BGP highly scalable. A disadvantage, however, is that although several disjoint paths might be available along an AS-path, they cannot be determined. For example, the bottom of Fig. 7.1 discloses the internal structure of the ASs in the top of the figure. For the sake of simplicity we have only depicted the border routers. Without loss of generality, we assume high path diversity between the nodes inside the ASs. Figure 7.1 shows that at the AS-graph level, there are no disjoint paths between AS1 and AS5 (all available paths traverse AS4). Yet, at the router-level, there are in effect several disjoint paths between the nodes in AS1 and AS5. In order to assess how some of the above limitations affect the number of disjoint paths between domains, we have conducted two different experiments in [139] that we detail here.

Experiment 1 – The goal of the first experiment is to study how the power-law relationships of the Internet topology contribute to the scarceness of link-disjoint paths at the AS-level. We compared the number of disjoint paths using ten AS-level topologies generated by means of the BRITE topology generator [85]. We used two different models for generating the test topologies: a Waxman model and Barabasi-Albert model. A Waxman model uses a probability function for interconnecting nodes based on the distance that separates them on the plane [132]. In this model, the node degrees are uniformly distributed, and hence they do not follow a power-law. A Barabasi-Albert model establishes links based on the preferential attachment principle [11]. This model follows a power-law. We used the default parameters provided by BRITE, i.e., α=0.15, and β=0.2, for the Waxman model.

All topologies have the same number of ASs and links, namely, 100 ASs and 400 links. For each topology we computed the number of link-disjoint paths between each pair of ASs. The average number of disjoint paths for

topologies belonging to these models is depicted in Fig. 7.2. The figure compares the percentage of AS pairs that have at least n disjoint paths, for each n $\geq$ 1. Our results show that for small values of n, the power-law topology has a smaller number of disjoint paths. For example, in the Waxman model, 57% of the AS pairs have at least 4 disjoint paths, compared with just 26% for the Barabasi-Albert model. This means that almost three quarters of the AS pairs have less than 4 disjoint paths in a power-law topology and this is just from the topology perspective. Additional reductions need to be considered after introducing BGP and the export policies.

The tail of the distribution shows that only a small number of ASs have a large number of disjoint paths between them. This small group of ASs represents the highly connected core of the Internet, which is almost a full-mesh. Unfortunately, most of the candidate disjoint paths between the ASs in the core are unavailable in practice, due to the export policies between domains (recall that a provider does not supply transit for the packets exchanged between its peers).

Experiment 2 – Our goal in this experiment is to study the effect of topology aggregation on the maximal number of node-disjoint paths at the router-level. With topology aggregation, the ASs do not reveal the details of their internal structure, but rather supply an aggregated representation to the outside world. Using BRITE we constructed several hierarchical network topologies that include 20 ASs, 200 nodes and 874 links. The topologies are constructed using a top-down approach. A set of ASs is generated first, according to a Barabasi-Albert model. Next, for each AS in the AS-level topology, BRITE generates a router-level topology using a Waxman model. Once again, we used the default parameters provided by BRITE for both models. Next, we constructed a corresponding aggregated topology by using a virtual node model. In such model, each AS is substituted by a single node, while two parallel links between the same pair of ASs are substituted by a single link. We then computed the maximum number of node-disjoint paths between each pair of routers for all topologies. We used node-disjoint paths because the aggregated topology does not provide information about the availability of link-disjoint paths that run through each AS. Next, we compared the percentage of router pairs that have at least n disjoint paths for each n $\geq$ 1 with and without aggregation. The experiment results for a typical topology are depicted in Fig. 7.3.

The results show that the number of disjoint paths in the aggregated topology can be up to 30% less than that in the original topology. Further reductions need to be considered after introducing the BGP decision process and the export policies in the aggregated topology (see [70] for simulation

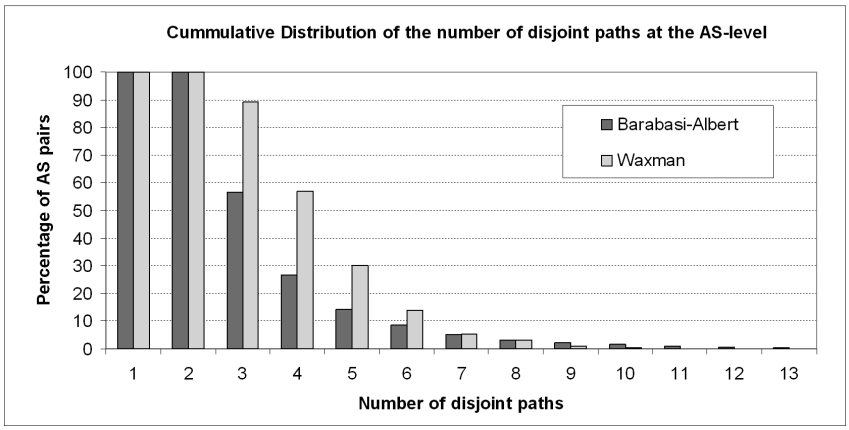


Figure 7.2: Number of disjoint paths in a power-law topology.

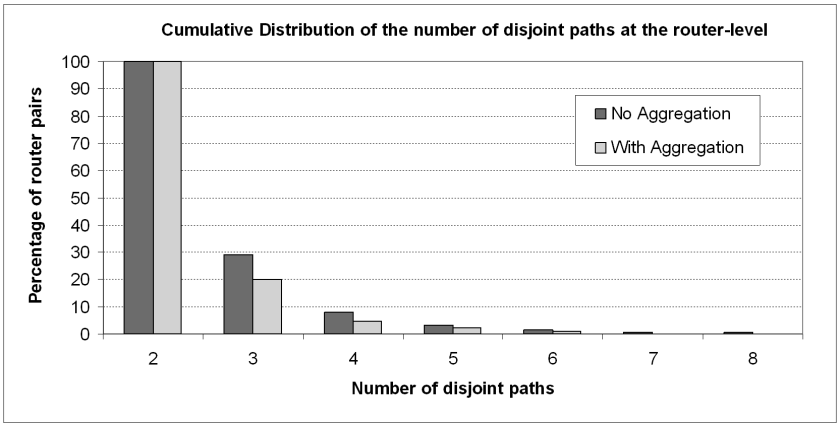


Figure 7.3: Number of disjoint paths with and without aggregation.

results about the scarcity of paths due to BGP).

Overall, the power-law relationship among ASs, together with the limitations imposed by BGP aggregation and the export policies, make AS-graphs inadequate to find disjoint paths across domains. Given that the AS-paths are the only information available in practice that can be used for inter-domain TE purposes, the provisioning of disjoint LSPs with QoS constraints is simply unfeasible in the framework of the current inter-domain network model.

7.3 The Path Computation Element

The limitations exposed above have motivated the creation of the PCE WG within the IETF. The aim of this initiative is to standardize a PCE-based model to distribute the computation of TE LSPs among different areas of a single domain or within a small group of domains. This model is not considered to be applicable to the entire Internet, and this stems from the fact that there is no such demand at the moment. Most of the ongoing work at the IETF is still focused on inter-area (single domain) issues. Even though the inter-domain case has begun to be analyzed, the discussions are in an early stage. This section provides an overview of the key aspects of this model, and succinctly explores its possibilities in terms of provisioning primary and backup QoS LSPs across domains. Besides a few recent standardizations [97], most of the work in the WG is in the draft stage. Many issues remain open, so from the alternatives that are being discussed, we present the one that we consider supplies the most practicable approach.

This approach proposes a decoupled architecture, in which path computation tasks are performed by a device that is detached from the head-end MPLS Label Switching Router (LSR). Such device is referred to as the PCE. Each domain may allocate one or more PCEs depending on its size. For instance, large transit domains can be split into several areas, and use one PCE to handle the path computations within each area. For the distributed computation of inter-area LSPs, a communication protocol is used between the PCEs of the involved areas [126]. Actually, the same model applies at the inter-domain level, so the set up of LSPs spanning multiple domains involves at least one PCE per domain [34].

Each PCE is capable of computing primary and backup QoS paths within a domain or an area of a domain. To accomplish this task, the network state information of the domain (area) is gathered into a Traffic Engineering Database (TED). The TED is fed by the intra-domain routing protocols (e.g. OSPF-TE or IS-IS-TE) and "raw" BGP information, i.e., by the set of BGP routes that are available before BGP chooses the best route. This increases the number of candidate paths inside the TED. The PCE uses the information contained in the local TED to find primary and backup QoS paths by means of heuristics especially designed to tackle the intractability of the path computation problem [93]. By detaching the path computation tasks from the routers, dedicated PCEs can relief the LSRs from intensive computations such as finding disjoint QoS paths.

The WG has already drafted the first version of the communication protocol between the LSRs and the PCEs as well as between cooperating PCEs [126]. In [126] the LSRs are termed Path Computation Clients (PCCs). The

protocol specifies both the PCC-PCE communication, and the PCE-PCE communication for the distributed computation of LSPs. The PCC-PCE part of the protocol supports path requests subject to multiple QoS constraints; it is able to return multiple (disjoint) paths, and takes into consideration features such as security and policies. Accordingly, some of the limitations exposed in the previous section are partially addressed by means of this approach. Figure 7.4 illustrates the PCE-based architecture. The LSR0 in AS0 is the head-end of a requested LSP toward a destination node located in a distant AS (not depicted in the figure). When LSR0 receives the LSP request, the following sequence of actions occurs:

(1) LSR0 requests PCE0 to compute the path.

(2) PCE0 queries the TED in AS0 and computes the segment of the inter-domain LSP up to the Next-Hop (NH) AS Border Router (ASBR). If more than one candidate path exists, the heuristic algorithm in PCE0 selects the "best" segment towards the destination (we will discuss this selection process in the next section). Suppose that PCE0 selects ASBR11, so it responds LSR0 with a set of strict hops toward this node. Notice that the NH ASBR denotes the ingress ASBR to the downstream domain, so the NH ASBR and the PCE computing the local segment of the path belong to different domains.

(3-4) These steps represent the signaling messages, i.e., the resource reservations and explicit path routing performed by a protocol like the Resource Reservation Protocol-Traffic Engineering (RSVP-TE).

Once the signaling messages reach ASBR11, the same process occurs inside AS1, which is represented as the actions from (5) to (8), and this process is repeated on a per-domain basis until the destination AS is reached.

Figure 7.5 shows a more detailed description of the sequence of actions and the role of the different protocols involved in the set up of an inter-domain LSP. The distributed path computation approach explained above is referred to as Explicit Route Object (ERO) expansion [125]. The name comes from the RSVP-TE ERO, which allows signaling a mix of strict and loose hops to be used in the path. A hop may be even an "abstract" node such as an entire AS. Abstract and loose hops are expanded inside each transit domain to a set of strict hops between the ingress ASBR and the NH ASBR.

This approach has two practical advantages. First, it supplies a scalable path computation scheme, since the responsibility and "visibility" of each PCE ends up in the corresponding NH ASBR. Second, it supplies an appealing approach to ISPs, since it leverages confidentiality by hiding the internal

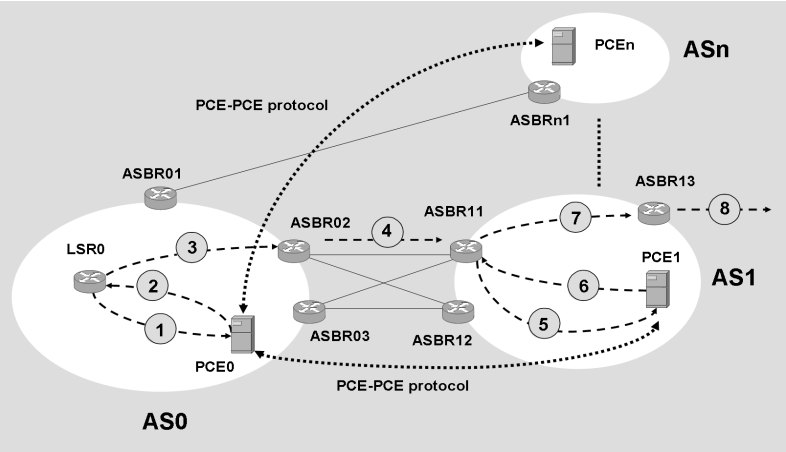


Figure 7.4: Per-domain LSP computation based on ERO expansion.

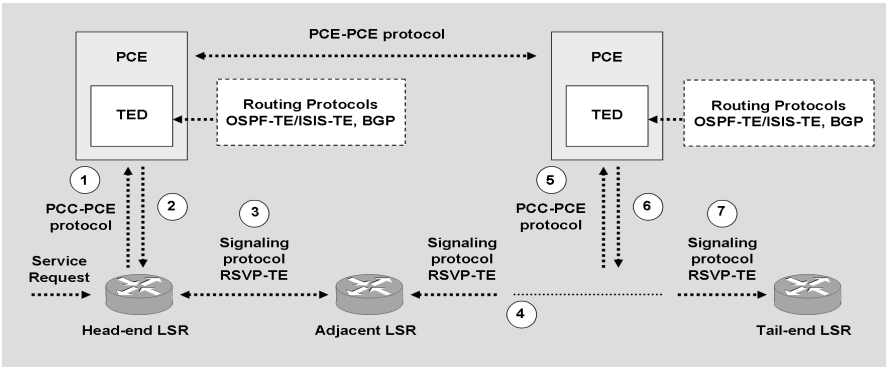


Figure 7.5: Request/Response messages and protocols involved.

network topology of downstream domains. The approach is simple since each PCE computes a piece of the LSP based on its knowledge of the state of resources within its AS, and the reachability information obtained from BGP. Unfortunately, the major drawback of computing paths by segments is that the resulting paths are likely to be far from optimal. For instance, it is a well-known fact that high quality paths are frequently uncorrelated with the routing choices made by BGP [55].

The issue that remains wide open is how to exploit the PCE-based model to compute high quality primary and backup LSPs across a small group of domains in a viable way, that is, without adversely affecting the scalability and the confidentiality features of the above approach. In the sequel, we explore the key challenges raised by this issue (a solution for it is proposed in Chapter 8).

7.4 Challenges to be faced

Splitting the computation of primary and backup inter-domain LSPs introduces a number of problems that need to be addressed in order to avoid coarse-grained solutions. In what follows, we examine the key challenges to be faced and discuss about the way to solve them.

7.4.1 TE information exchange model among domains

With the current PCE model each segment of an inter-domain LSP is derived from a very limited visibility of the state and topology of the network. As a result, no guarantee exists that the optimal QoS path (e.g. the shortest path) will be found. In fact, once a PCE has chosen the NH domain and established its segment of the LSP, there is no guarantee that a viable QoS path will be discovered through the NH domain. This is because no information apart from IP reachability is exchanged between domains. When this occurs, the NH ASBR signals back an error message indicating that its domain is unable to set up the next segment of the LSP (such error can occur either while computing the path segment or while signaling its establishment along the domain). When a PCE receives this error message, it iteratively tries other downstream domains until it succeeds or rejects the path request. This trial and error provisioning and signaling process is referred to as crankback [33].

An alternative approach is to work toward a TE information exchange model between domains. This model could be supported by the PCE-based architecture, so it could be applied to a rather small group of neighboring domains. In this framework, domains become capable of exchanging some highly aggregated topology and state information, which can be used to compute "entire" LSPs directly from the "source" PCE.

In order to preserve the confidentiality of ISPs and also keep the model scalable, domains never advertise their internal structure, but rather supply an Aggregated Representation (AR) to the outside world (see Chapter 8). Thus, a key aspect is to find an adequate AR that captures the available path diversity of QoS paths across a small group of domains. Certainly, a trade-off exists between the optimality of the resulting QoS paths and the size of the AR. Two different ARs based on the advertisement of the available disjoint paths between the border nodes of a domain can be found in [113] and [108].

The advantage of the AR is that it facilitates the computation of entire (primary and backup) shortest paths directly from the "source" PCE [113]. Since the source PCE only knows an AR of the whole network, the resulting paths are still a mix of strict and loose hops. The list of strict hops could be

the source node, the list of border nodes to be traversed across the different domains, and the destination node. Thus, approaches like this still need to rely on the ERO expansion, but with the advantage of increasing the number of strict hops conveyed in the signaling messages.

Issues like how TE information is to be distributed and updated need to be carefully investigated. One possibility could be to use the PCE protocol. However, [126] only proposes a request/response protocol for the computation of paths, making it inappropriate for this purpose. Sound alternatives are to propose extensions to the current specification of the PCE protocol, or to develop a new one facilitating the advertisement of TE information among a small group of PCEs.

7.4.2 The routing decision

Once each PCE knows an AR of the multi-domain network, a routing decision is required in order select the paths for the PCC requests. For example, in [108] the disjoint paths computed by a source PCE are both routed along the same chain of domains, since it is assumed that the AS-level path is known in advance (e.g. is pre-computed by BGP). The AR in this case is basically an abstraction of an AS-path. This routing approach has two major weaknesses. First, high quality paths (e.g. the shortest paths between the source and destination nodes) are not guaranteed to be discovered, since they might not belong to the pre-computed AS-path. Second, when several disjoint LSPs need to be established following the same AS-path, the utilization of network resources at the inter-domain level could be quite inefficient. Instead, if a source PCE is not constrained to route the LSPs along a given AS-path, then the shortest paths can be found and a more efficient use of the overall network resources can be achieved [113].

An alternative routing scheme is to avoid handling an AR of the small-sized multihomed network, but refine the selection of the NH domain and then repeatedly solve this problem on a per-domain basis. Two heuristics in this direction have been recently proposed in [99], but the focus there is on the selection of a single path. The resulting paths under these routing schemes are expected to be of higher quality than those that can be obtained with the current PCE-based approach. Still, these routing schemes cannot guarantee to find optimal QoS paths (e.g. the shortest paths) across domains. Another alternative that can be used as an interim solution (e.g. before the deployment of the PCEs) was proposed in [109]. This proposal exploits the multi-connectivity between peering ASs in order to find disjoint LSPs along a chain of domains.

Overall, the key issue is that the resulting paths from the current PCE

routing scheme (see Figs. 7.4 and 7.5) are far from optimal, so alternative routing strategies like [113, 108, 99] deserve to be investigated.

7.4.3 Strategy for the computation of restoration paths

The issue that arises is whether to compute the primary and restoration paths at the same time, or one after the other. The latter case is subject to the well-known trap topology problem [108], so network resources can be consumed more efficiently when both paths are computed simultaneously. Accordingly, the heuristic algorithm controlling the decisions made by a PCE should be able to compute disjoint paths at the same time.

7.4.4 Fast restoration after a failure

With local restorations, each AS can potentially protect its corresponding segment of a path. However, fully relying on this approach means that each domain needs to trust the restoration decisions made by downstream domains, which might not be acceptable for some ISPs as well as for some mission-critical applications. Indeed, after a distant failure in an inter-domain path, the source node has neither guarantee that the path will be restored nor that the restored one will actually comply with the QoS constraints. Thus, pre-computed restoration paths with prompt failure detection and fast restoration from the source LSR become necessary in some cases.

Clearly, some applications can be protected by means of local protections, i.e., on a per-segment basis, while others will need novel mechanisms at the application level to promptly detect a failure and switch to a backup path.

Overall, the PCE-based model facilitates the provisioning of primary and backup QoS LSPs across domains. The current proposals for finding such paths are based on a coarse selection of the paths from the source domain, and then rely on the ERO expansion technique within the subsequent domains traversed. The strengths of this approach are its scalability and the preservation of the confidentiality of ISPs networks. The main weakness is that the resulting paths are far from optimal.

In the next chapter we describe a solution that exploits the PCE-based model in order to aggregate and distribute enriched TE information among domains. This approach allows to compute “entire optimal” LSPs directly from the source PCE.

Chapter 8

Reliable Routing in IP/MPLS Multi-Domain Networks

In this chapter we focus on the problem of establishing two disjoint QoS paths across multiple domains. This problem, referred to as *Problem 2DP*, is considered in the context of the routing model inspired by the recently proposed PCE-based architecture [34].

The goal here is to describe a distributed routing model with provable performance guarantees that we developed in a recent work led by A. Sprintson [113]. In this routing model, each PCE is able to compute "entire optimal" primary and backup QoS paths to any destination. This approach allows to overcome the limitations of coarse-grained solutions such as those that arise by iteratively solving Problem 2DP on a per-domain basis [33]. Another major advantage of this strategy is that it avoids the well-known *trap topology* problems [108].

To achieve scalability and due to security and administrative considerations, routing domains do not advertise their internal structure, but rather supply an *Aggregated Representation* (AR) to the outside world. Accordingly, a key aspect in the design of a distributed multi-domain routing model is to find an adequate AR that captures the availability of diverse QoS paths across multiple domains. However, there is an inherent trade-off between the accuracy of the representation and the size of the data structures that need to be handled and advertised by the routers in the network. Our approach is to consider a setting in which a reduced set of neighboring domains are willing to extend the reachability of IP/MPLS LSPs across their boundaries. This enables each domain to provide an accurate representation of its traversal characteristics, which, in turn, enables finding optimal disjoint paths across the network. This approach is consistent with that adopted by the IETF PCE WG; the WG has openly stated that its efforts will focus on the application of the PCE-based model within a single domain or within a small group of neighboring domains, but it is not the intention of the WG to apply this model to the greater Internet [97].

The AR for a multi-domain network developed in [113], is small enough to

minimize the link-state overhead, and, at the same time, is sufficiently accurate, so that the PCEs can optimally find disjoint QoS paths across multiple domains. This solution guarantees that the confidentiality and administrative limits are respected between domains (e.g., neither the internal topology nor the full IGP state of the domains can be inferred from their ARs).

The problem of optimally finding two disjoint paths is considered both in a general setting, as well as subject to the usual export policies imposed by customer-provider and peer relationships between routing domains (see Section 2.3). In particular, the export policies determine the inter-domain links that the source PCE can use while computing paths for any source-destination pair. It turns out that the standard approach of representing a multi-domain network by a graph is inadequate for finding disjoint paths subject to the export policies. However, A. Sprintson has shown that the export policies can be efficiently represented by employing the concept of the *line graph* [113]. Based on this, our distributed algorithm can be easily extended for finding optimal disjoint paths that satisfy the export policy constraints.

For clarity of exposition, we focus on finding link-disjoint paths. Our results can be easily extended to finding node-disjoint paths by using the standard node splitting technique (see, e.g., [118]).

8.1 The network model

We begin with a definition of a general communication network. A network is represented by a directed graph $G(V, E)$, where V is the set of nodes and E is the set of links. Each link $e \in E$ is assigned a positive *weight* w_e, whose significance depends on the type of considered QoS requirement. For example, when the QoS requirement is an upper bound on the end-to-end delay, the link weight is its delay. Here we focus on additive weight metrics, i.e., the weight $W(P)$ of a path P is defined as the sum of the weights of its links, i.e., $W(P) = \sum_{e \in P} w_e$.

The goal of QoS routing is to find the best path that satisfies a QoS constraint. In this work, we accomplish this goal by finding a minimum-weight path between the source and the destination nodes. Clearly, such path has the best performance with respect to the QoS requirement that is captured by the link weight metric.

8.1.1 Extending the PCE-based model

We denote by $D_1, \ldots, D_k$ the set of routing domains in the network. Each routing domain D_i is a subgraph of the underlying network G. We assume that routing domains are mutually node-disjoint. The routing domains that include the source and destination nodes, s and t are referred to as D^s and D^t, respectively. A link that connects two nodes in the same domain is referred to as an intra-domain link. All other links connect different domains and are referred to as inter-domain links. We denote by E^{inter} the set of the inter-domain links in the network. A node v which is incident to an inter-domain link is referred to as a border node. The set of border nodes of a routing domain D_i is denoted by B_i.

In large communication networks, distributing the full link state information to every node in the network is not possible due to scalability problems. With topology aggregation, subnetworks, or routing domains, can limit the amount of link state information advertised throughout the network [124]. Our approach is that routing domains supply a short summary of the available (disjoint) paths that connect the border nodes of the domain. The efficiency of this approach stems from the fact that while the routing domains tend to be large, the number of border nodes in each domain is typically small.

We denote by A_i the AR of the routing domain i. The AR captures the transitional characteristics of the network and can be implemented by a (small) graph that includes several arrays summarizing the available routing paths between the border nodes of the routing domain.

In order to distribute the ARs of routing domains throughout the network we take advantage of the architecture recently drafted by the IETF PCE WG. Our PCE-based routing model utilizes a decoupled control plane for both the computation of the 2DP and the advertisement of routing information. This decoupling is two-fold. On the one hand, the PCEs are detached from the MPLS switch/routers forwarding the traffic. On the other hand, the aggregated topology, reachability, and path state information needed to compute the routing paths are decoupled from BGP and advertised directly between the PCEs [139]. This approach has two major advantages. First, it overcomes some of the most important limitations imposed by BGP [138]. For example, it allows to advertise multiple routes per destination prefix, and to convey path state information in the routing advertisements, which cannot be done at present with BGP-4. Overcoming these limitations is essential for the optimal computation of disjoint paths between multiple routing domains. Second, this approach can be incrementally deployed since it can coexist with the legacy IP IGP/BGP routed traffic.

The information available at the source PCE includes, the source domain D^s, a set of inter-domain links E^{inter}, and the ARs of the transit and destination domains.

Problem definition

In this work we focus on finding two link-disjoint paths in a multi-domain network with topology aggregation. The first path, referred to as a *primary* path, is used during the normal operation of the network. Upon a failure of a link in the primary path, the traffic is shifted to a *backup* path. In order to satisfy the required QoS constraint, we need to minimize the weight of both primary and backup paths. Accordingly, we consider the problem of finding link-disjoint paths of minimum total weight.

Problem 2DP (2 Link Disjoint Paths): Given a source node s and a destination node t, find two link-disjoint (s,t)-paths P_1 and P_2 of minimum total weight $W(P_1) + W(P_2)$.

We can use the path with minimum weight as a primary path and the second one as a backup path. A relevant problem is to find two paths P_1 and P_2 that minimize $\max\{W(P_1), W(P_2)\}$. The solution to this problem can achieve a better balance between the delay of the primary and backup path. However, this problem is $\mathcal{NP}$-hard [71].

Problem 2DP is a well studied problem. The standard algorithm used for solving this problem is due to Suurballe and Tarjan [119]. However, the existing algorithms were designed for the case in which the full topology is known to every node in the network. Accordingly, the goal of this study is to provide an efficient solution for the case in which only the aggregated representation of the network is known.

8.2 Related work

The problem of finding primary and backup paths subject to QoS constraints in the context of IP/MPLS networks has been widely studied at the intra-domain level. With the advent of the PCE-based architecture, a few recent works have started to extend the study of this problem to LSPs spanning multiple domains. In the current IGP/BGP routing context, a major issue is that the PCE in the source domain has to compute inter-domain LSPs based on a very limited visibility of the topology and state of the network, yielding solutions that are far from optimal. To cope with this, enriched topological

and path state information needs to be aggregated and available at the PCE in the source domain [139].

In [108] the authors compare the performance of some recently proposed distributed schemes for disjoint path computation of inter-domain LSPs. They assume that the AS-level path was previously computed by BGP at the source domain and that both disjoint paths belong to the same "chain" of domains. This approach has two major limitations. First, solving problem 2DP restricted to the AS-path selected by BGP will frequently return paths that are far from optimal. This is because BGP does not offer any guarantee about the quality of the chosen AS-path. Second, when several disjoint LSPs need to be established following the same (or part of the same) AS-path, crankback [33] or even blocking might occur, even though the paths could have been established along the alternative AS-paths available at the source domain.

In this thesis we study a PCE-based architecture that is completely decoupled from the BGP protocol. With this approach, the PCE at the source domain is not compelled to choose both paths along the same chain of domains. This allows the domains to use their multi-homed networks more efficiently. Once we extend the computation of the paths to an expanded AS topology, i.e., not restricting our study to a chain of domains, we need to consider the export policies between domains. This, however, introduces a major challenge. Whereas the chain of domains can be aggregated and represented as a directed graph, this cannot be done in the presence of the export policies. To solve this problem [113] introduces an AR of the expanded topology using line graphs.

In [99] the authors propose two heuristics so that the PCEs can solve the problem of finding inter-domain LSPs with low end-to-end delay. However, this work addresses the computation of only a single path (without a disjoint counterpart). In addition, the availability of inter-domain paths is inferred directly from the BGP routing information. Accordingly, the authors do not need to address the issue of finding an AR that captures path diversity and the internal structure of domains.

Overall, to the best of our knowledge, [113] is the first study optimally solving Problem 2DP in an expanded multi-domain IP/MPLS environment, subject to the common export policies. The contributions in [113] – which shall be detailed in this chapter – can be summarized as follows:

- We propose an accurate AR that captures the path diversity and the internal link state of each domain.

- We introduce a distributed routing algorithm that exploits an AR of the

multi-domain network in order to find an optimal pair of link-disjoint paths between the source and the destination in an efficient manner.

- We provide an efficient method for finding link-disjoint paths subject to the common export policies imposed by customer-provider and peer relationships between routing domains.

8.3 Link-disjoint paths in the general case

In this section we describe a distributed algorithm for finding two link-disjoint paths in a multi-domain network with topology aggregation.

The distributed algorithm for path computation consists of the three following steps. In the first step, each routing domain D_i computes its AR A_i. This computation is performed by the PCE of the domain. In the second step, the AR A_i of each domain D_i is distributed throughout the network. In the third step, the PCE in the source domain uses the assembled representation of the network for computing two disjoint paths between the source and the destination nodes.

The rest of this section is structured as follows. In section 8.3.1 we present our AR. In Section 8.3.2 we describe an algorithm for computing the AR of a domain. Then, in Section 8.3.3 we describe an algorithm for computing disjoint paths at the source PCE. Finally, in Section 8.3.4 we describe an algorithm for establishing two disjoint (s, t)-paths throughout the network.

8.3.1 Aggregated representation

We begin by the description of the AR. The purpose of the AR is to summarize the traversal properties of each routing domain in a way that allows the source PCE to select two disjoint paths of minimum weight.

Aggregation scheme for minimum distances

The problem of finding a suitable AR that enables efficient computation of the minimum weight paths across the network is well studied in the literature. The natural representation of a routing domain D_i is an array that stores, for each pair of border nodes b_j and b_l of D_i, the minimum weight of a path between b_j and b_l. This representation allows the source node to find optimal paths and has the space complexity of $\Theta(|B_i|^2)$.

This representation, however, cannot be used for finding two disjoint paths across the network. To illustrate this point, consider the routing domain depicted in Fig. 8.1. The domain has four border nodes $b_1, \ldots, b_4$.

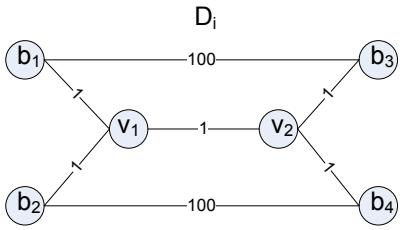


Figure 8.1: An example of a routing domain.

The numbers show the weights of the edges. In this domain, the minimum weight of the path between b_1 and b_3 and between b_2 and b_4 is equal to 3. However, the minimum weight of two disjoint paths, one between b_1 and b_3 and the second between b_2 and b_4 is equal to 103. This shows that additional information regarding the disjoint paths that run through the domain must be included in the aggregated representation.

Aggregation scheme based on the minimum weight of disjoint paths

A possible solution would be to keep, for each routing domain D_i and for each two pairs (b_j, b_l) and (b_x, b_y) of D_i, the minimum weight of two link-disjoint paths that connect b_j and b_x to b_l and b_y.

In addition, we need to keep, for each routing domain D_i and for every pair (b_j, b_l) of border nodes of D_i, the minimum weight of a path between b_j and b_l. This method provides complete information about the traversal characteristics of the routing domain, under the assumption that each path enters the routing domain at most once. The main drawback of this approach is that the aggregated information does not allow the source PCE to find two disjoint paths between s and t in an efficient way. Indeed, the most effective method for finding two disjoint (s,t)-paths includes two steps, the first step finds a shortest path (s,t)-path P' and the second step finds an augmenting (s,t)-path P'' of P'. The augmenting path P'' may use links of P' in the reverse direction, which allows to avoid the trap topology problems [108]. This method is employed by the standard disjoint path algorithm due to Suurballe and Tarjan [119], described in detail in the next section. However, the aggregation scheme based on the minimum weight of disjoint paths inside a domain does not allow to compute the "augmenting" inter-domain path in an efficient way. In what follows, we present an alternative aggregated representation that addresses this problem.

Aggregation scheme based on the disjoint paths algorithm

Let $D_i(V_i, E_i)$ be a routing domain and let B_i be the set of border nodes on V_i. In this section we present the AR A_i of D_i. The main goal in the design of the AR is to allow the source PCE to find the minimum weight of disjoint paths in an efficient way. We begin by presenting the disjoint path algorithm due to Suurballe and Tarjan [119]. The algorithm receives as an input a graph $G(V, E)$, a source node s, and the destination node t. The algorithm performs the following steps:

1. Find a shortest path P' between s and t in G;
2. Reverse all links in P' and negate their weight;
3. Find an augmenting shortest path P'' in the resulting graph $\hat{G}$;
4. Remove links that appear in P' and P'' in opposite directions;
5. From the remaining links of P' and P'', form two disjoint (s, t)-paths $\hat{P}_1$ and $\hat{P}_2$.

The idea of our scheme is to allow the source PCE to compute two disjoint paths in the aggregated environment in a similar way as if the entire network topology were known. To that end, A_i includes two components. The first component allows the source PCE to find a shortest path P_1 between s and t, while the second component allows the source PCE to find the second path P_2. The paths P_1 and P_2 correspond to the paths P' and P'', used by the algorithm due to [119].

In particular, the first component of A_i includes array M_i' that contains, for each two border nodes b_j and b_l of D_i, the minimum weight of a path between b_j and b_l. The second component of A_i includes a set of $|B_i|(|B_i|-1)$ arrays $\{M_i^{j,l} \ | b_j \in B_i, b_l \in B_i, b_j \neq b_l\}$, each array containing $|B_i|(|B_i|-1)$ elements. In particular, array $M_i^{j,l}$ contains, for any two border nodes b_x and b_y of D_i, the minimum weight of a path between b_x and b_y in $D_i^{j,l}$, where $D_i^{j,l}$ is a graph formed from D_i by inverting links that belong to a minimum weight path between b_j and b_l and negating their weights.

Figure 8.2 graphically presents the aggregated representation of the routing domain shown in Fig. 8.1. The representation we present is based on the following assumption.

Assumption 8.3.1. *A minimum weight path between a source node s and the destination node t traverses each routing domain D_i at most once.*

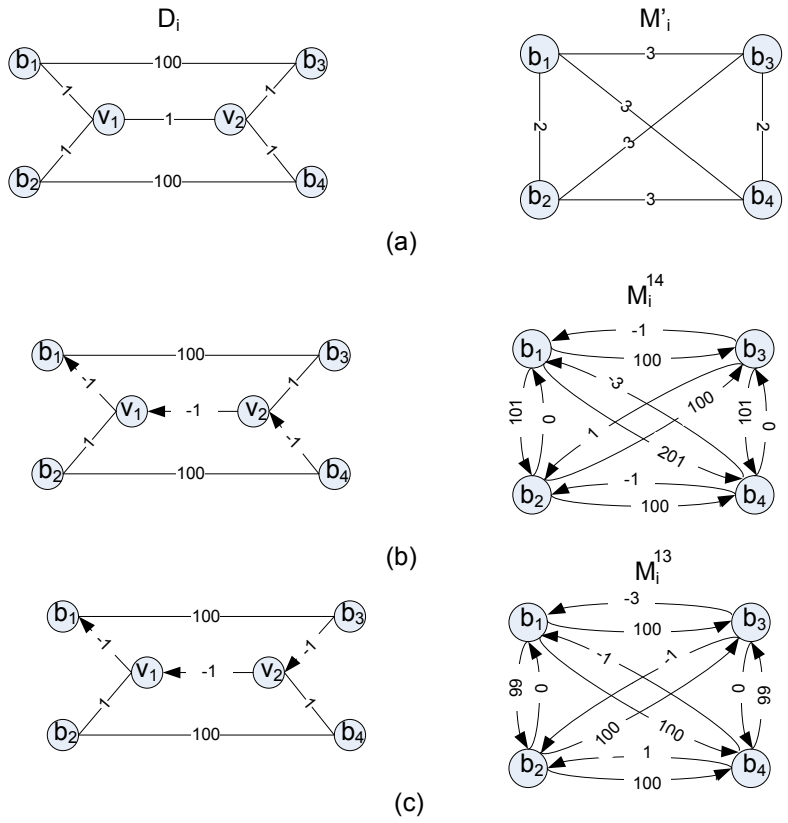


Figure 8.2: Aggregated representation of a routing domain.
(a) Array M_i' (b) The auxiliary graph $M_i^{1,4}$ (left) and array $M_i^{1,4}$ (right).
(c) The auxiliary graph $M_i^{1,3}$ (left) and array $M_i^{1,3}$ (right).

This fact significantly simplifies the construction of an aggregated representation. Our methods can be extended to deal with settings in which this assumption does not hold.

8.3.2 First step–Computing the AR

The AR A_i can be efficiently computed using Algorithm FindAR due to A. Sprintson [113], which appears in next page. The algorithm computes, for each pair of border nodes b_j, b_l of D_i, a shortest path $P_i^{j,l}$ between b_j and b_l in D_i and stores the result in array M_i'. Then, the algorithm reverses all links of $P_i^{j,l}$, negates their weights, and computes a minimum weight path between any pair of border nodes in the resulting graph. The minimum weights of these paths are stored in the array $M_i^{j,l}$. Since the resulting graph may con-

Algorithm 3 FindAR(D_i, B_i)

Input: D_i - a routing domain
B_i - the set of border nodes of D_i

Output: $A_i = \{M_i'\} \cup \{M_i^{j,l} \ / \ b_j \in B_i \ , b_l \in B_i\}$ of D_i

1: **for** each two border nodes b_j and b_l of D_i **do**
2: Compute a shortest path $P_i^{j,l}$ between b_j and b_l in D_i
3: $M_i'(j,l) \leftarrow W(P_i^{j,l})$
4: Construct an auxiliary graph $D_i^{j,l}$ formed from D_i by reversing all links of $P_i^{j,l}$ and negating their weights
5: **for** each two border nodes b_x and b_y of D_i **do**
6: Compute a shortest path $P_i^{j,l}(x,y)$ between b_x and b_y in $D_i^{j,l}$
7: $M_i^{j,l}(x,y) \leftarrow W(P_i^{j,l}(x,y))$
8: **end for**
9: **end for**

tain negative weights, we use a modification of the Dijkstra's algorithm due to Bhandari [15]. The computational complexity of the modified algorithm is identical to that of the original Dijkstra's algorithm.

Finding a shortest path between any pair of border nodes requires $|B_i|$ invocations of the shortest path algorithm. Thus, computing the AR A_i of D_i requires $O(|B_i|^3)$ invocations of the shortest path algorithm. Therefore, the computational complexity of computing the AR is $O(|B_i|^3(|V_i| \log |V_i| + |E_i|))$. The size of the aggregated representation is $O(|B_i|^4)$.

To derive a practical estimation of the size of this AR, let us compare this latter against the number of active entries in the BGP Forwarding Information Base (FIB) of the border routers in a Tier-1 ISP. At present, these border routers have around 2.3x10^5 active entries in their BGP FIB [21], and this scale does not represent an issue for the routers. In our case, an AR of 22 border routers on average per-domain (i.e. approximately $\sqrt[4]{2.3\text{x}10^5}$) represents the same load as operational routers have nowadays in a Tier-1 ISP. It is worth recalling that our proposals apply to a reduced set of neighboring domains, and that they can be incrementally deployed. In this scenario, an average of 22 IP/MPLS enabled border routers per-domain offers significant flexibility from a practical viewpoint.

8.3.3 Second step–Minimum weight of shortest paths

We assume that the source PCE has a detailed topology of the source routing domain, and in addition, the ARs of the transit and destination routing domains. The source PCE uses this information in order to construct a high-level description of two disjoint paths that connect s and t.

We note that while the AR A_i of a transit domain D_i captures the path diversity and link-state information of D_i, the AR of D^t captures the same properties, but for the paths between the border nodes of D^t and the destination t. Given that the AR of D^t follows the same principle as that of any transit domain, without loss of generality, in our model the destination t is considered as a border node of the routing domain D^t. This is motivated by the fact, that in order to find an optimal pair of link-disjoint paths, the source PCE needs some information about the paths between the border nodes of D^t and the destination t.

The operations performed by the source PCE are summarized by Algorithm Find2DP (due to A. Sprintson [113]), which appears in next page. Algorithm Find2DP begins by constructing an auxiliary graph $G'(V', E')$ that includes, for each domain D_i of G, the complete graph spanned by the border nodes of D_i. In addition, G' includes the source domain D^s and the set of inter-domain links E^{inter}. The purpose of the auxiliary graph is to summarize the network information available at the source PCE.

Next, the source PCE computes the shortest path P_1 between s and t. This is accomplished by assigning for each link (b_j, b_l) that connects two border nodes of the same domain D_i, the minimum weight of a path between b_j and b_l and finding a shortest path between s and t in the resulting graph. The minimum weights of the shortest paths that run through domain D_i are available through array M'_i.

Finally, the source PCE computes the second (s, t)-path P_2. To that end, for each domain D_i traversed by P_1 (i.e., P_1 contains a link that connects border nodes of D_i) we perform the following operations. Let (b_j, b_l) be a link in P_1 that connects border nodes of D_i and let $\hat{D}_i$ be the complete graph spanned by the border nodes of D_i. Then, we set the weights of the links of the subgraph $\hat{D}_i$ of G' according to array $M_i^{j,l}$. The path P_2 is found by applying the shortest path algorithm on the resulting graph.

The computational complexity of Algorithm Find2DP is $O(|V'| \log |V'| + |E'|)$, where V' is the set of nodes and E' is the set of links of the auxiliary graph $G'(V', E')$. Again, since the auxiliary graph contains negative weights, we use the algorithm due to Bhandari [15] for finding shortest paths in G'. The set V' includes all nodes in the source routing domain and the border nodes of all transit domains and the destination domain. The set E' includes

Algorithm 4 Find2DP(E^{inter}, $\{A_i\}$)

Input: E^{inter} - a set of the inter-domain links
For each routing domain D_i
$A_i = \{M_i'\} \cup \{M_i^{j,l} \ / \ b_j \in B_i \ , b_l \in B_i\}$ of D_i , the aggregated representation of D_i

Output: An auxiliary network $G'(V', E')$ and two paths P_1 and P_2 in G'

1: $V' \leftarrow V(D^s) \cup \{B_i \mid D_i \in G\}$
2: $E' \leftarrow E(D^s) \cup E^{inter}$
3: **for** each routing domain D_i of G **do**
4: **for** each two border nodes b_j and b_l of D_i **do**
5: $E' \leftarrow E' \cup (b_j, b_l)$
6: $w_{(b_j,b_l)} \leftarrow M_i'(j,l)$
7: **end for**
8: **end for**
9: Find a shortest path P_1 between s and t in $G'(V', E')$
10: Reverse all inter-domain links and links that belong to D^s in P_1 and negate their weight
11: **for** each routing domain D_i of G except D^s **do**
12: **if** P_1 contains a link (b_j, b_l) that connects border nodes of D_i **then**
13: **for** each two border nodes b_x and b_y of D_i **do**
14: $w_{(b_x,b_y)} \leftarrow M_i^{j,l}(x,y)$
15: **end for**
16: **end if**
17: **end for**
18: Find a shortest path P_2 between s and t in $G'(V', E')$

all links in the source domain, the set of inter-domain links and, in addition, a link between any two border nodes of the same domain.

8.3.4 Third step–establishing QoS paths

In the third step, the source PCE sends the paths P_1 and P_2 to every routing domain D_i traversed by these paths. At each domain, the PCE is responsible for establishing the portions of the disjoint paths that run through these domains. We consider the following cases.

1. Domain D_i is traversed by path P_1 and is not traversed by P_2. In this case, let (b_j, b_l) be the link in P_1 that connects the border nodes of D_i. Then, link (b_j, b_l) is substituted by the path $P_i^{j,l}$ computed at Line 2 of Algorithm FindAR.

2. Domain D_i is traversed by path P_2 and is not traversed by P_1. In this case, each link $(b_j, b_l) \in P_2$ that connects the border nodes of D_i is substituted by the path $P_i^{j,l}$ computed at Line 2 of Algorithm FindAR.

3. Domain D_i is traversed by both paths P_1 and P_2. Let (b_j, b_l) be the link in P_1 that connects the border nodes of D_i. Then, we perform the following operations. First, link (b_j, b_l) is substituted by the path $P_i^{j,l}$ computed at Line 2 of Algorithm FindAR. Second, each link $(b_x, b_y) \in P_2$ that connects the border nodes of D_i is substituted by the path $P_i^{j,l}(x, y)$ computed at Line 6 of Algorithm FindAR. Finally, all links of D_i that appear in $P_i^{j,l}$ and $P_i^{j,l}(x, y)$ in opposite directions are omitted from both $P_i^{j,l}$ and $P_i^{j,l}(x, y)$.

8.3.5 Illustrative Example

Figure 8.3 presents an illustrative example of our algorithm. The underlying communication network, depicted in Fig. 8.3(a), contains source domain D^s and two transit domains, D_1 and D_2. Figure 8.3(b) depicts the auxiliary network G' constructed by Algorithm Find2DP with link weights assigned according to the values of arrays M_1' and M_2'. This auxiliary network is used by the source PCE to compute the shortest path between the source and the destination nodes. The shortest path is marked in Fig. 8.3(b) by the bold lines and includes border nodes b_3 and b_6 of routing domain D_1 and nodes b_8 and b_9 of routing domain D_2. The weight of the P_1 is 11. Next, the source PCE turns to compute path P_2. To that end, the same communication network is used, but the weights are assigned according to arrays $M_1^{3,6}$ (for D_1) and $M_2^{8,9}$ (for D_2, see Fig. 8.3(c)). The shortest path in this network is marked by the bold lines and includes nodes b_4 and b_5 of routing domain D_1 and nodes b_7 and b_{10} of routing domain D_2. The weight of path P_2 is 205. Finally, the source PCE sends the two disjoint paths P_1 and P_2 to the PCEs of the routing domains D_1 and D_2. The PCE of the routing domain D_1 substitutes the two links (b_3, b_6) and (b_4, b_5) of D_1 by two disjoint paths $\{b_3, b_5\}$ and $\{b_4, v_1, v_2, b_6\}$, while the two links (b_8, b_9) and (b_7, b_{10}) of D_2 are substituted by two paths $\{b_7, b_9\}$ and $\{b_8, v_3, v_4, b_{10}\}$. The two disjoint paths in the original network are depicted in Fig. 8.3(d).

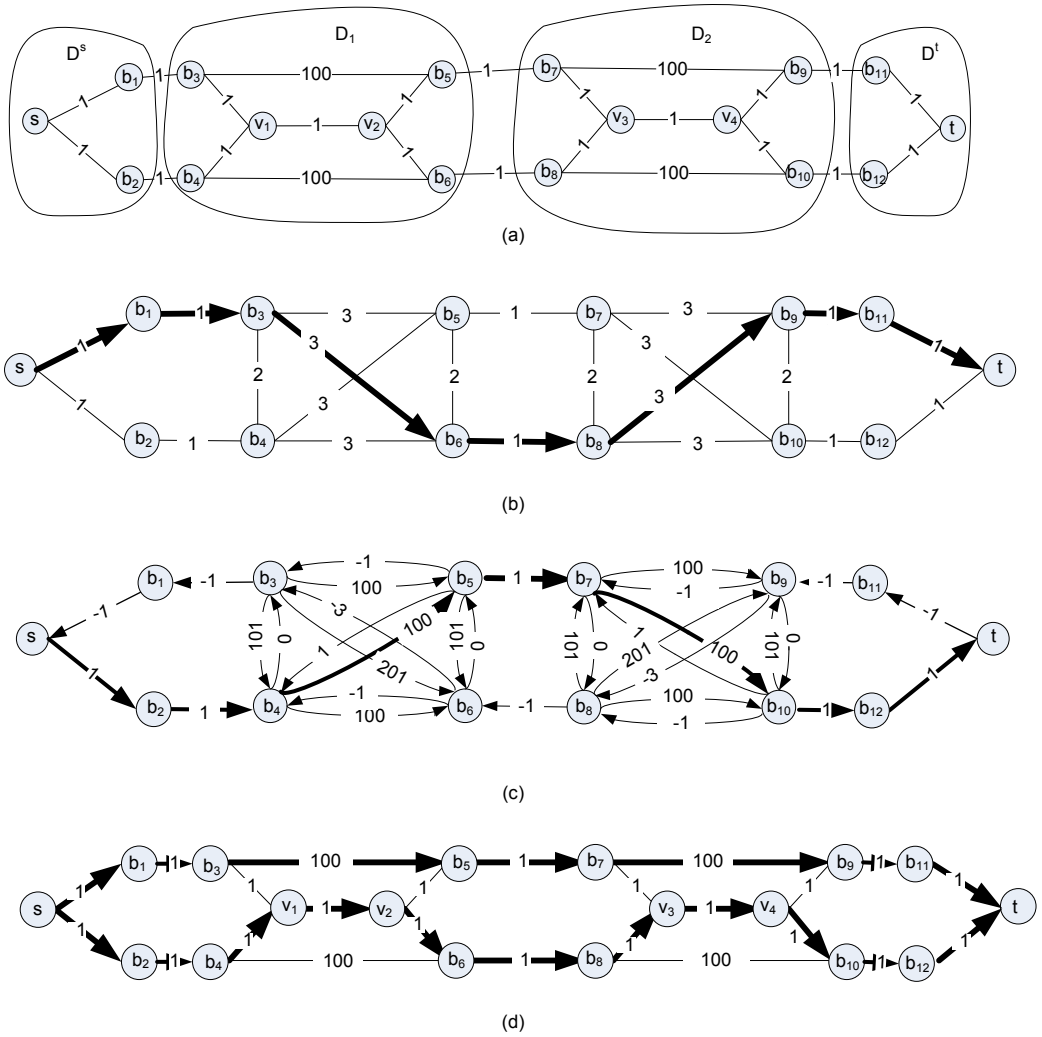


Figure 8.3: An illustrative example.
(a) The underlying communication network with two transit domains D_1 and D_2. (b) The auxiliary network $G'(V', E')$ with weights assigned according to arrays M_1' and M_2'. (Two directed links with identical weights are represented by a single undirected link) (c) The auxiliary network $G'(V', E')$ with weights assigned according to arrays $M_1^{3,6}$ and $M_2^{8,9}$. (d) Two disjoint paths in the underlying network.

8.3.6 Correctness proof

In the sequel, a proof of the optimality of Algorithm Find2DP proposed by A. Sprintson in [113] is provided.

Theorem 8.3.1. *If Assumption 8.3.1 holds, then Algorithm Find2DP finds two disjoint paths between s and t of minimal total weight.*

Proof: Suppose that the full topology of the communication network G is known. In this case, we can apply the algorithm due to [119] (as described in Section 8.3.1) to find two disjoint paths of minimal weight. Let P' and P'' be the paths identified in Lines 1 and 3 of this algorithm, respectively. The correctness of the algorithm implies that $W(P')+W(P'')$ is equal to the minimum weight of two paths between s and t.

Next, we show that for paths P_1 and P_2, identified by the Algorithm Find2DP, it holds that $W(P_1) \leq W(P')$ and $W(P_2) \leq W(P'')$. This is sufficient to prove the correctness of the algorithm. Indeed, in Step 3 (presented in Section 8.3.4) of the algorithm we use P_1 and P_2 to establish two link-disjoint (s,t)-paths that will be expanded by the traversed domains. It is easy to verify that the total weight of the resulting paths is equal to the total weight of P_1 and P_2.

We proceed to show that $W(P_1) \leq W(P')$. We note that path P' can be divided into subpaths $P'_1, \ldots, P'_h$ such that P'_1 connects s to a border node of D^s, P'_h connects a border node of D^t to t and for $2 \leq i \leq h-1$, P'_i either includes an inter-domain link or a link that connects two border nodes of a routing domain. We also note that the auxiliary network $G'(V', E')$ includes all links of the subpath P'_1 and also all subpaths P'_i that include inter-domain links. Further, all links of these subpaths have the same weight in G' as in the original network. For each subpath P'_i of P' that connects border nodes b_j and b_l of a routing domain D_x, the auxiliary network $G'(V', E')$ contains a link whose weight is less than or equal to $W(P'_i)$. Since P_1 is a minimum weight path in $G'(V', E')$, it follows that $W(P_1) \leq W(P')$.

Finally, we show that $W(P_2) \leq W(P'')$. Let $\hat{G}$ be the resulting graph after executing Line 2 of the algorithm due to [119] (as presented in Section 8.3.4). We note that path P'' can be divided into subpaths $P''_1, \ldots, P''_h$ such that P''_1 connects s to a border node of D^s, P''_h connects a border node of D^t to t and for $2 \leq i \leq h-1$, P'_i either includes an inter-domain link or connects two border nodes of the routing domain. We also note that the auxiliary network $G'(V', E')$ includes all links of the subpaths that traverse the source routing domains and all subpaths P''_i that include inter-domain links and these links have the same weight as in $\hat{G}$. For each subpath P''_i of P'' that connects border nodes b_j and b_l of a routing domain D_x, the

auxiliary network $G'(V', E')$ contains a link whose weight is less than or equal to $W(P_i'')$. Since P_2 is a minimum weight path in $G'(V', E')$, it follows that $W(P_2) \leq W(P'')$.

■

8.4 Link-disjoint paths under the export policies

In this section we discuss the problem of finding two disjoint paths in the network in the presence of export policies. The main challenge posed by the export policies is that the availability of the link for a particular connection depends on the previous hop. As a result, the standard representation of the network in the form of a graph is no longer adequate for routing purposes. For example, consider the multi-domain network depicted on Fig. 8.4(a). In this network, D_1 is a customer of both D_2 and D_3; D_7 is a customer of D_5 and D_6; D_4 is a provider of D_3 and D_6; D_2 is a peer of D_4, and D_4 is a peer of D_5. The export policies specified in Table 2.1 allow the following paths between D_1 and D_7: (a) $D_1 \rightarrow D_2 \rightarrow D_4 \rightarrow D_6 \rightarrow D_7$; (b) $D_1 \rightarrow D_3 \rightarrow D_4 \rightarrow D_5 \rightarrow D_7$; (c) $D_1 \rightarrow D_3 \rightarrow D_4 \rightarrow D_6 \rightarrow D_7$. Note the every link of the network is included in one of these paths. Thus, pruning a link from the network will result in omitting one of the feasible paths from the network. However, the path $D_1 \rightarrow D_2 \rightarrow D_4 \rightarrow D_5 \rightarrow D_7$ that belongs to the network is not allowed by the export policies. We conclude that the graph that depicts the connectivity among multiple domains, such as the one shown in Fig. 8.4(a), is not adequate for computing optimal paths subject to the export policies.

8.4.1 Line graphs

In order to efficiently find paths subject to export policies, A. Sprintson proposed in [113] to use the notion of the *line graph*. The rationale for this is that line graphs are able to capture the transit properties between the ingress and egress links of domains.

Definition 1 (Line Graph): Let $G(V, E)$ be a communication network, $D_1, \ldots, D_k$ be the set of routing domains in G, where $D^s = D_1$ is the source domain and $D^t = D_k$ be the destination domain, and E^{inter} be the set of inter-domain links. Then, the line graph $\hat{G}(\hat{V}, \hat{E})$ of G is a graph constructed as follows:

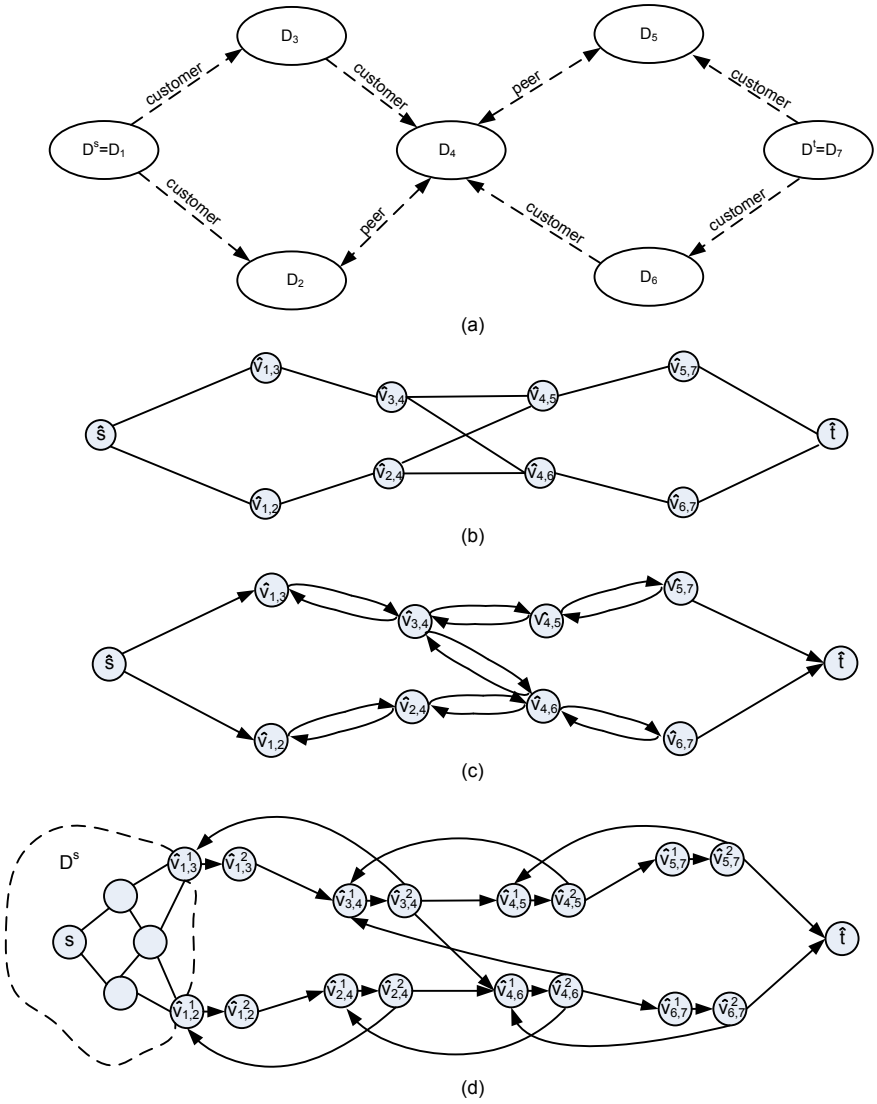


Figure 8.4: Line graphs.
(a) Multi-domain network. The lines represent the connectivity between domains. For example, D_1 is connected to D_3, and D_4 is connected to D_5. The directions of the links show the relationship between the domains. For example, D_1 is a customer of D_3, while D_4 is a peer of D_5. (b) The corresponding line graph with two special vertices $\hat{s}$ and $\hat{t}$. (c) Graph $\hat{\hat{G}}_1$. (d) Graph $\hat{\hat{G}}_2$.

1. For each inter-domain link $e_i \in E^{inter}$ in G add a corresponding node $\hat{v}_i$ to $\hat{V}$.

2. For each routing domain $D_i, 1 \leq i \leq k$, and for each two inter-domain links e_j and e_l incident to D_i in G:

   – Add a link $(\hat{v}_j, \hat{v}_l)$, where $\hat{v}_j$ and $\hat{v}_l$ are nodes in $\hat{G}$ that correspond to e_j and e_l, respectively.

3. Add special nodes $\hat{s}$ and $\hat{t}$.

4. For each inter-domain link e_i incident to the source routing domain D^s add a link between $\hat{s}$ and the node $\hat{v}_i$ in $\hat{G}$ that corresponds to e_i.

5. For each inter-domain link e_i incident to the destination routing domain D^t add a link between the node $\hat{v}_i$ in $\hat{G}$ that corresponds to e_i and $\hat{t}$.

Figure 8.4(b) depicts the line graph of the multi-domain network that appears in Fig. 8.4(a). In this figure, the node corresponding to a link between routing domains D_i and D_j in G is denoted by $\hat{v}_{i,j}$.

Let P be an (s,t)-path in G and let $D^s = D_1, D_2, \ldots, D_h = D^t$ be the set of routing domains traversed by P. We say that path $\hat{P} = \{\hat{s}, \hat{v}_{1,2}, \hat{v}_{2,3}, \ldots, \hat{v}_{h-1,h}, \hat{t}\}$ in $\hat{G}$ *corresponds* to P. The following proposition follows from the construction of the line graph $\hat{G}$.

Proposition 8.4.1. *Let P_1 and P_2 be two link-disjoint paths in G. Then, the two corresponding paths $\hat{P}_1$ and $\hat{P}_2$ in $\hat{G}$ are node-disjoint.*

For example, suppose that P_1 traverses routing domains D^s, D_3, D_4, D_5, D^t and P_2 traverses routing domains D^s, D_2, D_4, D_6, D^t. Then, the two corresponding paths in $\hat{G}$, $\hat{P}_1 = \{\hat{s}, \hat{v}_{1,3}, \hat{v}_{3,4}, \hat{v}_{4,5}, \hat{v}_{5,7}, \hat{t}\}$ and $\hat{P}_2 = \{\hat{s}, \hat{v}_{1,2}, \hat{v}_{2,4}, \hat{v}_{4,6}, \hat{v}_{6,7}, \hat{t}\}$ are node-disjoint.

8.4.2 Modified line graphs $\hat{G}_1$ and $\hat{G}_2$

As mentioned above, the export policies prohibit certain paths between routing domains. In order to take into account these policies, we introduce several modifications to the line graph. First, we replace each undirected link that connects nodes in $\hat{G}$ by two directed links in opposite directions. Then, for each directed link in the resulting graph we check whether it can be used under the export policies specified in Table 2.1, and if not, the link is removed from the graph. We denote the modified line graph by $\hat{G}_1(\hat{V}_1, \hat{E}_1)$. Figure

8.4(c) depicts the modified line graph $\hat{G}_1$ that corresponds to the network depicted in Fig. 8.4(a). Note that link $(\hat{v}_{2,4}, \hat{v}_{4,5})$ is omitted from $\hat{G}_1$ because D_4 is a peer domain of both D_2 and D_5, hence it cannot forward packets from D_2 to D_5.

We summarize the properties of the line graph $\hat{G}_1$ in Proposition 8.4.2.

Proposition 8.4.2. *Let P be a (s,t)-path in G and let $\hat{P}$ be a corresponding $(\hat{s},\hat{t})$-path in $\hat{G}$. Then, if P can be used under the export policies listed in Table 2.1, the path $\hat{P}$ also belongs to $\hat{G}_1$. Further, for each $(\hat{s},\hat{t})$-path $\hat{P}$ in $\hat{G}_1$, there exists a corresponding path P in G that satisfies the export policies.*

In this section, we adapt the algorithm presented in Section 8.3 for finding link-disjoint paths that satisfy the export policies. To that end, we take advantage of the modified line graph $\hat{G}_1$ described before.

As discussed above, for any two link disjoint paths P_1 and P_2 in G that satisfy the export policies, there exist two corresponding paths $\hat{P}_1$ and $\hat{P}_2$ in $\hat{G}_1$. Moreover, such paths are node-disjoint. Since the algorithm presented in Section 8.3 finds two link-disjoint paths, we need to introduce the following modifications to $\hat{G}_1$ in order to take advantage of this algorithm.

1. We split each node $\hat{v}_{i,j}$ in $\hat{G}$ into two nodes $\hat{v}^1_{i,j}$ and $\hat{v}^2_{i,j}$, connected by a link $(\hat{v}^1_{i,j}, \hat{v}^2_{i,j})$, such that all links into (out of) $\hat{v}_{i,j}$ are now into $\hat{v}^1_{i,j}$ (out of $\hat{v}^2_{i,j}$). The weight of $(\hat{v}^1_{i,j}, \hat{v}^2_{i,j})$ is set to be the weight of the corresponding inter-domain link between D_i and D_j in the original network G.

2. We replace the special node $\hat{s}$ and all links $(\hat{s}, \hat{v}^1_{1,i})$ incident to $\hat{s}$ by the source routing domain D^s, such that each node $\hat{v}^1_{1,i}$ incident to $\hat{s}$ in $\hat{G}_1$ coincides with the corresponding border node of D^s (i.e., the border node of D^s incident to the inter-domain link that corresponds to $(\hat{v}^1_{1,i}, \hat{v}^2_{1,i})$).

The resulting graph is denoted by $\hat{G}_2$. Figure 8.4(d) depicts graph $\hat{G}_2$ that corresponds to the multi-domain network depicted in Fig. 8.4(a). It is easy to verify that for any two link-disjoint paths P_1 and P_2 in G the corresponding paths $\hat{P}_1$ and $\hat{P}_2$ in $\hat{G}_2$ are also link-disjoint.

The links of $\hat{G}_2$ can be classified into three groups. The first group includes links $(\hat{v}^1_{i,j}, \hat{v}^2_{i,j})$ represent the inter-domain links in the original network G. The second category include links $(\hat{v}^2_{i,j}, \hat{v}^1_{j,l})$ that represent paths through transit routing domains. Such links are referred to as *cross-domain* links. Finally, the third group includes the links that belong to the source and destination routing domains. For example, in Fig. 8.4(d), links $(\hat{v}^1_{1,3}, \hat{v}^2_{1,3})$ and

$(\hat{v}^1_{4,6}, \hat{v}^2_{4,6})$ correspond the the inter-domain links in G that connect domains D_1 to D_3 and D_4 to D_6, respectively. In addition, the links $(\hat{v}^2_{1,3}, \hat{v}^1_{3,4})$ and $(\hat{v}^2_{2,4}, \hat{v}^1_{4,6})$ in $\hat{G}_2$ are cross-domain links that represent paths through domains D_3 and D_4, respectively.

8.4.3 Disjoint path algorithm

The disjoint paths algorithm in the presence of export policies is an extension of the distributed algorithm presented in Section 8.3. In particular, the first step (in Algorithm FindAR) remains the same and only minor and straightforward modifications are needed for the third step of the algorithm. For the second step, we use Algorithm Find2DP-EP (presented in next page) that performs operations on the modified line graph $\hat{G}_2$. Thus, the line graph $\hat{G}_2$ should be constructed prior to the application of the algorithm.

The algorithm uses the following definitions. For each cross-domain link $(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$ we denote by $e'(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$ and $e''(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$ the inter-domain links in G that correspond to nodes $\hat{v}_{i,j}$ and $\hat{v}_{j,k}$ in the line graph, respectively. We also denote by $x(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$ and $y(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$ the border nodes of D_j incident to links $e'(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$ and $e''(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$, respectively.

The algorithm begins by assigning to each cross-domain link $(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$ of $\hat{G}_2$ the weight equal to the minimum weight of a path between the border nodes $x(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$ and $y(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$ of the routing domain D_j that corresponds to $(\hat{v}^2_{i,j}, \hat{v}^1_{j,k})$. The minimum weights of the shortest paths are available through array M'_j. Next, we compute a minimum weight $(s, \hat{t})$ path P_1 in $\hat{G}_2$, which corresponds to the minimum weight of a (s,t)-path in G that satisfies the export policies.

Next, for each cross-domain link $(\hat{v}^2_{ij}, \hat{v}^1_{jk}) \in P_1$ we perform the following operations. Let D_j be the routing domain that corresponds to $(\hat{v}^2_{ij}, \hat{v}^1_{jk})$. Note that due to Assumption 8.3.1 path P_1 traverses each domain at most once. Then, we assign the weight of each cross-domain link $(\hat{v}^2_{zj}, \hat{v}^1_{jw})$ that corresponds to D_j (including $(\hat{v}^2_{ij}, \hat{v}^1_{jk})$) according to array $M_j^{x_1,y_1}(x_2, y_2)$, where $x_1 = x(\hat{v}^2_{ij}, \hat{v}^1_{jk})$, $y_1 = y(\hat{v}^2_{ij}, \hat{v}^1_{jk})$, $x_2 = x(\hat{v}^2_{zj}, \hat{v}^1_{jw})$, and $y_2 = y(\hat{v}^2_{zj}, \hat{v}^1_{jw})$. This operation correspond to lines 13 and 14 of Algorithm Find2DP.

Finally, the source node sends the paths P_1 and P_2 to every routing domain D_i traversed by these paths. At each domain, the PCE expands every cross-domain link of P_1 and P_2 into an intra-domain path by using the methods described in Section 8.3.4.

The following theorem summarizes the correctness of the algorithm.

Algorithm 5 Find2DP-EP($\hat{G}_2$, $\{A_i\}$)

Input: $\hat{G}_2$ - the modified line graph of G
For each routing domain D_i
$A_i = \{M_i'\} \cup \{M_i^{j,l} \ / \ b_j \in B_i \ , b_l \in B_i\}$ of D_i , the aggregated representation of D_i

Output: Two paths P_1 and P_2 in $\hat{G}_2$.

1: **for** each cross-domain link $(\hat{v}_{ij}^2, \hat{v}_{jk}^1)$ of $\hat{G}_2$ **do**
2: $w_{(\hat{v}_{ij}^2, \hat{v}_{jk}^1)} \leftarrow M_j'(x(\hat{v}_{ij}^2, \hat{v}_{jk}^1), y(\hat{v}_{ij}^2, \hat{v}_{jk}^1))$
3: **end for**
4: Find a shortest path P_1 between s and $\hat{t}$ in $\hat{G}_2$
5: **for** each cross-domain link $(\hat{v}_{ij}^2, \hat{v}_{jk}^1) \in P_1$ **do**
6: $x_1 \leftarrow x(\hat{v}_{ij}^2, \hat{v}_{jk}^1)$
7: $y_1 \leftarrow y(\hat{v}_{ij}^2, \hat{v}_{jk}^1)$
8: **for** each cross-domain link $(\hat{v}_{zj}^2, \hat{v}_{jw}^1)$ that corresponds to domain D_j **do**
9: $x_2 \leftarrow x(\hat{v}_{zj}^2, \hat{v}_{jw}^1)$
10: $y_2 \leftarrow y(\hat{v}_{zj}^2, \hat{v}_{jw}^1)$
11: $w_{(\hat{v}_{zj}^2, \hat{v}_{wk}^1)} \leftarrow M_j^{x_1, y_1}(x_2, y_2)$
12: **end for**
13: **end for**
14: Find a shortest path P_2 between s and $\hat{t}$ in $\hat{G}_2$

Theorem 8.4.3. *The proposed algorithm finds two disjoint paths between s and t that satisfy the export policies at minimum possible weight.*

Proof: The proof follows the same lines as that of Theorem 8.3.1.

The computational complexity of the algorithm presented in this section is similar to that of the algorithm presented in Section 8.3.

Chapter 9

Maximum MPLS Coverage at Minimum Cost

As discussed in Chapter 7, service providers have at present several incentives to extend the reach of long-lived MPLS paths across domains. A limitation, however, is that end-to-end MPLS connectivity is only feasible if intermediate transit providers support the establishment of MPLS paths through them. To this end, transit providers will negotiate special peering agreements with their MPLS customers [83, 54], for which the latter will be charged. Independently of the incentives that providers might have for extending MPLS beyond their boundaries (e.g. extending their VPN offer, distributing content, or providing end-to-end QoS support), providers will naturally incur additional costs for the establishment and maintenance of long-lived MPLS paths through transit providers.

Therefore, a key problem to be faced by providers is how to optimally solve the trade-off of exploiting as much as possible the advantages of having long-lived MPLS coverage, against the extra cost that this coverage will represent. In practice, providers will have an estimation of the additional income that can be obtained due to the MPLS coverage, as well as an expectation in terms of revenue. The difference between these two determines the budget that can be spent.

A provider will often have multiple candidate paths to reach the destination domains that it is willing to cover, and these paths can have associated different monetary costs. In practical settings, providers might not always have the chance to choose the cheapest alternative for their coverage strategy. This could be either because some of the targeted domains are not reachable through the cheapest alternatives (e.g. due to peering or routing policies), or because the available capacity is not enough to allocate the traffic demands.

The subject of this chapter is to formulate and efficiently solve the multi-objective decision problem of how a domain can maximize the MPLS coverage of traffic demands with minimum cost, subject to a budget and network capacity constraints. The problem is studied in the context of a PCE-based multi-domain MPLS network. In our model, the PCEs are used for both the

computation of the paths and the advertisement of routing information, taking advantage of the aggregated representation for a multi-domain network that we described in Chapter 8.

This chapter proposes an offline solution, based on the knowledge of the aggregated representation of the network provided by the PCE in the domain and the monetary costs of the candidate paths. The details about how a domain is aware of the monetary costs of the paths offered by its providers are out of the scope of this thesis (they could be either attached to the advertisements exchanged between the PCEs, or they could be handled offline). This study makes the following contributions.

- We show that, based on realistic assumptions and with only minor knowledge about the pricing schemes of transit providers, it is possible to steer the search of potential solutions towards specific regions in the objective space.
- We propose an *Evolutionary Multi-objective Algorithm* (*EMA*) that exploits the above finding to find candidate solutions in the Pareto sense, in a fast and highly efficient way.
- We prove that the search and update strategy of our EMA guarantees the elitism of the candidates monotonically.
- We also prove that our algorithm converges to an ε-Pareto Set.
- The evaluation results show that our EMA is capable of finding much better solutions than another powerful and highly utilized EMA, namely, the *Strength Pareto Evolutionary Algorithm* version 2 (SPEA2) [143, 53, 72, 75, 89], while complying with the network capacity and budget constraints.

A major advantage of the analysis and solution proposed in this chapter is that it can be easily generalized, and applied in other settings where constrained problems considering maximum coverage vs. cost are critical.

9.1 The network model

The multi-domain MPLS network model proposed here is supported by a PCE-based architecture [34]. Our focus is on a rather small set of MPLS domains Ω, in which pairs of domains in Ω have already negotiated peering agreements – like the ones proposed in [83, 54] – supporting the transit and termination of MPLS paths.

In this scenario, the multi-domain MPLS network is captured by the representation $G(V, E)$ that includes an aggregated representation of each domain in Ω, as well as inter-domain links. In this study, we use the aggregated representation scheme for a multi-domain network that we described in Chapter 8 [113]. This aggregated representation captures the transitional characteristics of the network, while guaranteeing that the confidentiality and administrative limits between domains are respected.

The key advantage of this approach is that it allows the PCE located in a source domain to optimally compute entire paths towards any reachable destination domain in Ω, avoiding coarse-grained solutions, like those relying on crancback [33] (notice that only destination domains in Ω for which the source domain has already negotiated a peering agreement will be reachable by the source).

In this general setting, a source domain S would like to establish long-lived MPLS paths to K destination domains $D_1, \ldots, D_K$. The total traffic volume to be covered between S and each D_i, $1 \leq i \leq K$, is known and it is denoted as d_i. Each link e in E has a bound $\hat{C}_e$ on the total amount of traffic that can be forwarded through e.

Based on the aggregated representation of the network, the PCE in the source domain S will typically have multiple alternative paths for the establishment of an MPLS path towards a destination domain D_i (see Fig. 9.1), each of which may have a different cost. For each destination domain D_i, the PCE in S has a set of feasible paths $P_i^1, \ldots, P_i^{N_i}$ that connect S and D_i, where N_i denotes the number of such paths. Here, the term "feasible" indicates that each of the N_i candidate paths will support the establishment of an MPLS path for the expected traffic demand d_i between S and D_i. Each path P_i^j, $1 \leq j \leq N_i$, is associated with a cost $c(P_i^j)$ that captures the total amount and reliability of network resources that need to be allocated in order to support forwarding traffic of volume d_i along P_i^j. We assume that domain S is multihomed to M ISPs (see Fig. 9.1), each of them with its own pricing scheme. Typically, the difference in the cost of the feasible paths will depend on the reliability offered by the different ISPs of S, as well as on their charging schemes (total volume, percentile-based, etc.) [129]. The source S has an overall budget B that can be spent on all paths.

In this framework, the goal of the source domain S is to maximize the coverage of traffic demands using MPLS paths with minimum cost, subject to the network capacity and the budget restriction B. We formulate this multi-objective problem in next section.

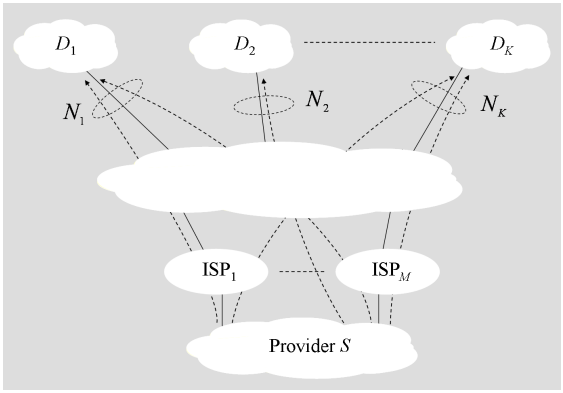


Figure 9.1: The Network Model.

9.2 Problem formulation

This section formulates the problem, and introduces a basic – and realistic – set of assumptions that will let us build the key contributions in this chapter.

Assumption 9.2.1. *The pricing functions of S's ISPs are increasing and concave in the traffic demand (with decreasing unit cost as the traffic volume increases). This reflects the fact that MPLS transit providers will expect decreasing marginal returns as the traffic demand of customers grow. This is precisely the case of the typical pricing schemes offered by ISPs at the moment [129]. Our assumption is that this will also be the case in multi-domain MPLS-enabled networks.*

Assumption 9.2.2. *The traffic demands that S would like to cover are all of the same nature. In other words, the traffic demands are only distinguishable by their volume. In a more complex scenario, S might have incentives to cover some of the traffic demands explicitly using more expensive paths. For the sake of simplicity, we consider that the traffic demands to be covered are equally treated by S.*

Assumption 9.2.3. *As in [129], we assume that traffic volume and reliability are the variables that MPLS providers will consider in their pricing functions. Each of S's ISPs has its own pricing function. In [129], the assumption was that reliability is a feature tightly linked to the ISP selection. In a multi-domain MPLS framework, reliability might be a different feature, since it depends on the peering agreements between S's ISPs and the transit domains that need to be crossed to reach each destination D_i. Therefore, our model is more general, in the sense that reliability is not linked to the ISP subscription, and the same ISP might offer more reliable or less reliable paths depending on its agreements.*

Assumption 9.2.4. *For a given traffic volume d_i, the variation in the cost $c(P_i^j)$ of the feasible paths P_i^j is due to the difference in their reliability as offered by S's ISPs. By Assumption 9.2.3, it is clear that the costs $c(P_i^j)$ are denoted without an ISP sub-index, given that the costs are functions of the paths per-se, rather than the ISP that is offering the candidate path.*

Assumption 9.2.5. *The cheapest paths considered in the problem formulation are sufficiently reliable for S's purposes. If some paths are not reliable enough, we assume that they have been already discarded (either by an offline process or by the routing policies ruling the PCE in S). Thus, S will prefer cheaper paths for covering a set of traffic demands, unless the allocation of the demands becomes unfeasible (e.g., blocking occurs). In such a case, S will try to arrange its coverage by shifting some demands to more expensive paths.*

This basic set of assumptions – which we claim are realistic – is all that is needed in order to develop the rest of our analysis. For each path P_i^j we associate a decision variable δ_i^j that is equal to 1 if the path P_i^j is selected, and 0 otherwise. Then, the problem can be formulated as the following multi-objective integer program (Table 9.1 introduces the notation used):

$$max\left[\sum_{i=1}^{K}\sum_{j=1}^{N_i}\delta_i^j d_i\ ,\ -\sum_{i=1}^{K}\sum_{j=1}^{N_i}\delta_i^j c(P_i^j)\right] \tag{9.1}$$

$$s.t. \qquad \forall i \in \{1,\ldots,K\}:\ \sum_{j=1}^{N_i}\delta_i^j \leq 1 \tag{9.2}$$

$$\forall e \in E:\ \sum_{P_i^j : e\in E}\delta_i^j d_i \leq \hat{C}_e \tag{9.3}$$

$$\sum_{i=1}^{K}\sum_{j=1}^{N_i}\delta_i^j c(P_i^j) \leq B \tag{9.4}$$

$$\forall i \in \{1,\ldots,K\} \quad \text{and} \quad \forall j \in \{1,\ldots,N_i\}:\ \delta_i^j \in \{0,1\} \tag{9.5}$$

Expression (9.1) represents the objective vector, where minimizing the total cost is equivalent to maximizing its negative. Expression (9.2) ensures that at most one path is selected per domain. Expression (9.3) ensures that the total traffic on all selected paths that use link e does not exceed its

Symbol	Description
	Known Data
$G(V, E)$	The aggregated representation of the multi-domain MPLS network
S	The source Provider
K	Number of destination domains for which S is willing to provide MPLS coverage
D_i	One of the destination domains to be covered: $i \in \{1, \ldots, K\}$
d_i	Aggregate traffic demand to be covered between S and D_i
D	Total traffic demands that S is willing to cover: $\sum_{i=1}^{i=K} d_i$
N_i	Number of feasible paths between S and D_i
$c(P_i^j)$	Cost of establishing and maintaining an MPLS path between S and D_i using path P_i^j
B	Budget of the source provider S
$\hat{C}_e$	The available capacity of link e before the coverage. Capacities are known by the PCE in S by means of the distributed routing process running between PCEs [113]. This routing process is what allows the PCE in S to assemble $G(V, E)$
	Unknown Data
δ_i^j	Decision variable $\{0, 1\}$ depending if path P_i^j is chosen or not
$\hat{D}$	Total demand finally covered: $\hat{D} = \sum_{i=1}^{K} \sum_{j=1}^{N_i} \delta_i^j d_i$

Table 9.1: Notation.

capacity $\hat{C}_e$. Note that the summation here is over all paths that use that link. Expression (9.4) ensures that the total cost of all selected paths does not exceed the total budget B. Finally, (9.5) ensures that each δ_i^j is either 0 or 1.

Independently of the technique chosen for solving a multi-objective optimization problem like (9.1), the outcome is a set of candidate solutions, which, typically, will not contain one that is better than the rest in each of

the objectives. In our problem, while some candidate solutions will increase the coverage (being more costly), others will be cheaper but at the price of providing poorer coverage.

Therefore, an additional step is typically needed in a multi-objective optimization problem, namely, a *Decision Maker* (DM), which, based on the set of candidate solutions, chooses the one that better fits the problem that is being solved. A general and widely used approach for the decision maker is to use a linear combination of the normalized objectives. Our specific approach is to use:

$$DM = sup\left[\frac{w_1}{D}\sum_{i=1}^{K}\sum_{j=1}^{N_i}\delta_i^j d_i + w_2\left(1 - \frac{1}{B}\sum_{i=1}^{K}\sum_{j=1}^{N_i}\delta_i^j c(P_i^j)\right)\right] \tag{9.6}$$

where, $w_k \in [0,1]$, $k = 1,2$ are the weights, and $w_1 + w_2 = 1$. It is important to note that, in (9.6), each of the normalized objectives varies between 0 and 1, since D is the total traffic volume that S would like to cover (see Table 9.1), and the normalized cost objective is given under the budget constraint in (9.4).

The most complex trade-off to solve is when $w_1 = w_2 = \frac{1}{2}$, i.e., when the objectives are equally important for the decision maker. Accordingly, we conservatively consider:

$$DM = DM\,|_{w_1=w_2=\frac{1}{2}} \tag{9.7}$$

It is worth highlighting some properties of (9.7):

- Candidate solutions with $DM < \frac{1}{2}$ are of no interest to our purpose. From (9.7) it is easy to see that zero coverage (i.e. $\delta_i^j = 0\ \forall\ i,j$) offers the decision maker a value of $DM = \frac{1}{2}$. Thus, all candidate solutions with $DM < \frac{1}{2}$ are of no practical interest, given that the decision maker will assess them worse than having no coverage at all. It is clear that the problem we are trying to solve is to find the best possible trade-off between coverage and cost in a setting where blocking might occur. In other words, rather than spending the entire budget B, the provider in domain S is seeking the most beneficial combination of coverage vs. cost.

- A $DM \geq \frac{1}{2}$ is where the potential solutions can be found. In particular, $DM^{max} = 1$ represents the optimal case (i.e., maximum coverage for free, see (9.7)).

It can be easily shown that the size of the problem is:

$$\sum_{n=0}^{K-1} \binom{K}{K-n} \prod_{i=1}^{K-n} (N_i + 1) \tag{9.8}$$

which, with an average of $N_i = 5$ candidates paths per-destination and only 20 destination domains, gives more than 7.6e+16 alternatives for the coverage.

We note that a single objective version of the problem in (9.1) is a special case of the *Multi-dimensional Knapsack Problem* (*MKP*) [65]. The problem can also be seen as a special case of general integer programming with the only restriction that all coefficients are positive and the variables are zero or one. It has been recognized that the MKP problem is a particularly difficult problem that requires sophisticated methods to be solved. We also note that in our problem the constraints matrix is relatively sparse, while the MKP problem is typically characterized by a dense constraint matrix. In this chapter, we exploit this property to design an algorithm that finds a set of near-optimal solutions in an efficient manner. Our focus here is on the more general multi-objective problem.

Evolutionary multi-objective optimization techniques offer a suitable and efficient way of solving large problems like the MKP problem, and like our problem [142]. The key of their efficiency lies in the update and search strategy adopted. We shall show that it is possible to take advantage of the minor knowledge about the pricing functions of S's ISPs, and exploit it to develop a fast and efficient search and update strategy for the evolutionary algorithm that we propose in Section 9.5.

In the next section, we introduce the basic background that is needed to understand our main contributions, which are later presented in Sections 9.4 and 9.5.

9.3 Pareto optimality: background

Consider the multi-objective problem in (9.1). This problem has two objectives y_k , $k = 1, 2$ and one *decision variable* $\delta_i^j \in X$, where X denotes the *decision space*. For each δ_i^j in (9.1) a specific coverage and total cost is chosen, so the *objective vector* is given by: $y(\delta_i^j) = [y_1, y_2] \in Y$, where Y denotes the *objective space*. Accordingly, the objective vector represents the mapping between the decision space and the objective space. Our goal is to maximize the objective vector. We now present a set of basic definitions regarding the concepts of Pareto dominance and Pareto optimality.

Definition 9.3.1. *(**Pareto Dominance**) A vector $y \in Y$ is said to dominate another vector $y' \in Y \Leftrightarrow y_k \geq y'_k \ \forall \ k$, and there exists at least one element such that $y_k > y'_k$. The dominance relationship is denoted as: $y \succ y'$.*

Definition 9.3.2. *(**Pareto Optimality**) A candidate solution $x \in X$ is said to be optimal in the Pareto sense $\Leftrightarrow$ there does not exist another solution $x' \in X$, such that $y(x') \succ y(x)$.*

Definition 9.3.3. *(**Pareto Set**) The Pareto Set, $X^* \subseteq X$ of a given multi-objective problem is the set of all candidate solutions $x^* \in X$ that are Pareto Optimal.*

Definition 9.3.4. *(**Pareto Front**) The Pareto Front, $Y^* \subseteq Y$ of a given multi-objective problem is the set of all $y(x^*) \in Y$ such that $x^* \in X^*$.*

Definition 9.3.5. *(ε-**Dominance**) Let $y, \ z \in Y$, then y is said to ε-dominate z for some $\varepsilon > 0 \Leftrightarrow (1+\varepsilon)y_k \geq z_k \ \forall \ k$. The ε-dominance relationship is denoted as: $y \succ_\varepsilon z$.*

Definition 9.3.6. *(ε-**Pareto Front**) An ε-Pareto Front Y^*_ε is a subset of the Pareto Front ($Y^*_\varepsilon \subseteq Y^*$), which ε-dominates the Pareto Front: $y_\varepsilon \succ_\varepsilon y^*$, $\forall \ y^* \in Y^*$ and $y_\varepsilon \in Y^*_\varepsilon$.*

Definition 9.3.7. *(ε-**Pareto Set**) An ε-Pareto Set X^*_ε is the set of candidate solutions x^*_ε in the decision space whose images $y(x^*_\varepsilon)$ belong to the ε-Pareto Front Y^*_ε.*

The concept of Pareto optimality basically states that it is not possible to improve some of the objectives without worsening at least one of the others. The Pareto Set is the collection of all candidate solutions that are Pareto optimal, i.e., candidates that are not dominated by any other candidate in the decision space.

The role of the *decision maker* is to choose one among the possible solutions present in the Pareto Set. In many large problems, the size of this set is practically prohibitive. For that reason, the most efficient techniques developed so far to tackle large-sized problems focus on finding a reduced set of candidate solutions, whose objective vectors are good approximations of those present in the Pareto Front. The decision maker then chooses one among the candidates present in this reduced (and approximate) set. This is the basis for the practical interest in the ε-Pareto approximation.

We now develop the main ideas of how we propose to search for candidate solutions in the objective space, by exploiting the knowledge about the shape of the pricing functions of S's ISPs.

9.4 The search and update strategy

We start by presenting a strategy for partitioning the objective space, which will help to steer the search of potential solutions towards specific regions in this partition in a fast and highly efficient way. Then, we introduce two lemmas and a theorem, which together support the search strategy. The search and update strategy proposed in this section will be the engine of the evolutionary search in Section 9.5.

A Partitioning Strategy of the Objective Space

It is worth highlighting that the partitioning process that we describe next is simply a way of splitting the objective space, and it is performed before even starting the search of feasible solutions. This partitioning process is carried out in steps, and the outcome of this process is depicted in Fig. 9.2.

We start with the total traffic demand D (i.e., the maximum possible coverage), and draw the maximum and minimum possible costs associated with it in the objective space. These are shown as $min(c(D))$ and $max(c(D))$ in Fig. 9.2. In the partitioning process, this is step zero.

In the next step (step one), we extract from D the minimum demand d_i that S wants to cover, i.e., $min\{d_i\}$, $1 \leq i \leq K$, obtaining (9.9), which represents the maximum traffic volume that can be covered taking into account $(K-1)$ domains. We repeat this process, but now instead of extracting the $min\{d_i\}$ from D, we extract the $max\{d_i\}$, $1 \leq i \leq K$, obtaining (9.10). Equation (9.10) represents the minimum traffic volume that can be covered taking into account $(K-1)$ domains.

$$D - min\{d_i\} = max(D_{K-1}) \quad i = 1, \ldots, K \tag{9.9}$$

$$D - max\{d_i\} = min(D_{K-1}) \quad i = 1, \ldots, K \tag{9.10}$$

Once we have obtained $max(D_{K-1})$ and $min(D_{K-1})$, we compute the maximum and minimum possible costs associated with them, that is, the $max(c(D_{K-1}))$ and $min(c(D_{K-1}))$.

In the second step of the partitioning process, we only consider the remaining traffic demands that S is willing to cover, i.e., the traffic demands $\{d_i^{(1)}\}$, $1 \leq i \leq (K-2)$ that were not extracted in step one. With this new set, we repeat the process by extracting the minimum and the maximum

of the remaining traffic demands from (9.9) and (9.10), respectively. The following recurrence relation represents the partitioning process at step r:

$$max(D_{K-r}) - min\{d_i^{(r)}\} = max(D_{K-r-1}) \quad i \in [1, K-2r] \tag{9.11}$$

$$min(D_{K-r}) - max\{d_i^{(r)}\} = min(D_{K-r-1}) \quad i \in [1, K-2r] \tag{9.12}$$

In the same way, the $max(c(D_{K-r}))$ and $min(c(D_{K-r}))$ represent the maximum and minimum possible costs to cover $(K-r)$ domains, respectively. To simplify the notation we define $L = (K-r)$. To complete the partition, we cut the accumulated costs according to the budget constraint B.

From Fig. 9.2 it can be seen that the objective space is divided in a set of overlapping boxes that we call Q_L, $1 \leq L \leq K$, where the sub-index L represents the number of destination domains covered.

In sum, the overall partitioning process is rather simple. For each subset L of the K destination domains that S is willing to cover, we represent in the objective space the maximum and minimum possible traffic demands considering L out K domains. We also represent the maximum and minimum possible costs associated with the coverage of those L domains. Clearly, all feasible solutions – if any – will be inside the area delimited by the union of the Q_L boxes that remain smaller than the budget constraint B. In the sequel, we formalize some of the concepts introduced above, and propose an efficient search strategy.

The Search Strategy in the Objective Space

Figures 9.2 and 9.3.a will help to understand the following definitions.

Definition 9.4.1. *A box Q_L in the partitioned objective space Y is defined as the set $P \in Y$, $P : (y_1(P), y_2(P))$ such that:*

$$min(D_L) \leq y_1(P) \leq max(D_L) \ \wedge \ min(c(D_L)) \leq y_2(P) \leq max(c(D_L))$$

Definition 9.4.2. *The vertex V_L of a box Q_L is defined as:*

$$V_L = \{P \in Q_L \ / \ y_1(P) = max(D_L) \ \wedge \ y_2(P) = min(c(D_L))\}$$

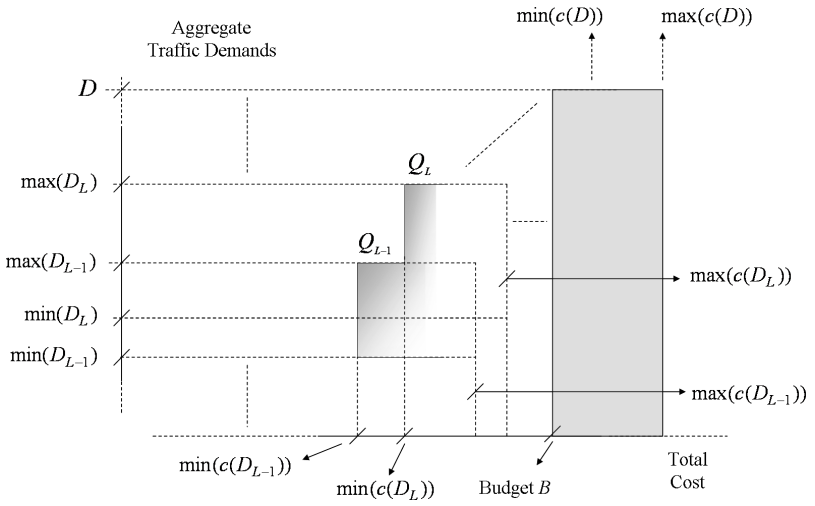


Figure 9.2: Partitioning the objective space.

Definition 9.4.3. *An iso-objective curve in the partitioned objective space Y is defined as:*

$$i(P) = \{P \in Y \ / \ DM(P) = constant\}$$

From (9.7), it can be easily shown that that the iso-objective curves are lines with slope $(\frac{D}{B})$, so the iso-objective line through $P_0 \in Y$ is given by:

$$i(P_0): \quad y_1 = \left(\frac{D}{B}\right)(y_2 - y_2(P_0)) + y_1(P_0) \tag{9.13}$$

In particular, Fig. 9.3.a shows the iso-objective line through the vertex V_L of the box Q_L.

Definition 9.4.4. *The region $\hat{Q}_L(P_0) \subset Q_L$ is defined as the set $P \in Q_L$ such that:*

$$\hat{Q}_L(P_0) = \left\{P \in Q_L \ / \ y_1(P) > \left(\frac{D}{B}\right)(y_2(P) - y_2(P_0)) + y_1(P_0)\right\}$$

As an illustrative example, Fig. 9.3.a shows $\hat{Q}_{L+1}(V_L)$. From the example, it can be seen that $\hat{Q}_{L+1}(V_L)$ is the portion of the box Q_{L+1} that remains above the iso-objective line that passes through the vertex V_L of the box Q_L.

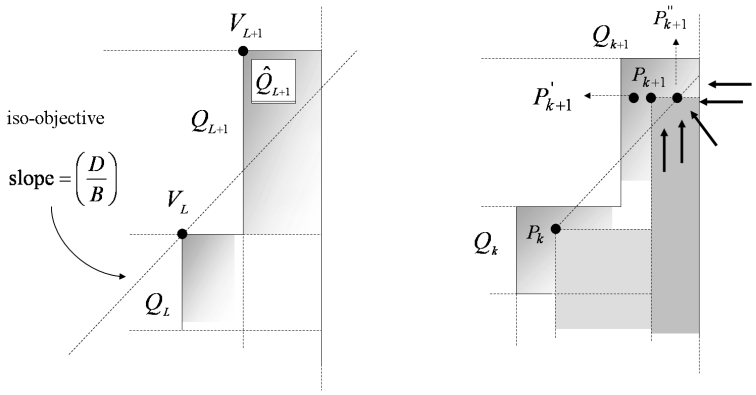


Figure 9.3: The search and update strategy.
(a) The left hand side shows the boxes, vertices, and an iso-objective line in the partitioned objective space. (b) Steering the search towards dominant individuals (right). The figure shows the iso-objective line through P_k. P''_{k+1} belongs to the iso-objective, so it is as good as P_k for the decision maker. On the other hand, P_{k+1} and P'_{k+1} are above the iso-objective and closer to the vertex of the box Q_{k+1}. Theorem 9.4.3 proves that both P_{k+1} and P'_{k+1} will be assessed better than P_k and P''_{k+1} by the decision maker.

Definition 9.4.5. *A vertex V_L is said to be feasible $\Leftrightarrow$ the total cost $y_2(V_L) < B$, and there exists a combination of feasible paths with enough capacity to cover the traffic demands $y_1(V_L)$ with $DM(V_L) \geq \frac{1}{2}$.*

Definition 9.4.6. *A vertex V_L is said to be the maximum feasible vertex $\Leftrightarrow \forall$ feasible vertex $V_k \neq V_L$, $y_1(V_L) > y_1(V_k)$.*

We proceed to explain the central concepts behind Definitions 9.4.5 and 9.4.6. For each box Q_L, $1 \leq L \leq K$, it is easy to observe that its vertex V_L is a dominant point in Q_L. This is because no other point inside Q_L offers a better coverage than V_L with a cheapest cost. Therefore, the vertices in the partitioned objective space are potential candidates to belong to the Pareto Front. However, the network capacity and budget constraints might cause that some – or even all – vertices encode points in the objective space that cannot be reached. In this context, Definition 9.4.5 establishes the conditions under which a vertex is reachable (i.e., feasible), whereas Definition 9.4.6 characterizes out of the set of vertices that are feasible, one that produces the maximum coverage. Note that, in Definition 9.4.5, we have $DM(V_L) \geq \frac{1}{2}$,

since as we mentioned in Section 9.2, points in the objective space with a $DM(V_L) < \frac{1}{2}$ have no practical interest.

Clearly, all the feasible vertices encode mutually non-dominated solutions, so in principle, it is not possible to choose one vertex over another – at least in the Pareto sense. Moreover, depending on the traffic volume that S is willing to cover, and the capacities and budget restrictions, many other points that are not vertices will be also feasible and thus are potential candidates to be part of the Pareto Front. The following lemmas and Theorem 9.4.3 set the basis of the strategy of how we plan to find the best candidate solutions among such points.

Lemma 9.4.1. *Let V_L be a feasible vertex in the partitioned objective space Y. If there $\exists\ P \in \hat{Q}_{L+1}(V_L)$ then: $DM(P) > DM(V_L)$.*

Figure 9.3.a helps to understand the statement of Lemma 9.4.1. The feasible vertices are always potential candidates for the optimal solution of the problem; given that they represent a feasible coverage at minimum cost (recall that each vertex dominates its own box). However, due to capacity and budget constraints some vertices might not be available. What Lemma 9.4.1 states is that under the general – and frequently used – decision strategy given in (9.7), candidate solutions encoded by points that are close to vertices of higher demands will be assessed better by the decision maker than vertices encoding lower demands.

Proof. (Lemma 9.4.1) Let $P \in \hat{Q}_{L+1}(V_L)$, by Definition 9.4.4:

$$y_1(P) > \left(\frac{D}{B}\right)(y_2(P) - y_2(V_L)) + y_1(V_L) \Rightarrow \frac{y_1(P)}{2D} + \frac{1}{2} - \frac{y_2(P)}{2B} >$$

$$\frac{y_1(V_L)}{2D} + \frac{1}{2} - \frac{y_2(V_L)}{2B}, \text{ then using (9.7)} \Rightarrow DM(P) > DM(V_L)$$

■

Lemma 9.4.2. *Let V_{L+1} be a feasible vertex, then: $y_1(V_{L+1}) > i(V_L)|_{y_2(V_{L+1})}$*

Lemma 9.4.2 states that the iso-objective line through the vertex V_L (see (9.13)), evaluated at $y_2(V_{L+1})$ is smaller than the traffic demand encoded by the feasible vertex V_{L+1}. This has two major corollaries. First, it shows that the graphical representation in Fig. 9.3.a is consistent when the objectives

are equally important (i.e. V_{L+1} is above the iso-objective line through V_L). The second corollary is that since V_{L+1} is feasible $\Rightarrow V_L, V_{L-1}, V_{L-2}, \ldots$ they are all feasible too. This can be shown by simply removing from the coverage the difference in the demand between V_{L+1} and V_L, and so on. Now using Lemma 9.4.1 and Lemma 9.4.2, it is easy to show that if V_{L+1} is feasible, then:

$$DM(V_{L+1}) > DM(V_L) \tag{9.14}$$

Hence, vertices encoding higher traffic demands will be assessed better by the decision maker than vertices encoding lower traffic demands.

Proof. (Lemma 9.4.2) By way of contradiction, let us assume that:

$$\begin{aligned} y_1(V_{L+1}) \leq \left(\frac{D}{B}\right)(y_2(V_{L+1}) - y_2(V_L)) + y_1(V_L) \Rightarrow \\ \frac{y_1(V_{L+1}) - y_1(V_L)}{y_2(V_{L+1}) - y_2(V_L)} \leq \frac{D}{B} \Rightarrow \frac{\Delta d}{\Delta c} \leq \frac{D}{B} \end{aligned} \tag{9.15}$$

where, Δd and Δc represent the variations in the traffic demand and in the cost, respectively, while increasing the coverage from $V_L \rightarrow V_{L+1}$. Given that V_{L+1} is feasible (Definition 9.4.5) and using Assumption 9.2.1 (see Section 9.1):

$$\frac{\Delta c}{c(V_L)} < \frac{\Delta d}{d(V_L)} \Rightarrow \frac{d(V_L)}{c(V_L)} < \frac{\Delta d}{\Delta c} \Rightarrow \text{by using 9.15 we have:}$$

$$\frac{d(V_L)}{D} - \frac{c(V_L)}{B} < 0 \Rightarrow \frac{1}{2} + \frac{d(V_L)}{2D} - \frac{c(V_L)}{2B} = DM(V_L) < \frac{1}{2}$$

Since V_{L+1} is feasible, V_L is feasible too. By Definition 9.4.5, $DM(V_L) \geq \frac{1}{2}$, which contradicts with he above result.

■

Theorem 9.4.3. *Let V_L be the maximum feasible vertex.*
If $\bigcup_{k>L} \hat{Q}_k(V_L) \neq \emptyset \Rightarrow$ the optimum $\in \bigcup_{k>L} \hat{Q}_k(V_L)$. If, on the contrary, $\bigcup_{k>L} \hat{Q}_k(V_L) = \emptyset \Rightarrow V_L$ is optimum as well as any P / $P \in i(V_L)$.

Proof. (Theorem 9.4.3)

Case 1) Let $\Psi = \bigcup_{k>L} \hat{Q}_k(V_L) \neq \emptyset$.

By induction and Lemma 9.4.1, it is easy to prove that any $P \in \Psi$ complies with: $DM(P) > DM(V_L)$, which, in simple terms, means that the points P in the boxes above the iso-objective line through V_L and to the right of V_L are better assessed by the decision maker than V_L. Next, using again an inductive argument and employing Lemma 9.4.2, it is possible to show that the points P in the boxes below the iso-objective line through V_L and to the left of V_L are worse than V_L for the decision maker. Therefore, the optimal solution is in Ψ.

Case 2) $\bigcup_{k>L} \hat{Q}_k(V_L) = \emptyset$.

By induction and Lemma 9.4.2, V_L is better assessed by the decision maker than any other feasible vertex, given that it is the maximum. Since these vertices are the only potential candidates below the iso-objective line through V_L and to the left of V_L, V_L stays as a better candidate. Given that the region Ψ above the iso-objective line through V_L is empty, the optimal solution belongs to the iso-objective $DM(V_L)$.

■

The key advantage of Theorem 9.4.3 is that it can be used to steer the search towards dominant points in the objective space in a fast and highly efficient way using an evolutionary algorithm.

The Update Strategy in the Objective Space

We proceed to describe the combination of the search and update strategy for the evolutionary algorithm that we shall specify in Section 9.5. The first step of the strategy is to get the set of feasible vertices. There are two possible cases: i) at least one feasible vertex exists; ii) no feasible vertices can be obtained under the current capacities of the links and the budget constraint. In the first case, clearly a maximum feasible vertex exists. By Theorem 9.4.3, if a feasible vertex exists, then the optimal solution is either the one encoded by the vertex itself – and its iso-objective line – or else it is located inside the region Ψ above the iso-objective through the maximum feasible vertex. Our approach is to steer the search towards points above the iso-objective that passes through the maximum feasible vertex, and that are

located close to the upper set of unfeasible vertices. Then, each time that we find a new candidate solution $P \in \Psi$ (see Theorem 9.4.3), the *update* strategy is to compute the iso-objective line through P and steer again the search above its iso-objective line. New potential solutions will be located above this iso-objective line and close to the unfeasible vertices. The search and update strategy is depicted in Fig. 9.3.b.

As mentioned before, there might be some extreme cases where none of the vertices is feasible. In such cases, we use the same search and update strategy as described above, but rather than starting the search from the maximum feasible vertex, we start with an initial population that is randomly generated and is close to the vertices. By mating and variation [142] we get a first feasible candidate – if any – and then steer the search as described above.

9.5 Evolutionary multi-objective algorithms

The goals of this section are to introduce our EMA, its main properties – particularly its elitism and convergence – and briefly describe the main features of an improved version of the *Strength Pareto Evolutionary Algorithm* (*SPEA*), namely, SPEA2, which is a well-known and powerful evolutionary algorithm [143, 53, 72, 75, 89]. The performance of our EMA is contrasted against SPEA2 in Section 9.6.

The reader who wants to get deeper into evolutionary multi-objective optimization can find an excellent tutorial here [142].

9.5.1 Maximum Coverage at minimum Cost (MC^2)

We proceed to describe our EMA, named *Maximum Coverage at minimum Cost* (MC^2). The algorithm is shown in Algorithm 6.

As many EMA do, we keep an *archive* A composed by the best potential solutions found throughout the evolutionary process. For the initial population we choose the set of feasible vertices in the partitioned objective space – if any – or we randomly generate a set of candidates close to the vertices in case they were unfeasible.

Our algorithm performs the usual set of operations carried out by evolutionary techniques, namely, *Mating Selection*, *Variation* and *Environmental Selection* (see Fig. 9.4). The reader is referred to [142] for a comprehensive description about these operations.

In our particular EMA, the Mating Selection is performed between the maximum feasible vertex V_L and the $(L+n)$ subsequent non feasible vertices,

where $0 \leq n \leq (K - L)$. The outcome of this mating produces individuals with similar characteristics to V_L, and also to V_{L+n} $\forall$ n.

The Variation corresponds to a mutation towards more expensive paths, which are likely to support the traffic coverage.

On the other hand, the Environmental Selection compares the candidates found and only the bests are kept in the archive A. With each new generation we obtain the *Best_Candidate*, which is determined as the candidate with the highest DM found up to that moment. The iso-objective lines through the successive *Best_Candidates* found along the evolutionary process are the basis for steering the search towards better solutions in the partitioned objective space. The details of the overall evolutionary process are described in Algorithm 6.

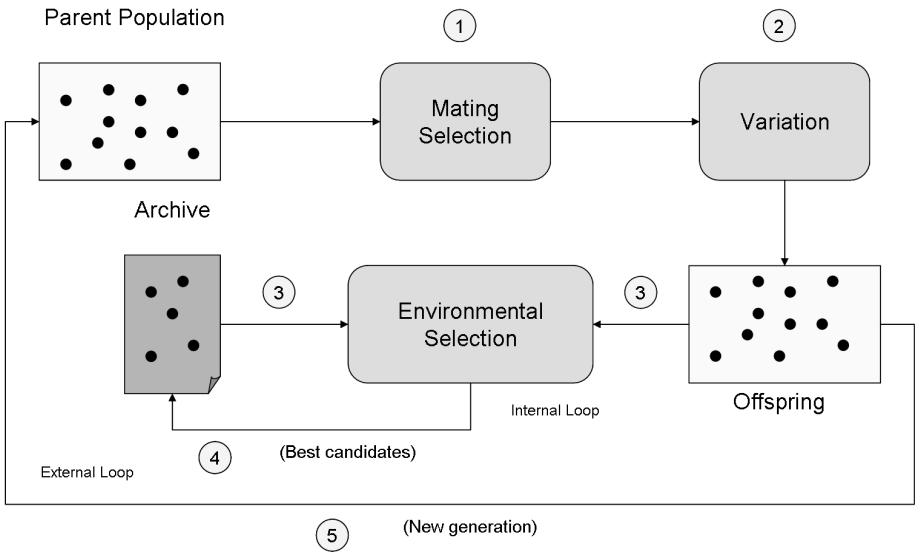


Figure 9.4: Evolutionary process.

The next section presents the main properties of MC^2. In particular, we prove that MC^2 guarantees the elitism of the candidates kept in the archive monotonically. We also prove that the search and update engine of MC^2 converges to an ε-Pareto Set.

Algorithm 6 EMA MC^2($G(S, D_i)$, $c(P_i^j)$, d_i)

Input: $G(S, D_i)$ - a graph with all the feasible paths between S and $D_i \; \forall \; i = 1, \ldots, K$
$c(P_i^j)$ - cost of the paths between S and $D_i \; \forall \; i, j$
d_i - Traffic volume to be covered for domain $D_i \; \forall \; i$

Output: The set of paths chosen, the covered demand, and its cost: $\{\delta_i^j, \sum\sum \delta_i^j d_i, \sum\sum \delta_i^j c(P_i^j)\}$

Generate an empty archive A and the initial population
Search for the maximum feasible vertex V_L and $\{V_{L+1}, \ldots, V_K\}$
Best_Candidate $\leftarrow V_L$
Compute iso-objective line through *Best_Candidate*
for each new generation g until stop condition **do**
  Perform Mating Selection
  Perform Variation
  /* Perform Environmental Selection */
  for each individual o in offspring O **do**
    if $DM(o) <$ iso-objective through *Best_Candidate* **then**
      discard o
    else
      if $\exists$ individuals $a \in$ archive A / $o \succ a$ **then**
        replace *all* dominated individuals a by o
      else
        add o to A /* since $o \not\prec a \; \forall \; a$ in archive A */
      end if
      Best_Candidate $\leftarrow o$
      Compute iso-objective line through *Best_Candidate*
    end if
  end for
  /* End of Environmental Selection */
  $g \leftarrow O$ /* Update generation */
  Update stop condition
end for

9.5.2 Elitism and Convergence of MC^2

The goals here are to prove that:

(i) The search and update strategy proposed in MC^2 guarantees the elitism of the individuals stored in the archive monotonically.

(ii) MC^2 converges to an ε-Pareto Set of $Y^{(t_f)}$, where $Y^{(t_f)}$ denotes the state of the *objective space* at the final instant t_f: $Y^{(t_f)} = \bigcup_{t=0}^{t=t_f} P^{(t)}$ with $P^{(t)} \in Y^{(t)}$

In the following we formally state and prove each of these items.

Elitism

Each new generation g of algorithm MC^2 produces several individuals – some of which will encode potential solutions that need to be added to archive A. A fundamental property to guarantee the convergence of an EMA is that the environmental selection process ensures the non-deterioration of the individuals kept in the archive.

The "deterioration" of the individuals in archive A might appear in two different ways along the evolutionary process. One way is that an individual $o^{(t_2)}$ contained in the archive at generation $g(t_2)$ (i.e., $o^{(t_2)} \in A^{(t_2)}$) may be dominated by a former member $o^{(t_1)}$ that was contained in the archive at generation $g(t_1)$, with $t_1 < t_2$, but was discarded at some instant t between $t_1 < t \leq t_2$. The other way of deterioration is that $o^{(t_2)} \in A^{(t_2)}$, but $o^{(t_2)} \notin$ Pareto Set of $Y^{(t_2)}$. We now proceed to prove that the archiving strategy proposed in MC^2 does not suffer from this deterioration problem.

Theorem 9.5.1. *The archiving strategy of MC^2 guarantees the elitism of the individuals kept in $A^{(t)}$ monotonically.*

Proof. (Theorem 9.5.1)

By way of contradiction, let us assume that MC^2 deteriorates the candidate solutions kept in $A^{(t)}$. As described above this could only occur if:

Case 1) $o^{(t_1)} \in A^{(t_1)}$, $o^{(t_1)} \notin A^{(t_2)}$ with $t_1 < t_2$, and there $\exists\, o^{(t_2)} \in A^{(t_2)}$ /

$$o^{(t_1)} \succ o^{(t_2)} \tag{9.16}$$

Case 2) $\exists\ o^{(t_2)} \in A^{(t_2)}\ /\ o^{(t_2)} \notin$ Pareto Set of $Y^{(t_2)}$.

In **Case 1)**, if $o^{(t_1)} \notin A^{(t_2)} \Rightarrow o^{(t_1)}$ was replaced by another individual at some instant t between $t_1 < t \leq t_2$ (step 14 of Algorithm MC^2). This means that at time t there $\exists\ o^{(t)}\ /\ o^{(t)} \succ o^{(t_1)}$, and $o^{(t)}$ was inserted in $A^{(t)}$ in place of $o^{(t_1)}$. If $o^{(t)}$ is still in the archive at time $t_2 \Rightarrow o^{(t)} \succ o^{(t_1)} \succ o^{(t_2)} \Rightarrow o^{(t)} \succ o^{(t_2)}$. This implies that at time t_2 there are two members of the archive $o^{(t)}$ and $o^{(t_2)}$, where the former dominates the latter. This, however, contradicts either steps 11 or 14 of Algorithm MC^2, because:

(*) if $o^{(t)}$ was in the archive before $o^{(t_2)}$, then $o^{(t_2)}$ could have never been added to A, since the iso-objective $DM(Best\_Candidate) \geq DM(o^{(t)}) > DM(o^{(t_2)})$, given that $o^{(t)} \succ o^{(t_2)}$ (step 11).

(*) if $o^{(t_2)}$ was in the archive before $o^{(t)}$, then $o^{(t_2)}$ could not be in the archive at time t_2, since it must have been replaced by $o^{(t)}$ at time $t \leq t_2$ (step 14).

Now, if on the other hand, $o^{(t)}$ is not in the archive at time t_2, this implies that there must be at least one individual $o^{(t')}$ in $A^{(t_2)}$, such that $o^{(t')} \succ o^{(t)}$, and which replaced whether $o^{(t)}$ or an individual that dominated $o^{(t)}$ at some instant $t' \leq t_2$ (step 14 of Algorithm MC^2). Once again, we end up with two members of the archive at time t_2, $o^{(t')}$ and $o^{(t_2)}$, such that $o^{(t')} \succ o^{(t_2)}$. Using the same reasoning as above, this contradicts either steps 11 or 14 of Algorithm MC^2. This concludes the proof of Case 1).

The proof of **Case 2)** is trivial, since there are only two possibilities for the dominance relationships between $o^{(t_2)} \in A^{(t_2)}$ and other members of the archive at time t_2. Either there $\exists\ o'^{(t_2)} \in A^{(t_2)}\ /\ o'^{(t_2)} \succ o^{(t_2)}$, or $\nexists\ o\ /\ o \succ o^{(t_2)}\ \forall\ o \in A^{(t_2)}$ with $o \neq o^{(t_2)}$. If the fist possibility holds, then by the same reasoning as above $o^{(t_2)}$ should not be part of the archive at time t_2. If on the other hand, the second possibility holds, the assumption that $o^{(t_2)} \notin$ Pareto Set of $Y^{(t_2)}$ is contradicted.

■

A corollary of Theorem 9.5.1 is the strict dominance between the current and former members of the archive.

Corollary 9.5.2. *If there* $\exists\ o^{(t_1)} \in A^{(t_1)}\ /\ o^{(t_1)} \notin A^{(t_2)}$ *with* $t_1 < t_2$, *then there* $\exists\ o^{(t_2)} \in A^{(t_2)}\ /\ o^{(t_2)} \succ o^{(t_1)}$

Proof. (Corollary 9.5.2)

The proof follows the same lines as that of Theorem 9.5.1. ■

An important feature of the archiving strategy proposed in MC^2 is that it does not introduce archive truncation [142]. Furthermore, only non-dominated solutions encoding a feasible coverage, and that are either within or above the iso-objective line through the *Best_Candidate* at any given moment can be added to the archive. These features not only ensure the non-deterioration of the quality of the solutions kept in A, but also guarantee that points of interest in the objective space are never missed along the evolutionary process (MC^2 does not cause *unreachability* of candidate solutions, see [51, 142]).

An archive truncation method is not needed due to the efficient search and update strategy of MC^2, which is based on iteratively computing iso-objective lines and finding individuals above it. The approach is to keep a rather small set of individuals in A in each iteration. These individuals are always close to the vertices $\{V_L, \ldots, V_K\}$, which is the locus where the best candidates can be found.

Convergence

The convergence of MC^2 can be proved after Theorem 9.5.1 as follows.

Theorem 9.5.3. *MC^2 converges at time t_f to the ε-Pareto Set of $Y^{(t_f)}$, where $Y^{(t_f)} = \bigcup_{t=0}^{t=t_f} P^{(t)}$, $P^{(t)} \in Y^{(t)}$*

Proof. (Theorem 9.5.3)

From Theorem 9.5.1 it is clear that all members of $A^{(t_f)}$ are mutually non-dominated, and they will be assessed by the decision maker better than any other individual found in $t < t_f$. From Case 2) in the proof of Theorem 9.5.1, it can be inferred that all the individuals in the archive at time t_f belong to $X^{*(t_f)}$, i.e., the Pareto Set of $Y^{(t_f)}$.

Let $P^{A^{(t_f)}}$ denote the points in the objective space $Y^{(t_f)}$ encoding the solutions present in the archive at time t_f. For each $P^{A^{(t_f)}} \in Y^{(t_f)}$ there $\exists$ a vertex V_L such that $V_L \succeq P^{A^{(t_f)}}$, $L \in \{1, \ldots, K\}$, and this holds independently of whether V_L is feasible or not. Moreover, for each $P^{A^{(t_f)}} \in Y^{(t_f)}$ there $\exists$ a vertex V_L such that:

$$Eucl(P^{A^{(t_f)}}, V_L) = min\left\{ \|V_L - P^{A^{(t_f)}}\|_2 \text{ s.t. } \overrightarrow{V_L P}^{A^{(t_f)}} \subset \bigcup_L Q_L \right\} \quad (9.17)$$

i.e., the Euclidean distance between $P^{A^{(t_f)}}$ and V_L is minimum (subject to the fact that all points of vector $\overrightarrow{V_L P}^{A^{(t_f)}}$ should be inside the area delimited by the union of the Q_L boxes in the partitioned objective space Y). Then, defining:

$$\varepsilon \triangleq max\{Eucl(P^{A^{(t_f)}}, V_L)\} \quad (9.18)$$

it is easy to show that $A^{(t_f)}$ is an ε-Pareto Set of $Y^{(t_f)}$, since:

(*) the members of the archive $A^{(t_f)}$ belong to the Pareto Set of $Y^{(t_f)}$

(*) the members of the archive $A^{(t_f)}$ encode solutions that ε-dominate the Pareto Front of $Y^{(t_f)}$, given that all the solutions should be located inside $\bigcup_L Q_L$.

■

9.5.3 SPEA2

SPEA2 [143] is an improved version of SPEA (Zitzler and Thiele (1999) [144]). The strengths of SPEA2 lie in its elitism preservation, its fitness assignment, its density preservation technique, and as MC^2, it uses an external archive where non-dominated solutions are stored. SPEA – and particularly SPEA2 – has become a reference EMA as it is being actively used to solve large multi-objective problems in many different fields, including medicine, engineering, etc [53, 72, 75, 89].

The pseudo-code of SPEA2 is shown in Algorithm 7[1]. The performance of our EMA algorithm (MC^2) is contrasted against SPEA2 in the next section.

9.6 Performance evaluation

In order to validate the performance of the EMA MC^2 proposed in this chapter, we performed a series of simulations using the PAN European network

[1]Extracted from [142].

Algorithm 7 SPEA2 Main Loop

Input: M (Offspring population size)
N (Archive size)
T (Maximum number of generations)

Output: A^* (Non-dominated set)

1: Initialization: Generate an initial population P_0 and create the empty archive (external set) $A_0 = \emptyset$. Set $t = 0$.
2: Fitness assignment: Calculate fitness values of individuals in P_t and A_t.
3: Environmental selection: Copy all nondominated individuals in P_t and A_t to A_{t+1}. If size of A_{t+1} exceeds N then reduce A_{t+1} by means of the truncation operator, otherwise if size of A_{t+1} is less than N then fill A_{t+1} with dominated individuals in P_t and A_t.
4: Termination: If $t \geq T$ or another stopping criterion is satisfied then set A^* to the set of decision vectors represented by the nondominated individuals in A_{t+1}. Stop.
5: Mating selection: Perform binary tournament selection with replacement on A_{t+1} in order to fill the mating pool.
6: Variation: Apply recombination and mutation operators to the mating pool and set P_{t+1} to the resulting population. Increment generation counter ($t = t + 1$) and go to Step 2.

[56], which is composed by 28 nodes and 41 links (see Fig. 9.5). We considered 1 domain as the source S, 7 transit domains and 20 destination domains.

The simulations are divided into three different sets: i) variable traffic demands; ii) variable network capacity; and iii) variable budget B. The results that we show here are the outcome of 100 simulation rounds for each of the three different sets of evaluations. To be precise, we run: i) 100 simulations by randomly changing the traffic demands d_i to be covered for each of the 20 destination domains; ii) 100 simulations by randomly changing the network capacity; and iii) another set of 100 simulations by progressively increasing the budget constraint.

The trials are aimed at providing supporting evidence of the quality of the solutions that MC^2 is capable of finding. To this end, we contrast the performance of MC^2 against the powerful and widely used EMA SPEA2 [143]. The simulations have a preconfigured limit of 3000 generations for both MC^2 and SPEA2, and SPEA2 is set up with the usual archive size of 100 individuals.

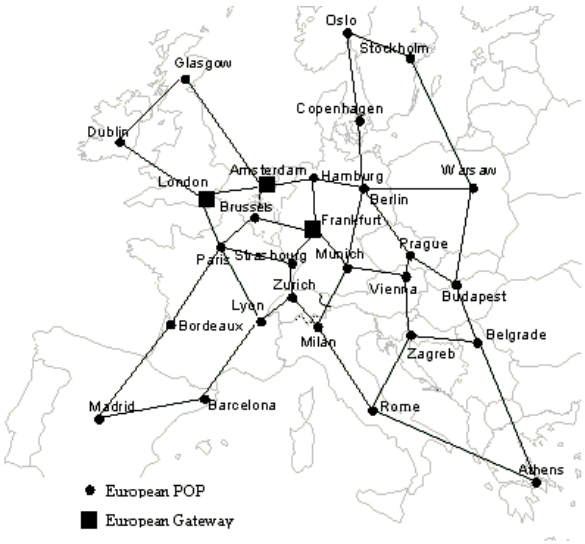


Figure 9.5: PAN European network.

Our results show that MC^2 performs much better than SPEA2 for all the conditions tested. All the results provided here can be reproduced, as the source code of MC^2 used during the tests is available from [84]. In the sequel we analyze the main results of our tests.

Provided that the decision maker will ultimately decide based on the value DM, the quality of the solutions obtained by MC^2 and SPEA2 are contrasted according to this value. Recall that the higher the value of DM, the better is the quality of the solution obtained. In addition, we show how the quality of the different solutions obtained maps to costs and the coverage of traffic demands.

Figure 9.6 shows the Cumulative Distribution Function (CDF) of the relative difference between MC^2 and SPEA2 for the 100 rounds of demand and capacity variation tests. The reference for building the CDF is MC^2 in both cases. The results clearly show that MC^2 outperforms SPEA2 in both sets of trials.

For the demand variation, MC^2 obtains at least a 5% of improvement in 80% of the cases, at least a 10% of improvement in 30% of the cases, and at least 15% of improvement for 10% of the cases. For the capacity variation the tests show that for 50% of the cases, MC^2 obtains an improvement of at least 20%, with a maximum improvement of 28%. For 34% of the cases, both algorithms obtain the same values.

Table 9.2 helps to understand the impact of the relative improvement in the decision value DM. The table shows the mean and standard deviation

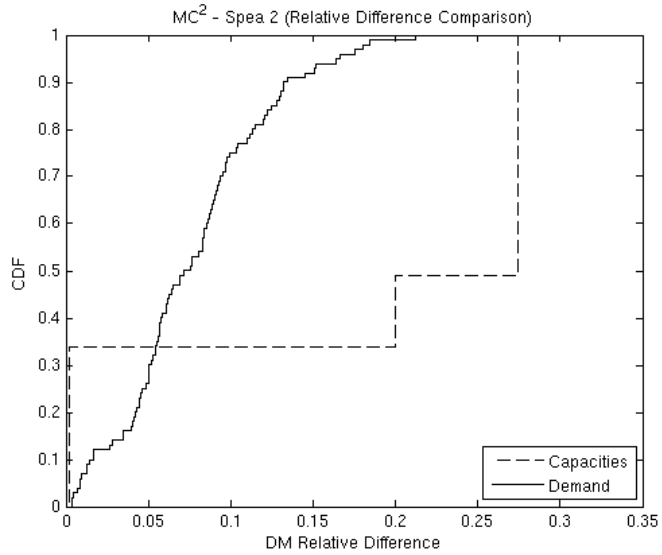


Figure 9.6: DM - Relative Difference comparison.

values of DM, the coverage and its associated cost. Clearly, MC^2 is able to find more efficient solutions with comparable costs. For example, MC^2 is able to pass from covering 55% to 80% of the demand, with just a 2% increase in the overall cost for the trials performed in the capacity case.

Table 9.2 shows that the costs of the best solutions found (i.e. the optimum DM) usually imply spending around half of the total budget. For the set of tested traffic demands and network capacity constraints, an investment of half of the budget is where the best trade-off – hence the maximum revenue – is found.

The last study performed to validate the behavior of MC^2 involves changing the maximum budget with a constant increase of 1 unit per test. The budget units are left generic on purpose, in order to draw attention to the fact that its usage may be generalized to any currency or budget schema used by a service provider.

Figure 9.7 shows the comparison between both algorithms. MC^2 presents a smooth increase of the DM value, which is due to the fact that it is always able to locate the best candidates in the partitioned objective space. Clearly, the DM is limited by the budget at the beginning (recall that values less than $\frac{1}{2}$ are of no practical interest), but when enough budget is available, the network capacity is what actually limits reaching higher values of DM.

	Mean DM		**STD** DM	
	Demand	Capacity	Demand	Capacity
SPEA2	0.5115	0.5165	0.015	0.020
MC^2	0.5485	0.6260	0.015	0.065
	Mean Coverage		**STD Coverage**	
	Demand	Capacity	Demand	Capacity
SPEA2	43%	55%	0.05	0.05
MC^2	52%	80%	0.04	0.15
	Mean Cost		**STD Cost**	
	Demand	Capacity	Demand	Capacity
SPEA2	51%	53%	$\simeq 0$	$\simeq 0$
MC^2	53%	55%	0.01	$\simeq 0$

Table 9.2: Mean and STD for DM, covered demand and cost.

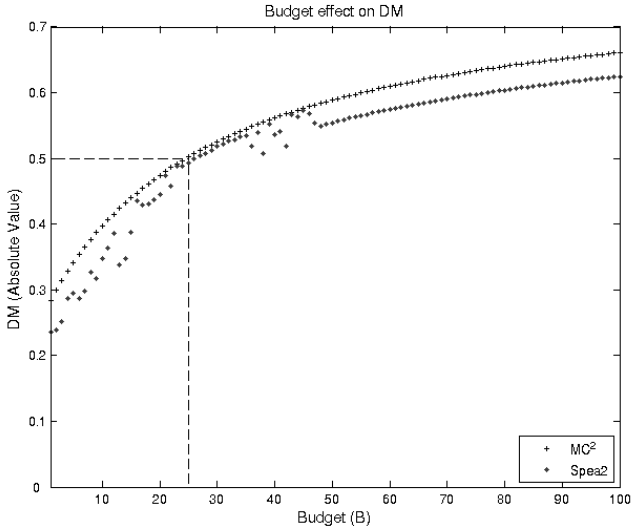


Figure 9.7: Comparison of SPEA2 and MC^2 for different budgets.

Figure 9.7 clearly shows that a minimal budget is always required to have a feasible solution, which of course depends on the maximum demand of the whole system and the reliability offered by S' ISPs. Our simulations show that investing less that 25 units of budget will not provide the minimum revenue worth the effort, given that $DM < \frac{1}{2}$.

9.7 Conclusions on IP/MPLS multi-domain networks

In this part of the thesis we have reviewed the advantages of extending the reach of MPLS paths beyond domain boundaries, especially, in order to improve the performance and reliability of both public and private inter-domain communications. This is quite likely the next-step and hence the near future of inter-domain routing and TE control.

We have also deepen into the analysis of the existing limitations hindering the deployment of MPLS at the inter-domain level, and discussed about ways to solve them. In particular, we have examined the strengths and the possibilities offered by the IETF's PCE-based model. By taking advantage of this model, we have broadening the IETF's conception of the PCE and proposed a distributed routing algorithm for optimally finding two disjoint (primary and backup) QoS paths across multiple IP/MPLS domains. This algorithm was devised for a PCE-based architecture that can work completely decoupled from BGP.

The algorithm exploits an aggregated representation of a multi-domain network that captures the path diversity and the link-state characteristics of transit paths that run across different routing domains. This representation is used by the distributed routing algorithm, allowing each PCE to optimally compute two disjoint QoS paths for any source-destination pair in the multi-domain IP/MPLS network. The two disjoint paths problem was formulated and solved both for a general multi-domain network setting, as well as subject to the export policies imposed by customer-provider and peer relationships between routing domains. The algorithms proposed can be used in many practical settings, in particular, when high-quality primary and backup QoS LSPs need to be established across a reduced set of neighboring domains.

Then, by exploiting this distributed routing model between PCEs we have formulated and efficiently solved the problem of how a domain can maximize the MPLS coverage of traffic demands with minimum cost, subject to a budget and network capacity constraints. The problem was formulated as a multi-objective integer program, and solved by means of an evolutionary algorithm named *Maximum Coverage at minimum Cost* (MC^2) that we proposed and tested in this chapter. Our most important contribution in this subject is the search and update engine of MC^2, which allows to steer the search of potential solutions in a fast and highly efficient way. The key is the way in which we exploit the knowledge about the concavity of common pricing functions of ISPs.

The most promising outcome of this part of our work is that the con-

tributions are general in scope and can be applied in other problems. In particular, our proposals can be applied in settings where constrained problems considering maximum coverage vs. cost are critical, given that costs associated with concave pricing functions are widely used in practice.

Part IV

Route Control: Future

Chapter 10

Multi-Domain Optical Networks

Future optical networks need to be prepared to efficiently handle the expected changes in the provisioning model of transport services. Among the most important changes expected in the long-term are the following:

(i) Customers must no longer be limited to purchase monthly or yearly contracts for one-time capacity. Instead, Network Service Providers (NSPs) should be able to offer end-to-end optical connections with the capacity required for periods of months, weeks, days, hours, or even minutes (e.g., for resilience reasons).

(ii) Customers must be able to acquire or release end-to-end optical connections on demand and in real-time, so the set up (tear-down) of those lightpaths should be solved dynamically.

(iii) Customers must be able to control the path selection and set up processes of their optical connections, according to their specific needs in terms of performance and reliability.

To address these requirements, the International Telecommunications Union (ITU) [59] has developed a network model that offers the automatic delivery of optical transport services, including the establishment of switched end-to-end optical connections. This model is referred to as *Automatically Switched Optical Network* (ASON) [62]. In an ASON, optical connections can be set up and released on demand directly by end-customers, using appropriate signaling and routing protocols (see ITU-T[1] Recommendations G.7713.1, G.7713.2, G.7713.3, and G.7715). Each node in the ASON model is equipped with a *Control Plane*, which is responsible for the establishment and release of optical connections. The ITU-T Recommendation G.8080 [62] describes

[1]ITU-T is the specific "Telecommunications" standardization sector of ITU.

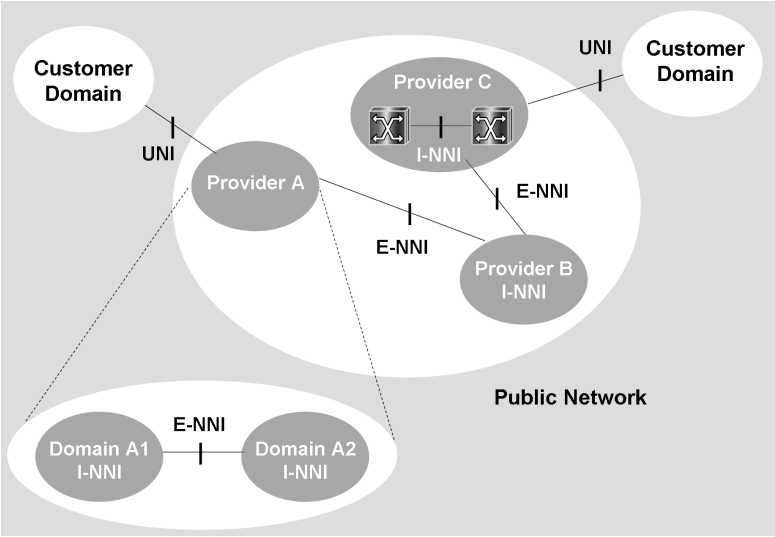


Figure 10.1: The ASON model.

the architecture of the ASON model, including the components of the distributed control plane that handle the dynamic discovery, routing, set up, and release of end-to-end optical connections.

Figure 10.1 depicts the most basic concepts behind the ASON model. The figure shows a multi-domain scenario with three different providers, namely, providers A, B, and C, and two customer networks. Provider's A network is divided into two independent *Routing Control Domains* (RCDs) A1 and A2 (e.g. due to geographic, vendor incompatibilities, or policy reasons). Providers B and C on the other hand, are represented as a single RCD.

Figure 10.1 also shows three different types of standardized interfaces, which are referred to as *reference points*. The User Network Interface (UNI) supports the communication and signaling operations between the customer's and provider's optical networks. The Internal Network-Network Interface (I-NNI) supports the communication and signaling operations between different devices inside the same RCD. Finally, the External Network-Network Interface (E-NNI) supports the communication and signaling operations either between RCDs within a provider (like in the case of provider A) or between different providers. The visibility of the internal structure and resources within each RCD is controlled by the policies ruling the exchange of information through the different Network Interfaces (NI) of the RCDs.

The distributed control plane of an ASON will determine the characteristics and the set of capabilities that a multi-domain optical network will be endowed with in order to support the automatic delivery of inter-domain

connections. More specifically, the role of the control plane is key to support:

(a) Interworking between different RCD, and particularly, between different providers (this is usually referred to as inter-carrier interworking).

(b) The auto-discovery of nodes and networks at the inter-domain level.

(c) End-to-end lightpath provisioning, including the dynamic set-up and release of inter-domain connections.

(d) Automatically switched connections within both public and private multi-domain networks.

(e) Guaranteed security for control plane related tasks.

(f) Guaranteed traffic performance and reliability across multiple domains. In particular:

- QoSR, i.e., the capability to find and select inter-domain lightpaths subject to performance constraints. To this end, the routing process needs to learn and disseminate specific path state information between different RCDs.

- Quality of Resilience (QoR), i.e. fast rerouting and guaranteed service restoration of inter-domain connections upon a link or a node failure. To accomplish this, a RCD must be able to establish primary and backup disjoint paths – that could be subject to QoS constraints – and learn about the resilience offer of neighboring RCDs.

Clearly, the success of a distributed control plane model will strongly depend on the standardization efforts and the momentum gained during its development. At present, three standardization bodies are independently working in the subject: ITU [59], the IETF [60], and the Optical Internetworking Forum (OIF) [90]. As mentioned above, ITU-T has developed ASON together with a set of recommendations regarding its architecture, its management aspects, the control plane requirements, etc. On the other hand, the IETF's WG named *Common Control and Management Plane* (CCAMP) [24], is leading the development and standardization efforts around the *Generalized Multiprotocol Label Switching* (GMPLS) suite of protocols. GMPLS is an extension of MPLS that supports the establishment of more general LSPs – referred to as Generalized LSPs (G-LSPs) – including the automatic set up and release of optical connections. Unlike the ITU's ASON model, which basically defines and architecture and a set of recommendations around it, the aim of the IETF's CCAMP WG is to standardize a suite of IP-based

protocols supporting the advance and deployment of GMPLS. At present, GMPLS arises as a strong candidate to become the control plane of future optical networks. Finally, the role of the OIF is mainly to achieve implementation agreements between vendors, and develop standards upon them. For example, the OIF has recently released the "E-NNI OSPF-based Routing - 1.0 (Intra-Carrier) Implementation Agreement" (available from [90]). Part of the work of the OIF has also served to fill the gap between the ITU's ASON requirements and the IETF's GMPLS-based control plane.

An important point, however, is that most of the work done so far by these bodies addresses the intra-domain aspects of future optical networks. The discussions concerning multi-domain issues are in a very early stage yet, so despite some of the topics listed above (from (a)–(f)) have started to be analyzed by the standardization bodies (see for example [35, 125, 120]), the situation is that the majority of them are largely open at present.

From this wide open set of topics, we focus in this part of the thesis on proposing solutions in the areas of routing and TE control in multi-domain optical networks, as they represent effective tools to tackle the last of the topics listed above (f). In particular, in Chapter 11 we present a multi-domain optical network model that extends in several aspects the concepts coming from the IETF. The reach of this model includes a network architecture and the details about the information exchange between RCDs. Then, in Chapter 12 we propose and contrast three distributed route control strategies, two of which fully exploit the multi-domain network model proposed in Chapter 11, while the other offers our improved version of the optical extension of BGP [16, 39, 130, 131], which we named OBGP+.

In the remainder of this chapter we argue in favor of taking advantage of the opportunity to change that future optical networks offer, and particularly discuss about the adequate steps towards a route control model for an optical Internet.

10.1 The opportunity to change

As described in Sections 3.2, 3.3, and 7.2, improving the performance and reliability of inter-domain communications in the context of the current routing and TE control model, represents a complex problem given the limitations imposed by BGP. For example, tools like QoSR have been recognized as a missing piece in the inter-domain routing model [26]. Indeed, QoSR, is becoming a strong requirement in the current Internet, and it is quite likely that this requirement will also be present in the next generation optical Internet. Accordingly, it is widely accepted now that in addition to reachability infor-

mation, neighboring domains should become capable of exchanging highly aggregated and useful path state information.

Despite the well-known limitations of BGP in the areas of QoSR and TE control, during the past few years some researchers have proposed to adopt an *Optical Border Gateway Protocol* (OBGP) as the future inter-domain routing protocol for optical networks [16, 39, 130, 131]. The aim of these proposals is to extend BGP so that it can convey and signal optical path information between OBGP neighbors. The strength of this approach is that future optical networks will benefit from the well-known advantages of the BGP-based routing model, such as scalability, clear administrative limits of routing domains, fully-distributed network administration based on filtering and routing policies, etc. The weakness, on the other hand, is that the routing model of future optical networks will inheritate the well-known issues in BGP [138]. Indeed, a multi-domain routing model mostly centered on the exchange of reachability information – like the one we have today or the one offered by OBGP – is not going to be enough. This is confirmed by a number of research initiatives recently started, like [40] and [41]. It is worth highlighting that even though OBGP has not yet found support in the IETF community[2], there are still ongoing research efforts in the subject [130, 131].

An alternative inter-domain routing approach appeared in 2002 when the OIF proposed the Domain-to-Domain Routing Protocol (DDRP) [13]. DDRP is basically a hierarchical extension of OSPF-TE, supported by a modified version of Dijkstra's algorithm – to avoid the scalability limitations of using a link state protocol like OSPF-TE at the inter-domain level. However, DDRP has mainly two drawbacks. First, it represents a major change in terms of routing and service provisioning compared with the current IP-based Internet, since it proposes to move towards a fully hierarchical model. Second, the modified Dijkstra algorithm still offers very limited flexibility and functionality in the areas of inter-domain QoSR and TE. For instance, the modified algorithm returns a single optimal path at a time so complementary algorithms need to be adopted for diverse routing computation and path protection.

With a different approach, the standardization efforts being carried out at the IETF mention the need to work on new protocols, or extensions to the existing ones, in order to enable the advertisement of inter-domain TE information. In [35], the authors even mention the possibility of adding TE extensions to BGP. This is a reasonable – though certainly not optimal – option in the context of inter-domain IP/MPLS networks. We claim, however, that there is no need that future optical networks inheritate the well-known

[2]The last IETF drafts regarding OBGP are of 2001 [16].

limitations of the BGP-based model.

Neither BGP/OBGP nor DDRP will be able to provide the expected functionalities in the area of inter-domain route control for an optical Internet. Future on-demand lightpath provisioning networks will natively demand QoSR, QoR, and enhanced TE control between RCDs. This has leveraged the proposals of different network state aggregation schemes and updating policies at the inter-domain level for wavelength-routed optical networks [74, 141].

Given that inter-domain routing and TE control are becoming active research areas in optical networks, it seems sound to address these issues from their very foundations. It is necessary to investigate how to endow the control plane of future optical networks with the ability to compute and efficiently convey aggregated path state information between domains. We argue in favor of a change, with emphasis in adopting the best of BGP while avoiding the worst of it.

10.2 Towards a new route control model

At present, there are essentially two route control models for optical networks, namely, the *Overlay* and the *Peer* models.

Overlay Model – This model calls for maintaining separate networks or RCDs. Each RCD can run internally whatever set of protocols that best suits its needs, and they do not need to be compatible with those running on other RCDs. The compatibility in the communication with other RCDs is achieved through the standardized reference points shown in Fig. 10.1. The I-NNIs and E-NNIs are the control interfaces over which the optical network connections are accomplished, involving basically lightpath routing and signaling. Devices inside a RCD use the I-NNIs to exchange critical information within their network, and E-NNIs to exchange information with other RCDs.

For a provider, an external RCD is either a customer or another provider. To request capacity from the network, customers access the optical network through the UNI reference point (see Fig. 10.2), which completely hides the internal routing and the state of resources in the provider's RCD from the customer. The key in this model is that customers' devices cannot "see" inside the NSPs' networks. Despite this, the UNI allows customers to establish optical connections dynamically across the optical network, using neighbor-discovery and service-discovery mechanisms. Customers' nodes can request the NSP for an optical connection, and the NSP can either grant it or deny

it. The customers' requests can be quite sophisticated, asking for example for a certain circuit size with a particular grade of restoration.

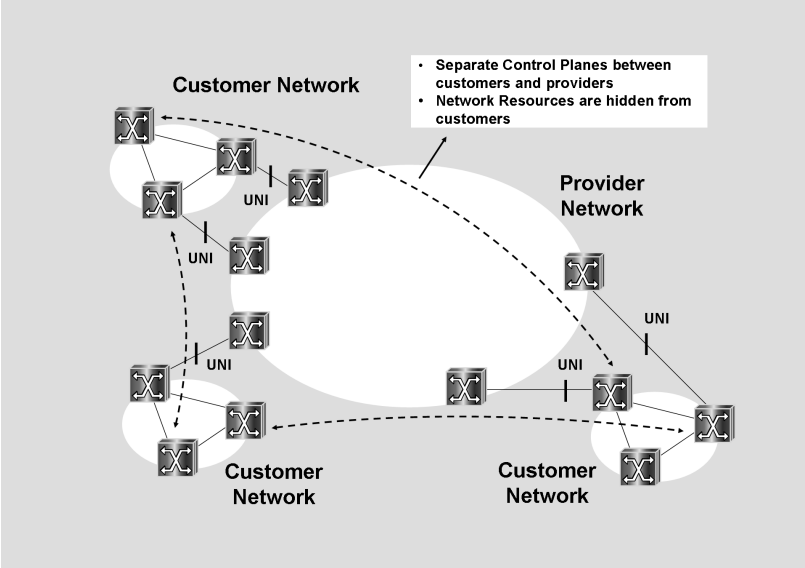


Figure 10.2: Overlay architecture for the intra-domain case.

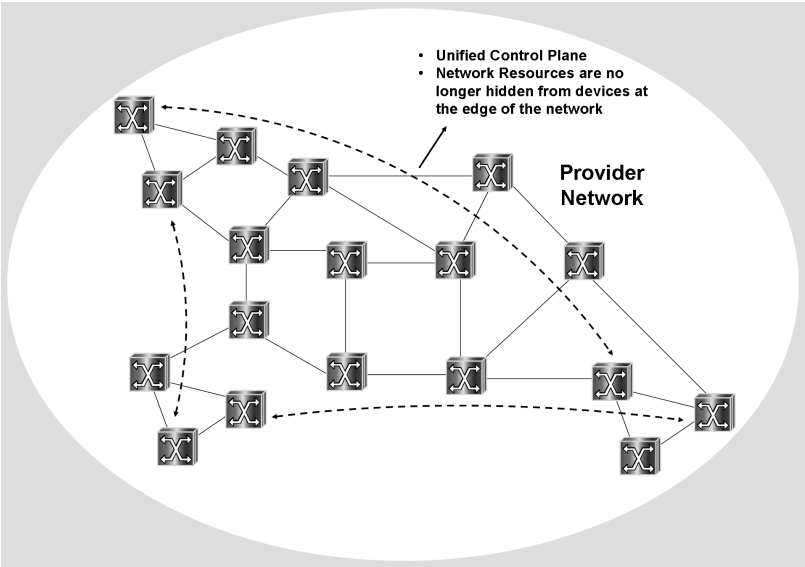


Figure 10.3: Peer architecture for the intra-domain case.

In the case of a provider-to-provider interface, some routing information is exchanged through the E-NNI reference points, so that a RCD receives reachability information about distant RCDs. The routing information exchanged is highly condensed, so as to avoid that internal details of a RCD can be inferred from its routing advertisements.

In the Overlay model, the assignment and management of network resources are firmly controlled by NSPs, so legacy voice operators tend to support this model. Figure 10.2 depicts the Overlay architecture and network model for the intra-domain case.

Peer Model – The Peer model argues for a single network (i.e. a unified control plane) in which devices at the edge of the network can decide how connections are routed and allocated along the core. The separation between the customer and provider networks becomes blurred. In fact, there is no such a concept of a "customer" in the Peer model. Figure 10.3 shows the same architecture of Fig. 10.2, but in the Peer model case.

In the Peer model, optical switches have complete knowledge of the topology and network resources inside the RCD. In this model all internal reference points are of type I-NNI. This model is tightly linked with the IETF's concept of a unified GMPLS control plane, supported by the IP protocol. An important issue in this model is that a significant amount of state and control information will flow between the IP and optical layers as the size of the network grows. Thus, for scalability, and also security and administrative considerations, the Peer model might be appropriate for a single RCD, but it is not applicable in a multi-domain environment.

In sum, while the Overlay model completely hides the path state information between RCDs – a missing piece today – the Peer model results inadequate in the inter-domain case. Clearly, a new model is needed.

Hybrid Inter-Domain Route Control Model – An Hybrid and enriched Overlay/Peer combination can provide the appropriate route control model for the future optical Internet. On the one hand, the NSP-to-NSP and customer-to-NSP relationships can be modeled similarly as in the Overlay case, but with two modifications. First, the Hybrid model is conceived for the inter-domain case, so the customer-to-NSP relationships require an E-NNI interface when the customer and the NSP are in different ASs. When the customer is within the AS of the provider, the usual UNI interface is used (see Fig. 10.4). Second, in the Hybrid model useful and highly aggregated TE information is exchanged between RCDs through the E-NNIs, and this applies whether in the case of a customer-to-NSP relationship or in the NSP-to-NSP case.

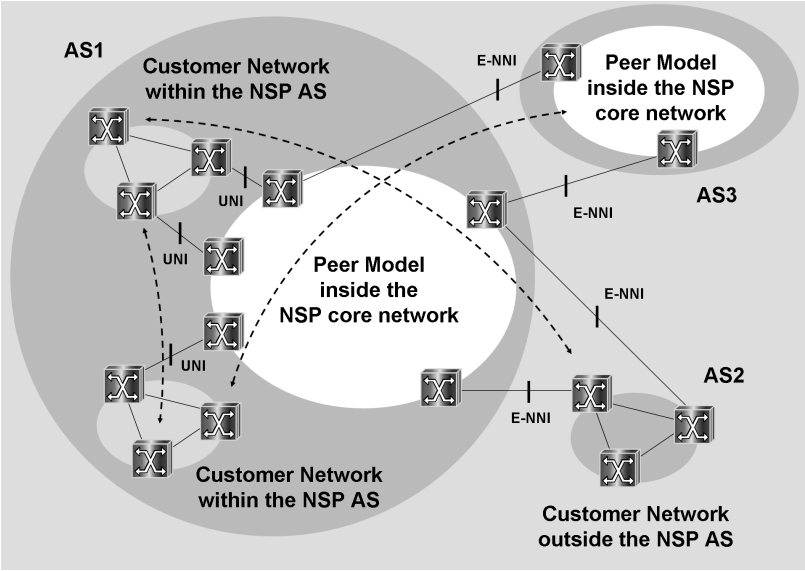


Figure 10.4: Hybrid route control model for the inter-domain case.

On the other hand, the Peer model supported by a standardized GMPLS control plane represents an appealing solution for the core network inside RCDs. The Peer model offers the chance to effectively exploit the internal connectivity of a RCD, providing substantial advantages in terms of resilience and traffic load balance across the RCD. Figure 10.4 shows the internal structure of AS1, where a Peer cloud provides transit to customer clouds connected through both UNIs and E-NNIs.

The Hybrid model arises as an attractive trade-off between both models, since it scales supporting multiple and administratively independent RCDs, while it leverages the deployment of a GMPLS control plane inside RCDs.

The Hybrid multi-domain network model that we conceive is supported by a fully distributed and decoupled control plane. The "decoupling" consists of clearly splitting the control and data planes by means of physically separated networks. Other highly scalable and successfull networks relied on this kind separation in the past, like the Signaling System #7 (SS7) used in telephony. In the future, having dedicated fibers and nodes to distribute routing and signaling information between RCDs is in fact desired. One of the biggest differences expected between current IP networks and future optical networks is precisely to pass from an in-band signaling approach (this is the case with BGP today), to an out-of-band signaling model.

The role of the nodes in the dedicated control plane of the Hybrid model is two-fold. As mentioned before, they are the ones that distribute the routing and signaling information between RCDs. In addition, they are in charge

of performing the path computation of the lightpaths in a distributed way – just as in the IETF's PCE case. The nodes in the Hybrid control plane can compute primary and backup lightpaths subject to QoS and/or QoR constraints, using the TE information received through the different reference points of their RCD.

The TE information exchanged between RCDs fulfills the following requirements:

(i) Path State Information (PSI) must be advertised between RCDs in addition to the usual reachability information.

(ii) The PSI received from downstream RCDs must be assembled and aggregated together with local PSI, and advertised to upstream RCDs.

(iii) This PSI flow must supply a standardized coupling between the different segments along a lightpath. This will support the computation of end-to-end optical paths in an efficient way.

(iv) The PSI exchanged must be completely independent of the intra-domain routing and signaling protocols. In this sense, enhancements or even a complete replacement of any of the protocols used inside a domain must not affect the routing and TE information exchange model between domains.

(v) Special care must be taken while developing aggregated PSI schemes, and while deciding the frequency of the updates associated with the information sent across domain boundaries.

These issues are the subject of study in the next chapter. Then, in Chapter 12 we shall blend this Hybrid model with the best of BGP to develop three distributed route control strategies for multi-domain optical networks.

Chapter 11

A New Route Control Model

The Hybrid route control model proposed in Chapter 10 can be supported by the introduction of a new network component, which we named *Inter-Domain Routing Agent* (IDRA). These IDRAs will act, among other things, as the glue between the intra- and the inter-domain routing schemes of RCDs. Our goals in this chapter are to describe an IDRA-based network architecture for the Hybrid model, and present the details about the routing and TE information exchange between the IDRAs.

11.1 Inter-Domain Routing Agents (IDRAs)

Each RCD may allocate one or more of these agents depending on its scale[1], which are the ones in charge of computing paths, and exchanging routing and TE advertisements between neighboring RCDs.

The Hybrid model clearly separates the control and data planes, so independent circuits are used to physically connect the IDRAs between each other. In this framework, a pair of neighboring IDRAs can establish two kinds of peering relationships: i) a customer-provider; or ii) a provider-to-provider relationship. When the IDRAs are in the same AS, the peering is referred to as *intra-AS peering*, and when they belong to different ASs the peering is named *inter-AS peering*. An example of a provider-to-provider intra-AS peering can be described using Fig. 10.1, assuming that provider A represents a single AS, and the RCDs A1 and A2 allocate the peering IDRAs. Another example is depicted in Fig. 11.1 between the IDRAs IDRA11 and IDRA12 in AS1. Figure 11.1 shows that AS1 is splitted into two RCDs, each of which is managed by one IDRA.

The customer-provider inter-AS case can be described by means of Fig. 10.4, when the peering IDRAs are located in AS1 and AS2. On the other hand, the provider-to-provider inter-AS case applies between all peering IDRAs in Fig. 11.1, except between IDRA11 and IDRA12.

[1]For scalability and reliability reasons clusters or even distributed clusters of IDRAs can be managed inside a RCD.

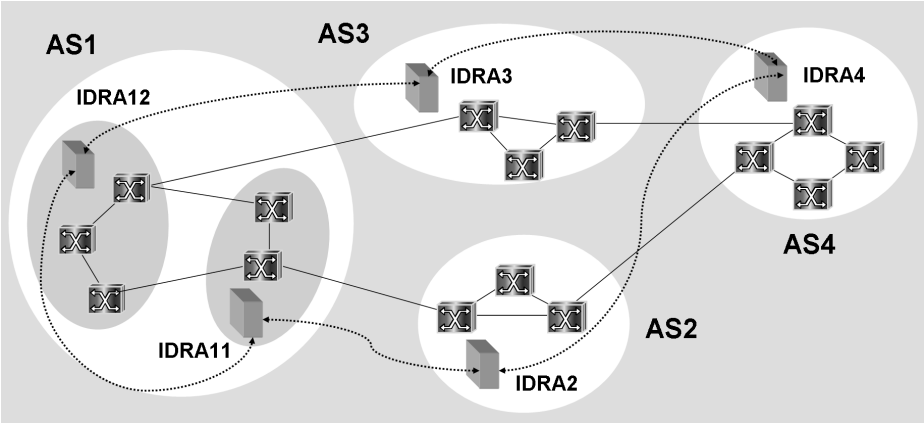


Figure 11.1: The IDRA-based multi-domain network architecture. For scalability and functionality reasons, the agents are decoupled from the optical nodes.

In all the peering relationships described above, the interface used for the exchange of routing and signaling information between the IDRAS is an E-NNI. The IDRAs are conceived as devices to control the routing and TE processes inside each of the various RCDs in which a large AS might be split, or even an entire AS. Customers' networks connected through UNIs are not considered here as "independent" RCDs[2], and hence are not in need of one of such agents. Figure 11.2 shows the inter-domain ligthpath set up steps within a RCD, when the path request is triggered from a customer network connected through a UNI. These steps can be summarized as follows:

(1) The customer requests a path to a Distribution Layer (DL) device.

(2) The DL device interrogates the local IDRA in order to find the best route towards the destination.

(3) The IDRA responds with the best path (the details about the RWA process running on the IDRAs will be developed in the next chapter).

(4) The DL device forwards an Explicit Path Setup Request to the corresponding local border node (l_{b2} in the example shown in Fig. 11.2).

[2]Such customers' networks are not an autonomous piece of the inter-domain route control model supported by the IDRAs, since from the inter-domain viewpoint they belong to the RCDs owned by their providers.

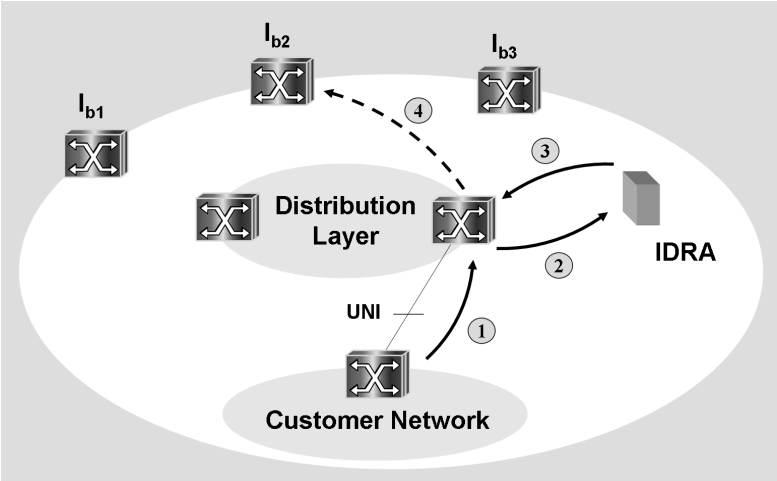


Figure 11.2: Lightpath computation and set up processes.

11.2 TE information exchange model

As mentioned before, the IDRAs are responsible for distributing inter-domain routing and TE information, and deciding within each RCD the best path to reach a destination. To this end, the advertisements distributed by the IDRAs contain the usual *Network Reachability Information* (NRI), in addition to TE information consisting of : i) *Path State Information* (PSI); and ii) the set of offered *services* by the RCDs along a path (see Fig. 11.3).

The role of the PSI is to capture the "state" of resources along an inter-domain path. During the composition of the advertisements, the IDRAs aggregate the PSI along a path taking into account the state of both the intra- and the inter-domain segments of the path. The advertised PSI is rich enough so that upstream RCDs can choose high quality inter-domain paths, and at the same time is sufficiently aggregated so that administrative limits and business protection considerations of RCDs are respected.

Figure 11.3 illustrates the flow of the advertisements between the IDRAs (from the destination domain AS_D towards the source domain AS_S). The $IDRA_i$ assembles the PSI received from $IDRA_{i+1}$ with its local PSI, and advertises $IDRA_{i-1}$ the aggregate: $PSI^{(i-1)} = PSI^{(i)} \oplus PSI^{(i+1)}$, where the operator $\oplus$ denotes an appropriate PSI assembling function. The data conveyed in the PSI as well as the strategy to update them are detailed in Section 11.3.2.

On the other hand, the role of the *services* is to endow the Hybrid model with the capability that RCDs exchange more complex TE data structures.

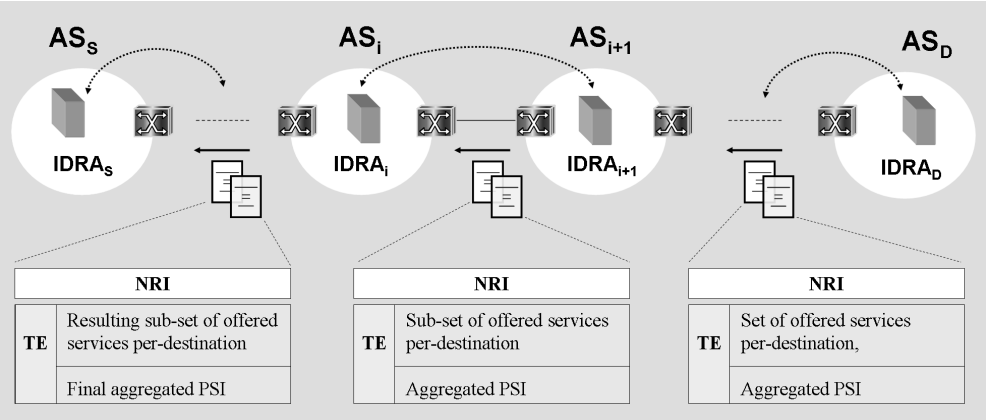


Figure 11.3: Information flow between the IDRAs.

For instance, a RCD could advertise that it offers wavelength conversion for a group of destinations, while it can offer multi-hop traffic grooming for another group[3]. We foresee that the TE service offer must have the ability to evolve in time, so the services portion of the IDRA-based model is wide open, admitting the replacement, the enhancement, or the incorporation of new TE services.

In this framework, the IDRAs are endowed with flexible input/output policy-based filtering capabilities so that each NSP may use and advertise to its neighbors the sub-set of services that it supports, and for which is willing to provide transit for. Figure 11.3 shows that the service offer received at any upstream RCD may be a condensed sub-set of the services advertised by AS_D, due to the filtering processes performed by the intermediate domains in the path. Such sub-set of services can be associated with particular destinations or entire ASs. For example, a NSP may advertise that it offers high path diversity for a particular set of destinations, while it offers poor or no diversity at all for the rest of the advertised destinations [14]. The Hybrid model is highly flexible in this particular point, given that a NSP may utilize just a single set of services associated with all the destinations advertised, or it may advertise different sets of services for different sets of destinations as in the example above.

The advertisement of TE services introduces several advantages when compared with the current inter-domain routing paradigm (we recall that

[3]Traffic grooming is a mechanism for multiplexing (demultiplexing) several traffic flows onto (from) a high-capacity lightpath. With grooming network resources are more efficiently used, since it avoids assigning the entire bandwidth of a wavelength channel to a low-capacity connection.

at present there are no mechanisms to advertise enriched TE information in a flexible and effective way). For instance, the RWA strategy configured in one IDRA could prefer – at some extent – a route with a better service protection over the route showing the best path state (e.g. the minimum-weight path) at the time of selection. Therefore, each IDRA is capable of determining the best path to reach any given destination whether: i) based on a number of specific requirements in terms of services; ii) based on the PSI along the candidate routes; or iii) based on a combination of them. This approach leverages the development of more sophisticated heuristics and QoSR algorithms devised to efficiently cope with the $\mathcal{NP}$-hard issues present in Multi-Constrained Path (MCP) selection problems [25, 67].

11.3 Information exchange between the IDRAs

This section describes in detail the content of the NRI and PSI exchanged between the IDRAs[4]. The distributed RWA algorithms that we shall present later in Chapter 12 efficiently exploit the information described in this section.

We assume that the optical nodes, namely, the *Optical Cross-Connects* (OXCs) do not perform wavelength conversion, so each lightpath computed by an IDRA is subject to the wavelength continuity constraint. We proceed now to describe the NRI and the aggregated PSI conveyed by the IDRAs in Fig. 11.3.

11.3.1 Network Reachability Information (NRI)

Let L, F, and Ω denote the number of links, the number of fibers per-link, and the number of wavelengths (colors) per-fiber, respectively, at each destination OXC. For the sake of simplicity we assume that all destination OXCs are identical, and that each network sinking traffic is connected to only one OXC[5]. Thus, $LF\Omega$ is an upper bound of the number of available wavelengths to reach any destination within a domain. Each AS may select – according to its local TE and routing policies – the particular subset of wavelengths that can be used by an upstream domain to reach the local

[4]The development of TE services is left for future work. Furthermore, the signaling details between the IDRAs are not described here, and are left for future work as well.

[5]The information exchange model described here can be easily generalized if these assumptions are not met.

networks. Consequently, the reachability information contained in the NRI messages sent by an IDRA consists of:

(i) The set of destination networks $\{d\}$ and their associated AS-path.

(ii) The Next-Hop (NH) to reach those destinations, i.e., the address of the ingress OXC in the RCD from which the advertisement was sent. It is worth noticing that the NH concept is basically the same as in the case of BGP. The only difference is that instead of advertising it "in-band" like occurs with BGP, it is advertised separately from the data plane through the dedicated control plane composed by the IDRAs.

(iii) A set of pairs $(\Lambda_1,\ M_{\Lambda_1}),\ \ldots,\ (\Lambda_N,\ M_{\Lambda_N})$ available for each destination d, where Λ_i, $i \in \{1, \ldots, N\}$ denotes a particular wavelength, and M_{Λ_i} denotes the maximum multiplicity of Λ_i. Clearly, $N \leq \Omega$, and $M_{\Lambda_i} \leq LF\ \forall\ i$.

Another important difference between BGP and the IDRA-based model in terms of NRI, is that instead of advertising only the "best" route for any given destination, the IDRAs can advertise more than one route for the same destination – even with the same NH. In sum, the NRI distributed between the IDRAs is composed by:

$$\Phi_{NRI}(d) = \left[d,\ NH(d),\ \left\{ (\Lambda_1, M_{\Lambda_1}), \ldots, (\Lambda_N, M_{\Lambda_N}) \right\}_d \right] \tag{11.1}$$

For each destination network, a transit AS may filter and advertise a subset of Φ_{NRI} to its upstream domains, or simply retransmit the NRI messages received. When a new destination network becomes available, or an already known one becomes unavailable, the NRI messages are triggered immediately by an IDRA. In any other case, the NRI should only change over large timescales compared to the PSI, according to the local optimizations and TE actions performed by the different RCDs.

It is worth highlighting that conversely to the NRI conveyed in BGP, $\Phi_{NRI}(d)$ does not include the RCD-path (i.e. the counterpart of the AS-path in the BGP case) to reach destination d. In the IDRA-based model, rather than comparing candidate routes according to the length of the RCD-path, the IDRAs exploit the TE information contained in the routing advertisements to compare the routes. An important point, however, is that

the performance of the route selection strategy used by the IDRAs will substantially depend on the length of the lightpaths chosen[6], so the path length should be taken into account during the RWA decision process. To this end, the length of the path is embedded in the PSI messages exchanged between the IDRAs.

11.3.2 Aggregated Path State Information (PSI)

The PSI is composed by *aggregated wavelength availability* and *aggregated load* information. Each IDRA advertises PSI messages by aggregating and assembling the following three pieces of information:

(i) Intra-domain PSI.

(ii) PSI related to the inter-domain links towards its downstream domains.

(iii) The already aggregated PSI contained in the inter-domain advertisements received from downstream domains.

In the sequel we will describe how this aggregation process is done.

Aggregated Wavelength Availability Information:

Let r and q be a pair of OXCs inside a RCD, $P(r,q)$ be a candidate path between r and q, and l be a link within the path $P(r,q)$. An IDRA computes the *Effective Number of Available Wavelengths* (ENAW) of type Λ_i between the OXCs r and q as follows:

$$W_{r,q}(\Lambda_i) = \max_{P(r,q)} \left\{ \min_{l \in P(r,q)} [W_l(\Lambda_i)] \right\} \tag{11.2}$$

The rationale in (11.2) can be easily interpreted by means of Fig. 11.4. For instance, in AS1 the ENAW of type Λ_1 between the nodes $OXC15$ and $OXC12$ is $W_{15,12}(\Lambda_1) = 3$. This is because from the two possible paths between these nodes, the path that goes through $OXC13$ has a minimum $W_{15,12}(\Lambda_1) = 1$, whereas the one that goes through $OXC11$ has a minimum $W_{15,12}(\Lambda_1) = 3$. Then, the maximum between both of them is 3. The ENAW given in (11.2) is especially important between two border OXCs in a transit domain, since it captures the practical availability of the wavelength Λ_i

[6]Under the same path sate conditions, the blocking probability of establishing a lightpath is higher when longer paths are chosen. In addition, the utilization of network resources is less efficient when longer paths are used.

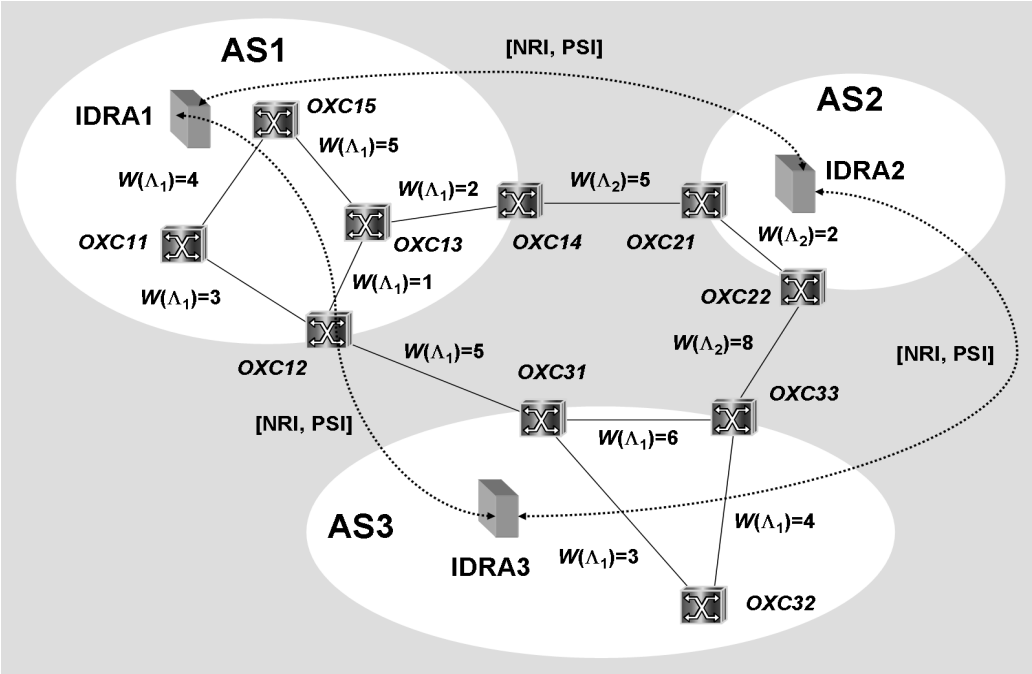


Figure 11.4: NRI and PSI exchange between the IDRAs.

within the domain. In addition, (11.2) offers highly aggregated network state information, so this is the intra-domain portion of the wavelength availability component of a PSI aggregate.

For the inter-domain portion, each IDRA knows which wavelengths are actually being used on its inter-domain links, and it also knows which wavelengths are effectively available downstream through the PSI advertisements received from neighboring IDRAs. Let $W_{l_b,r_b}(\Lambda_i)$ denote the number of available wavelengths of type Λ_i in the inter-domain link between the local border node l_b, and a remote border node r_b. For instance, in Fig. 11.4 the IDRA1 in AS1 is aware that $W_{12,31}(\Lambda_1) = 5$. Similarly, let $W^{adv}_{r_b,d}(\Lambda_i)$ denote the ENAW of type Λ_i between the remote border node r_b and the destination node d, advertised by the downstream IDRA in r_b's domain. Using these two inter-domain components and (11.2), an IDRA advertises upstream that the ENAW between a local border node l_b and a distant destination node d is:

$$W^{adv}_{l_b,d}(\Lambda_i) = \min\left\{W_{l_b,l'_b}(\Lambda_i),\, W_{l'_b,r_b}(\Lambda_i),\, W^{adv}_{r_b,d}(\Lambda_i)\right\} \qquad (11.3)$$

For instance, in Fig. 11.4 the IDRA1 advertises to the IDRA2 that the ENAW of type Λ_1 to reach $OXC32$ is:

$$W^{adv}_{14,32}(\Lambda_1) = \min_{\Lambda_1}\{W_{14,12},\, W_{12,31},\, W^{adv}_{31,32}\} = \min\{2,5,4\} = 2 \qquad (11.4)$$

Aggregated Load Information:

This comprises two sets of state information, namely, aggregated costs and aggregated blocking ratios. On the one hand, an additive cost is associated with each candidate (path, wavelength) pair. This cost reflects the current load in the availability of wavelengths in a path, allowing an IDRA to tiebreak when two or more candidate paths offer almost the same ENAW. The cost associated with a candidate path $P(s,d)$ between a local OXC s and a distant OXC d for wavelength type Λ_i is computed by an IDRA as follows:

$$C_{P(s,d)}(\Lambda_i) = \begin{cases} \left(\dfrac{1}{\min\left[W_{s,l'_b}(\Lambda_i), M_{\Lambda_i}\right]} + \right. \\ \left. \dfrac{1}{\min\left[W_{l'_b,r_b}(\Lambda_i), M_{\Lambda_i}\right]} + \dfrac{C^{adv}_{P(r_b,d)}(\Lambda_i)}{H^{adv}} \right) H \\ \infty \qquad \text{if} \quad W_{s,l'_b}(\Lambda_i) = 0 \quad \vee \quad W_{l'_b,r_b}(\Lambda_i) = 0 \end{cases} \qquad (11.5)$$

wherein, H is the number of hops from s to d considering each intra-domain sub-path as just one hop. Similarly, H^{adv} is the number of hops between the remote border node r_b and the destination node d, advertised by the downstream IDRA in r_b's domain. The term $C^{adv}_{P(r_b,d)}(\Lambda_i)$ denotes the cost from r_b to d advertised by the downstream IDRA. The ∞ in (11.5) reflects the lack of local resources to handle a connection between the nodes s and d for a particular wavelength Λ_i. If this is the case, an IDRA will remove from the NRI field of its advertisements all the destinations that were reachable through the path $P(s,d)$ for Λ_i.

The rationale in (11.5) is that the cost increases when the ENAW along an inter-domain path decreases. Likewise, the cost increases when the length

of an inter-domain path increases, so an IDRA will "generally" choose the $(P(s,d), \Lambda_i)$ pair with the lowest cost[7]. It is worth highlighting that different candidate paths offering the same ENAW will frequently have different costs (loads). For instance, in Fig. 11.4 $OXC14$ can reach $OXC33$ both through AS1 and through AS2. The ENAW of type Λ_1 through AS1 is $W_{14,33}(\Lambda_1) = 2$, and this is also the case for Λ_2 through AS2, i.e., $W_{14,33}(\Lambda_2) = 2$. From (11.5), it can be easily shown that from these two paths, the IDRA1 prefers the one through $OXC21$ given that Λ_2 is less loaded than Λ_1 (notice that $H = 3$ for both paths).

The second type of load information contained in a PSI message is an ordered sequence of aggregated *Blocking Ratios* (BRs) coming from downstream domains. Our approach is that each domain j appends in the BR sequence its $BR_j(d)$, which corresponds to the nominal ratio of path requests toward a destination d that have been blocked due to the lack of resources inside domain j. Each domain computes and updates its nominal BRs on a reasonable time-basis (in the order of several minutes, hourly, daily, etc), so that $(1 - BR_j(d))$ roughly represents the nominal probability of traversing domain j while trying to reach destination d[8]. In realistic settings it is expected that each BR in the path sequence remains low and its variations shouldn't be significant. We anticipate that the sequence of nominal BRs will aid in development of the stochastic model supporting the most effective of the RWA strategies proposed in Chapter 12. In sum, the path state information received by an IDRA for destination d is composed as follows:

$$\Phi_{PSI}(d) = \left[\left\{ W^{adv}_{r_b,d}, \left(C^{adv}_{P(r_b,d)}, H^{adv} \right) \right\}_{\Lambda_i}, \left\{ BR_j(d) \right\} \right] \tag{11.6}$$

To advertise the PSI associated with the destinations contained in the NRI messages, we take advantage of the *Keepalive* messages exchanged between neighboring IDRAs. Similarly as in the case of BGP, the IDRAs exchange Keepalive messages to confirm that neighboring IDRAs are still operative. In BGP, Keepalive messages are of fix length, consisting only of the 19-byte BGP header. In the IDRA-based model, we extend the BGP

[7]The term "generally" here is because in Chapter 12 we will introduce two different RWA strategies for the IDRAs. In one of the strategies, the IDRAs always choose minimum-cost paths. However, in the other strategy routes are chosen according to the minimum cost until the the ENAW is low (i.e. until the ENAW reaches a pre-defined threshold). When this occurs the path selection is driven by a stochastic estimation process rather than by cost.

[8]We consider that both network operators and customers can benefit from this approach, which is also aligned with some of the main ideas proposed in [74].

Keepalive concept with the purpose of conveying PSI, when relevant PSI needs to be updated. In other words, the update of PSI between RCDs is handled by means of a dedicated – and physically independent – control plane, supported by specialized IDRAs that exchange non-dummy Keepalive messages with their neighbors.

In Section 12.2.1 we shall show that when a RWA algorithm exploits the highly aggregated PSI given by the ENAW in (11.2), or the cost in (11.5), it is possible to obtain drastic reductions in the number of blocked inter-domain lightpath requests, compared against a RWA approach like OBGP [16, 39, 130, 131]. However, when the ENAW along all candidate paths is low, even though these RWA algorithms outperform OBGP, they still yield blockings that can be considerably improved (see Sections 12.3 – 12.6). The reason for this is that the path state information considered in (11.5) does not take into account the traffic demands. This is precisely what we address in the stochastic model developed in 12.3.

Chapter 12

RWA Strategies for Multi-Domain Optical Networks

This chapter introduces and contrasts three distributed RWA strategies for multi-domain optical networks. First, we present OBGP+, which is our improved version of the optical extension of BGP [16, 39, 130, 131]. Our aim in this case is to show that by simply integrating plain and highly aggregated PSI in OBGP, it is possible to drastically improve its performance[1] (see Fig.), and this can be accomplished without increasing the number or the frequency of routing updates exchanged between domains (see Table). As we will show, the strengths of OBGP+ lie in the fact that it is able to partially exploit the PSI introduced in Chapter 11. More precisely, OBGP+ uses the ENAW given in (11.2).

It is important to highlight that OBGP+ has its roots in BGP, so the IDRAs are no part of the OBGP+ routing model. The introduction of OBGP+ allows us to demonstrate in a straightforward way that, even without changing the architecture (like is the case of the IDRA-based model), by simply endowing a routing protocol like OBGP with the capability to compute, aggregate, and convey "only minor" PSI, it is possible to drastically reduce its blocking ratio.

Next, we present two RWA strategies for the IDRA-based network architecture, namely, Cost and Cost+Kalman [140]. Cost exploits the additive cost proposed in Chapter 11, and as we shall show it significantly improves the performance obtained with OBGP+. Cost+Kalman on the other hand, enhances Cost in such a way that it performs exactly as Cost for low and medium traffic loads, but it outperforms this latter when the traffic load on the network grows large. The Cost+Kalman strategy is supported by a stochastic model and a Kalman filter that are used together to estimate the occupancy of the wavelengths along an inter-domain path prior to the RWA

[1]The performance metric considered here is the blocking ratio of inter-domain lightpath requests.

decision. As we will show, this approach becomes especially effective when the availability of wavelengths is scarce.

In sum, this chapter presents a sequence of three RWA strategies for multi-domain optical networks, each of which gradually improves the performance obtained with the previous one. We will also show that all the RWA strategies proposed here, drastically improve the performance achievable with OBGP.

12.1 The OBGP+ RWA algorithm

As BGP, OBGP is essentially a shortest AS-path routing algorithm that exchanges NRI, but it does not handle PSI. Understanding that this is a missing piece in the routing models provided by BGP and OBGP is easy nowadays, but contributing with solutions capable of highly improving the performance of these routing protocols without increasing the number and frequency of the routing messages exchanged between domains is a challenging task.

As a first step in this direction, we propose OBGP+. Our OBGP+ handles the highly aggregated PSI supplied by the ENAW introduced in Section 11.3.2. Accordingly, each OBGP+ node computes and advertises the ENAW along the candidate paths, as described in (11.2) and (11.3). In order to avoid the typical increase in the number of routing messages associated with the update of PSI, OBGP+ uses the same approach proposed for the IDRA-based model in Section 11.3.2, that is, OBGP+ nodes are able to piggy-back this information in non-dummy Keepalive messages, when relevant PSI needs to be updated.

Algorithm 8 shows a simplified version of the OBGP+ decision process, which is the result of a set of enhancements that we introduced to OBGP [16]. From Algorithm 8 it is clear that OBGP+ is essentially a "shortest AS-path highest ENAW" RWA algorithm, given that it usually prefers the shortest AS-path (step 2 of the algorithm), but if more than one candidate lightpath exists, then it chooses the one with the highest ENAW (step 3).

12.1.1 Performance evaluation of OBGP+

The aim of this section is to contrast the performance of OBGP+ against OBGP. We assume that both OBGP and OBGP+ handle exactly the same NRI and treat it exactly in the same way. Similarly as in (11.1), the NRI distributed between OBGP/OBGP+ nodes is composed by:

$$\Phi_{NRI}(d) = \Big[\text{AS-path, NH, } (\Lambda_i, M_{\Lambda_i})\Big]_d \tag{12.1}$$

Algorithm 8 OBGP+($\{P(s,d)$, Λ_i, M_{Λ_i}, $z_i\}$)

Input: $\{P(s,d)\}$ - set of paths between nodes s and d
Λ_i - a particular wavelength on path $P(s,d)$
M_{Λ_i} - Multiplicity of wavelength Λ_i on path $P(s,d)$
z_i - ENAW of type Λ_i along the path $P(s,d)$
Output: $(P^{best}, \Lambda^{best})$ - The best lightpath between s and d

1: Choose the (path, wavelength) pair with the highest local preference (LOCAL_PREF) /* As in BGP */
2: If the LOCAL_PREFs are equal, choose the shortest AS-path and assign the wavelength with highest ENAW among the ones available on that path. If more than one wavelength has the same (highest) ENAW along the shortest AS-path, choose the wavelength with the lowest ID
3: If the AS-path lengths are equal choose the (path, wavelength) pair associated with the highest ENAW
4: If the ENAWs are equal prefer external paths over internal paths
5: If the paths are still equal prefer the one with the highest ENAW to the next-hop OXC (i.e., to the OXC r_b in the neighboring domain)
6: If more than one path is still available run OBGP tie-breaking rules /* As in BGP */

Our interest here is to compare two different performance metrics, namely, the *Blocking Ratio* (BR) of inter-domain lightpath requests, and the number of routing messages exchanged to achieve this blocking.

To this end, we have conducted extensive simulations using OPNET Modeler [92]. The simulation results presented here can be reproduced using our OPNET modules, which are available online from [91].

The inter-domain scenario chosen for the trials was the complete PAN European network topology illustrated in Fig. 9.5. We recall that this multi-domain network is composed by 28 domains and 41 inter-domain links. For the network topology inside each domain in the PAN, we have randomly chosen a minimum number of nodes equal to the number of inter-domain links of that domain, up to a maximum of seven nodes inside each domain. This approach guarantees that each inter-domain link of a domain in the PAN is supported by a different border node. We have used 5 fibers per-link, and 16 wavelengths per-fiber thoughout all the PAN European Network.

In order to assess the impact of the frequency of update in the PSI, we have used different *Keepalive Update Intervals* K_T during the trials. K_T cor-

responds to the time interval between the delivery of non-dummy Keepalive messages conveying PSI. At present, most implementations of BGP use a default Keepalive value of 60 seconds, and three consecutive Keepalive messages need to be lost so that a BGP router proceeds to shutdown a BGP session. In our simulations we have tested three different scaled and normalized values: $K_T = 1$, $K_T = 3$, and $K_T = 5$ units through the simulation runtime. Clearly, the higher the values of K_T, the more time is needed by OBGP+ nodes to detect an react when a neighbor becomes inoperative. Therefore, a major advantage of conveying PSI piggy-backed in Keepalive messsages is that low values of K_T are desired both to increase the responsiveness between OBGP+ neighbors as well as to support updating PSI more frequently.

The simulation results shown here were obtained using cross Poisson traffic between different domains, where we have randomly picked at least 10 sources and 8 destinations along the PAN European network. As shown in Fig. 12.1, the trials were performed for different traffic loads, varying from 100 Erlangs up to 300 Erlangs. As mentioned before the metric used to assess the performance achieved by OBGP and OBGP+ was the BR of inter-domain lightpath requests, which is a usual performance metric used in wavelength routed optical networks.

Figure 12.1 shows the results obtained with OPNET for the different traffic loads and the different Keepalive update intervals K_T assessed. Clearly, OBGP+ outperforms OBGP, and it becomes evident that even minor PSI, like the ENAW, is enough to drastically reduce the blocking obtained $\forall$ K_T. Whereas OBGP experiences blocking for all traffic loads tested, OBGP+ starts to show some negligible blocking only after reaching 200 Erlangs.

In order to quantify the reductions supplied by OBGP+ in terms of blocking, we define the following *Improvement Factor* (IF) of OBGP+ vs. OBGP:

$$IF \triangleq \left(\frac{BR^{(OBGP)}}{BR^{(OBGP+)}} \right)_{\text{Traffic (Erlangs)}} \tag{12.2}$$

Table 12.1 summarizes the improvement factor *IF* for the highest traffic load simulated, i.e., 300 Erlangs, as well as the number of routing messages exchanged for the different traffic loads tested. The results show that OBGP+ is able to reduce the BR by more than one order of magnitude even for the highest traffic load evaluated.

Moreover, OBGP+ always needs less overall number of routing messages than OBGP. The reason for this is twofold. First, because PSI updates are never triggered between OBGP+ neighbors. Instead, they are piggy-backed in the Keepalive messages used in both OBGP and OBGP+. And second,

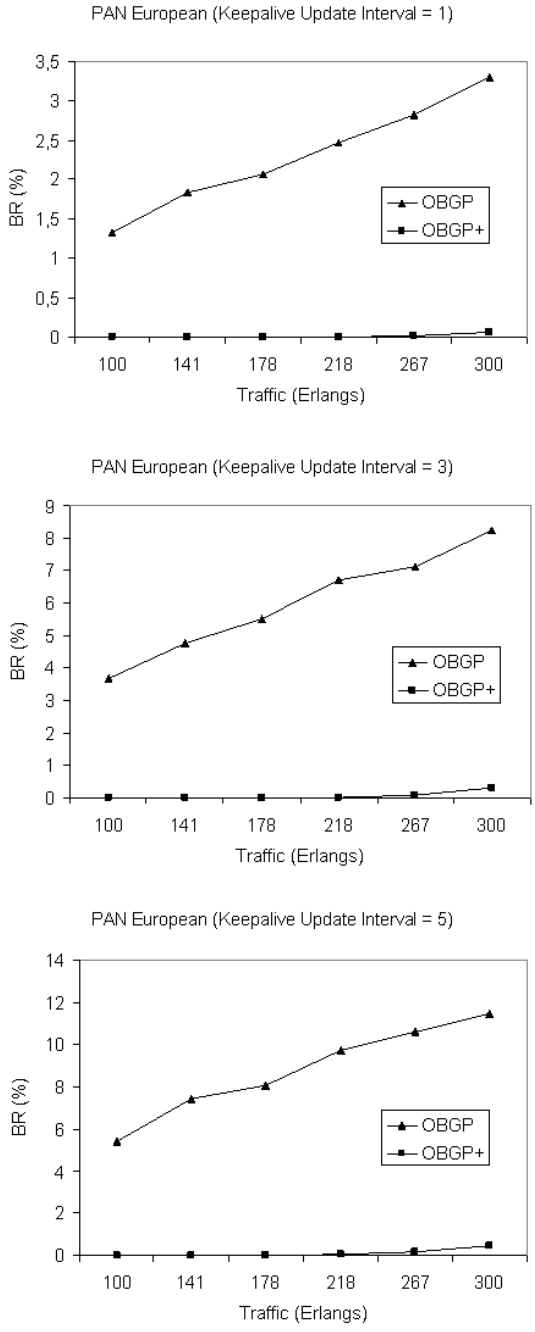


Figure 12.1: Comparison between OBGP and OBGP+
(for different traffic loads and for different Keepalive Update Intervals K_T).

	Keepalive Update Interval $K_T = 1$		**Keepalive Update Interval** $K_T = 3$		**Keepalive Update Interval** $K_T = 5$	
	OBGP	OBGP+	OBGP	OBGP+	OBGP	OBGP+
***IF* for 300 Erlangs**	$\simeq 49$		$\simeq 26$		$\simeq 23$	
Traffic (Erlangs)	Routing Messages	Routing Messages	Routing Messages	Routing Messages	Routing Messages	Routing Messages
100	19,367,000	13,885,369	17,039,387	13,650,055	15,064,687	13,417,374
141	22,085,832	13,918,733	18,724,788	13,700,950	16,604,889	13,506,549
178	22,763,245	14,176,309	19,349,592	13,896,000	16,800,385	13,634,036
218	23,056,532	14,313,930	19,880,766	14,094,911	17,128,965	13,912,425
267	25,026,790	14,540,331	20,747,022	14,371,841	17,690,792	14,197,668
300	25,346,526	14,996,355	20,839,770	14,491,483	17,863,027	14,266,768

Table 12.1: Improvements of OBGP+ vs. OBGP.
Improvement Factor in the blocking requests for 300 Erlangs, and overall number of routing messages exchanged.

OBGP tends to exhaust the available wavelengths along the shortest AS-path before switching to an alternative path. This triggers network reachability messages and path exploration after paths become blocked. Conversely, OBGP+ explicitly considers the ENAW in the RWA algorithm when two or more paths exhibit the same AS-path length, so it is able to provide a much better traffic distribution than OBGP, with drastic reductions in the BR, and hence, less network reachability messages need to be exchanged.

To summarize, we have shown that by integrating even minor path state information in OBGP, like the ENAW in (11.2), it is possible to reduce its blocking ratio by a factor that might roughly vary between 20 and 50, depending on how frequently the aggregated path state information is updated between domains. We have also shown that these improvements can be achieved without needing to exchange more routing messages than with OBGP. In fact, OBGP+ reduces the number of routing messages exchanged between routing domains.

In what follows, we will introduce two RWA strategies for the IDRA-based route control model proposed in Chapter 11. We shall show that when additional – though highly aggregated – PSI like the cost in (11.2) is exploited

by the IDRAs, it is possible to:

- Obtain significant reductions in the blocking compared to OBGP+.
- This can also be achieved without incrementing the number of routing messages exchanged with OBGP+.

12.2 The Cost RWA Algorithm

The Cost RWA algorithm runs on the IDRAs introduced in Chapter 11, and it takes advantage of the TE information exchanged among them. It is based on the computation of the additive cost given in (11.5), so the algorithm handles both wavelength availability information and load information. Therefore, it is expected to perform better than OBGP+, given that it handles more path state information than this latter. The results obtained in Section 12.2.1 confirm this fact.

A simplified version of the lightpath selection process followed by Cost is shown in Algorithm 9[2]. The algorithm shows that the IDRAs choose minimum-cost paths (step 5), and if more than one path shows the same (minimum) cost, the IDRAs tie-break first by the ENAW along the candidate paths, then by the shortest number of hops H, and after that, by following essentially the same steps as OBGP+.

12.2.1 Performance evaluation of Cost

The aim of this section is to contrast the performance of Cost against OBGP and OBGP+. To this end, we tested Cost exactly in the same setting (i.e. the complete PAN European topology), and the same conditions described in Section 12.1.1. The main difference is that now each domain in the PAN European network allocates one IDRA. The OPNET modules of the IDRAs are also available from [91].

Figure 12.2 shows the results obtained with OPNET for the different traffic loads, and for the different Keepalive Update Intervals (K_T) assessed. Clearly, both Cost and OBGP+ outperform the legacy version of OBGP, and when the traffic load increases Cost supplies significant improvements compared to OBGP+. More precisely, Cost is able to reduce by more than a 60% the blocking achieved with OBGP+ for the highest traffic load assessed (300 Erlangs), and this occurs $\forall\ K_T \in \{1, 3, 5\}$.

[2]We recall that the IDRAs are capable of choosing more than one path per-destination. For the sake of simplicity in the exposition of the algorithm we only show here the selection of a single route.

Algorithm 9 Cost($\{P(s,d)$, Λ_i, M_{Λ_i}, z_i, $C^{adv}_{P(r_b,d)}(\Lambda_i), H^{adv}\}$)

Input: $\{P(s,d)\}$ - set of paths between nodes s and d
Λ_i - a particular wavelength on path $P(s,d)$
M_{Λ_i} - Multiplicity of wavelength Λ_i on path $P(s,d)$
z_i - ENAW of type Λ_i along the path $P(s,d)$
$C^{adv}_{P(rb,d)}(\Lambda_i)$ - the cost from the remote border OXC r_b to destination d advertised by the downstream IDRA using wavelength Λ_i
H^{adv} - number of hops between the remote border OXC r_b and d advertised by the downstream IDRA
Output: $(P^{best}, \Lambda^{best})$ - The best lightpath between s and d

1: **for** each $(P(s,d)$, $\Lambda_i)$ pair **do**
2: Compute the cost $C_{P(s,d)}(\Lambda_i)$ /* Equation (11.5) */
3: **end for**
4: /* Lightpath Selection Process */
5: Choose the (path, wavelength) pair with the minimum cost
6: If the costs are equal choose the path with the highest ENAW
7: If the ENAWs are equal choose the path with the shortest number of hops H, and assign the wavelength with the lowest ID
8: If the hops H are equal prefer the path with the highest ENAW to the remote border r_b
9: If more than one path is still available run OBGP tie-breaking rules

Table 12.2 compares the improvement factor *IF* of Cost vs. OBGP+ for the highest traffic load simulated[3], as well as the number of routing messages exchanged for the different traffic loads tested. The results confirm that Cost is able to reduce the BR obtained with OBGP+ by more than 60% for the highest traffic load evaluated, since the improvement factor varies approximately between 2.5 and 3. Moreover, Cost always needs less overall number of routing messages than OBGP+, and thus, less than OBGP as well. The reason for this is that by decrementing the blocking, Cost reduces the exchange of network reachability messages and path exploration that OBGP+ needs when blocking starts to occur.

One of the most important strengths of the Cost RWA strategy is that it

[3]Similarly as in (12.2), the *IF* in this case is: $IF \triangleq \left(\frac{BR^{(OBGP+)}}{BR^{(Cost)}}\right)_{\text{Traffic (Erlangs)}}$

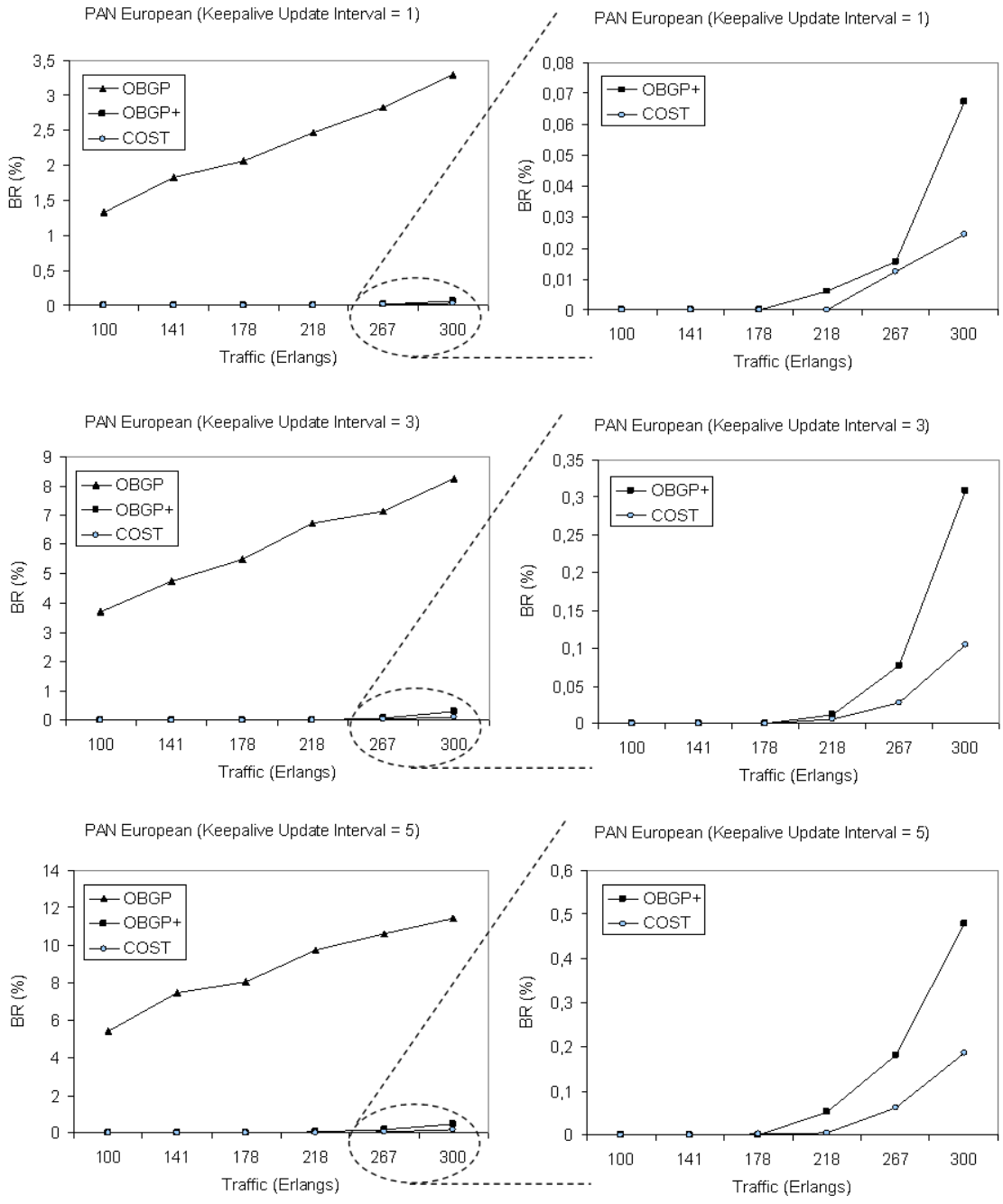


Figure 12.2: Comparison between OBGP, OBGP+, and Cost
(for different traffic loads and for different Keepalive Update Intervals K_T).

	Keepalive Update Interval $K_T = 1$		**Keepalive Update Interval** $K_T = 3$		**Keepalive Update Interval** $K_T = 5$	
	OBGP+	Cost	OBGP+	Cost	OBGP+	Cost
***IF* for 300 Erlangs**	$\simeq 2.75$		$\simeq 2.97$		$\simeq 2.56$	
Traffic (Erlangs)	Routing Messages	Routing Messages	Routing Messages	Routing Messages	Routing Messages	Routing Messages
100	13,885,369	11,709,023	13,650,055	11,538,557	13,417,374	11,343,159
141	13,918,733	12,038,433	13,700,950	11,704,861	13,506,549	11,385,050
178	14,176,309	12,055,702	13,896,000	11,778,611	13,634,036	11,400,721
218	14,313,930	12,131,877	14,094,911	11,779,592	13,912,425	11,423,262
267	14,540,331	12,179,014	14,371,841	11,806,324	14,197,668	11,439,672
300	14,996,355	12,331,307	14,491,483	11,830,229	14,266,768	11,582,755

Table 12.2: Improvements of Cost vs. OBGP+.
Improvement Factor in the blocking requests for 300 Erlangs, and overall number of routing messages exchanged.

supports the computation of lightpaths based on highly synthesized PSI, like the additive cost given in (11.5), the ENAW along a path given in (11.2), and the number of hops H traversed to reach a destination.

A weakness, however, is that the RWA strategy provided by Cost does not take into account the traffic demands during the lightpath selection process. Indeed, when the number of available wavelengths along all candidate paths is low, it is possible to modify the RWA strategy used by Cost and considerably improve its performance. That is precisely the goal of our third RWA algorithm, namely, Cost+Kalman. The following two sections are dedicated to provide the theoretical support of the modified Cost+Kalman RWA strategy that we shall propose later in Section 12.5.

12.3 Stochastic estimation of the wavelength availability

In state dependent circuit-switched networks the occupancy and the traffic arrival rates are typically coupled to each other, since the occupancy determines the traffic carried by the network and the carried traffic determines in turn the occupancy [115]. As a consequence, models developed to obtain explicit forms of the occupancy are highly complex, and typically involve multidimensional Markov processes leading to a set of coupled non-linear equations [88]. Unfortunately, the occupancy does not have a close-form expression, so numerical evaluations and complex iterations are additionally needed in order to find either the blocking probabilities or the occupancy along the paths. And all this applies assuming complete knowledge of the arrival and departure rates of connections along the different segments of the network, which is never the case in multi-domain settings.

In this section we propose a quite different approach. We aim at relaxing the model complexity by deriving an approximate linear stochastic model to roughly estimate the number of available wavelengths along an inter-domain path between two routing updates[4], and rely on a Kalman-based predictor-corrector to refine the previous estimation. Unlike traditional Kalman filters, which use noisy measurements as their observations, we use the information contained in the routing updates as noisy measurements of the current wavelength occupancy along the paths. Based on these observations, we estimate the ENAW on the candidate paths until the next routing update, and use this information to influence the RWA decision when the number of available wavelengths makes a lightpath request prone to be blocked. We shall show later in this chapter that this estimation considerably improves the performance of the RWA strategy in terms of the lightpath blocking ratio. The approximate model we are proposing here is based on a noisy extension of a simplified model derived for the two domains in Fig. 12.3, so we will first focus on this case.

Figure 12.3 shows a source domain AS1 and a directly connected destination domain AS2 consisting of a single OXC $OXC2$. The inter-domain calls from AS1 to AS2 for wavelength Λ_i are assumed to be Poisson with exponentially distributed arrival rate λ. The duration of these calls are also assumed to be exponentially distributed with departure rate μ. For the scenario in Fig. 12.3 we assume the same arrival and departure rates $\forall\ \Lambda_i$. It is worth highlighting that in extended scenarios this might not always be the

[4]What matters here is the inaccuracy of the PSI at the time that the routing update is received, and especially, during the interval between two routing updates.

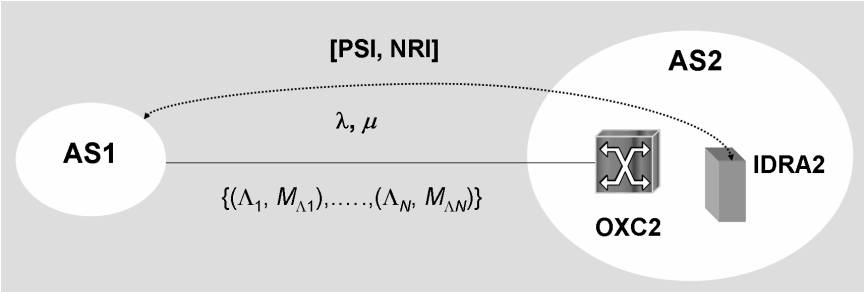


Figure 12.3: Estimation of the number of available wavelengths. The estimation in this simple case is between two directly connected ASs, where the destination AS consists of only one OXC.

case. In Sections 11.3.1 and 11.3.2 we explained the flexibility that an AS has while composing its NRI advertisements. Such flexibility can cause that a given destination is reachable for some wavelengths but unreachable for some others, so the traffic demands may differ for different wavelengths.

Let $x_i(t)$ denote the ENAW of type Λ_i at time t in the inter-domain link in Fig. 12.3. Our goal is to estimate $x_i(t)$ according to the preceding traffic demands. Let $p_k(t)$ be the probability that the ENAW of type Λ_i at time t is k, that is, $p_k(t) = Prob\{x_i(t) = k\}$. The process $x_i(t)$ evolves in $t \in [(n-1)\,T, nT]$, $n \in N^+$, according to the birth and death model in Fig. 12.4, where T denotes the *observation time interval*. This latter is the average time interval between two routing updates, taking into account the updates coming from NRI messages, the ones coming from PSI messages, and also the ones coming from the allocations performed by the local IDRA.

The birth and death process in Fig. 12.4 has $(M_{\Lambda_i} + 1)$ states, where the state "0" indicates that no wavelengths of type Λ_i are available in the inter-domain link. Then, the state transitions can be described by the following set of differential equations for the probabilities $p_k(t)$:

$$\dot{p}_k(t) = [M_{\Lambda_i} - k + 1]\,\mu p_{k-1}(t) - [k\lambda + (M_{\Lambda_i} - k)\,\mu]\,p_k(t) \; + \\ (k+1)\lambda p_{k+1}(t) \tag{12.3}$$

With boundary conditions:

$$\begin{cases} \dot{p}_0(t) = -M_{\Lambda_i}\mu p_0(t) + \lambda p_1(t) \\ \dot{p}_{M_{\Lambda_i}}(t) = \mu p_{M_{\Lambda_i}-1}(t) - M_{\Lambda_i}\lambda p_{M_{\Lambda_i}}(t) \end{cases} \tag{12.4}$$

Then, we use the expected value of $x_i(t)$ as its estimator. Using (12.3) and (12.4), the expected value can be derived as follows:

$$\dot{E}\left[x_i(t)\right] = \frac{d}{dt}\sum_{k=0}^{M_{\Lambda_i}} k p_k = \sum_{k=0}^{M_{\Lambda_i}} k\dot{p}_k = \mu M_{\Lambda_i} - (\lambda+\mu)E\left[x_i(t)\right] \tag{12.5}$$

Integrating (12.5) in the observation time interval yields (12.6):

$$E\left[x_i(nT)\right] = E\left[x_i((n-1)T)\right]e^{-(\lambda+\mu)T} + \left(\frac{\mu M_{\Lambda_i}}{\lambda+\mu}\right)\left[1 - e^{-(\lambda+\mu)T}\right] \tag{12.6}$$

Equation (12.6) allows to recurrently estimate $x_i(t)$. If the state is known at the beginning of the observation interval, (12.6) solves the estimation problem for the inter-domain scenario in Fig. (12.3) until the next observation interval. As mentioned at the beginning of this section, accurately extending

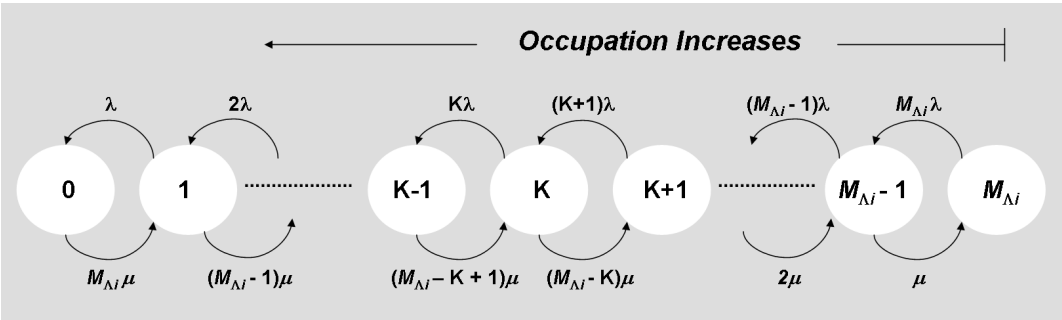


Figure 12.4: Birth and death process.
The process models the availability of wavelengths of type Λ_i for the two directly connected ASs in Fig. 12.3.

this model for multiple traffic demands and multiple domains is not possible, due to the lack of information needed to build the model. In the sequel we propose a straightforward way to "roughly" estimate $x_i(t)$ in such cases, and rely on the strengths of the Kalman filter to refine this estimation.

Let l_{ID} denote a particular inter-domain link of the AS for which we want to derive the estimation. We define D_{Λ_i} as the set of all possible destinations that are reachable through l_{ID} using wavelength Λ_i. We assume that the inter-domain calls requesting a route through l_{ID} to a destination $d \in D_{\Lambda_i}$ arrive as independent Poisson processes with exponentially distributed arrival rate λ_d. The duration of these calls are also assumed to be exponentially distributed with the same departure rate μ, and they are assumed to be independent of previous arrivals and holding times. Based on these assumptions, we propose to extend the estimation in (12.6) as follows:

$$E\left[x_i(nT)\right] \approx E\left[x_i((n-1)T)\right] e^{-\gamma_{i,n-1}T} + \left(\frac{\mu M_{\Lambda_i}}{\gamma_{i,n-1}}\right)\left[1 - e^{-\gamma_{i,n-1}T}\right] \quad (12.7)$$

$$\gamma_{i,n-1} = \left(\sum_{d \in D_{\Lambda_i}} \lambda_d \prod_{j=1}^{H^{adv}} \left[1 - BR_j(d)\right] + \mu \right)_{t=(n-1)T} \quad (12.8)$$

The rationale in (12.7)–(12.8) is three-fold. First, the model captures the essential characteristics of state-dependent circuit switched networks. Second, the model is simple and easy to compute since it uses aggregated state information that is locally available. And third, the inherent coupling between the arrival rates and the occupancy is straightforwardly relaxed by means of the BR advertisements.

In this framework we define the following constants:

$$\begin{cases} A_{i,n-1} = e^{-\gamma_{i,n-1}T} \\ \\ B_{i,n-1} = \left(\dfrac{\mu M_{\Lambda_i}}{\gamma_{i,n-1}}\right)\left[1 - e^{-\gamma_{i,n-1}T}\right] \end{cases} \quad (12.9)$$

In order to simplify the notation we define $E\left[x_i(nT)\right] = x_{i,n}$. Then, using

the approximation in (12.7) and the constants defined above we have:

$$\begin{cases} x_{i,n} = A_{i,n-1}x_{i,n-1} + B_{i,n-1} + w_{i,n-1} \\ \\ w \sim (0, Q) \end{cases} \tag{12.10}$$

where w represents the process noise (a.k.a the model noise), which is assumed to be white, with zero mean and variance Q [112].

The linear stochastic difference equation in (12.10) is the main result of this section, and it is precisely the input for the discrete-time Kalman filter.

12.4 The Kalman filter

Kalman filters have been widely used in different disciplines, like adaptive control [112], ATM networks [66], and recently, in IP/MPLS networks [7], given their optimal estimation-prediction error characteristics . They are also powerful tools, since they offer a computationally efficient way to optimally estimate the state of a controlled process. The estimation is optimal in the sense that Kalman filters minimize the covariance of the estimation error [112].

The basic principle of Kalman filters is that they alternate between two steps, namely, a prediction step and a correction step. The idea is to predict the next state of a process based on the partial knowledge of the current state, and then adjust this prediction with the new information coming from the observations. The adjusted state is then considered as the new prediction and so on. In the sequel we introduce the prediction-correction steps for our particular problem. We start by defining the following set of variables in Table 12.3.

Using (12.10), the usual prediction-correction Kalman steps yield [112]:

Prediction Step:

$$\begin{cases} x^{-}_{i,n} = A_{i,n-1}x^{+}_{i,n-1} + B_{i,n-1} \\ \\ \varepsilon^{-}_{i,n} = A^{2}_{i,n-1}\varepsilon^{+}_{i,n-1} + Q \end{cases} \tag{12.11}$$

Correction Step:

$$\begin{cases} K_{i,n} = \left(\dfrac{\varepsilon_{i,n}^{-}}{\varepsilon_{i,n}^{-} + R} \right) \\ x_{i,n}^{+} = x_{i,n}^{-} + K_{i,n}[z_{i,n} - x_{i,n}^{-}] \\ \varepsilon_{i,n}^{+} = [1 - K_{i,n}]\varepsilon_{i,n}^{-} \end{cases} \tag{12.12}$$

An important feature of Kalman filters is that their convergence is not biased by the initial state [112]. Another important aspect tightly linked to the precision of Kalman filters is the dynamic estimation of the variances of the process and observation noise, i.e., Q and R, respectively. This is usually performed by means of maximum likelihood estimation techniques.

In what follows, we introduce the Cost+Kalman RWA algorithm, and particularly explain: i) the link between our Cost RWA strategy (see Algorithm 9) and the Kalman filter; and ii) when and how the Kalman-based estimation is used.

Symbol	Description
$x_{i,n}^{-}$	A *priori* estimate of the ENAW Λ_i on path P (predicted state)
$x_{i,n}^{+}$	A *posteriori* estimate of the ENAW Λ_i on path P (corrected state)
$e_{i,n}^{+} = x_{i,n} - x_{i,n}^{+}$	Estimation error
$\varepsilon_{i,n}^{+} = E\left[(e_{i,n}^{+})^2\right]$	Estimation error covariance
$K_{i,n}$	Gain of the filter. Kalman filters set their gain $K_{i,n}$ so as to minimize the estimation error covariance
$z_{i,n} = x_{i,n}^{+}$	$z_{i,n}$ is the observation. Denotes the ENAW Λ_i on a path P observed from the routing updates
$v_{i,n}$	$v_{i,n}$ is the observation noise, which is assumed to be white, with zero mean and variance R [112]

Table 12.3: Notation for the Kalman filter.

12.5 The Cost+Kalman RWA Algorithm

For the Cost+Kalman RWA algorithm we define a configurable threshold T_h that triggers the utilization of the filter in the RWA decision of the Cost algorithm. When the effective number of "all" available wavelengths along the candidate paths is below or equal to T_h, the RWA running on the IDRAs is driven by the Kalman filter. If this is not the case, the RWA is performed using Cost, i.e., the IDRAs choose the $(P(s,d), \Lambda_i)$ pair with minimum cost.

Algorithm 10 describes the behavior of the Cost+Kalman RWA strategy. Clearly, Cost+Kalman coincides with Cost until the threshold T_h is reached in all candidate paths. When this occurs (i.e. when the wavelength occupancy along all candidate paths is high) the Kalman-based estimation-correction filter takes the control of the lightpath selection process. Therefore, the Kalman filter is basically a module aiding the Cost RWA. In particular, when the threshold $T_h = 0$, the Kalman RWA algorithm is off all the time, and the Cost+Kalman RWA algorithm is identical to Cost. The details about the Kalman filtering process are shown in Algorithm 11.

12.6 Performance evaluation of Cost+Kalman

Similarly as in Sections 12.1.1 and 12.2.1, our goal is to contrast the performance of the Cost+Kalman RWA strategy against OBGP+ and Cost. The simulation results shown here were obtained using a C-based simulator developed at the Department of Computer Architecture of the Technical University of Catalonia, since by the time of this writing the Cost+Kalman RWA was being implemented and integrated to OPNET. The C-based simulation tool used here, has been successfully used in various recent works like [78, 79, 140].

Our simulations were conducted over the same topology that we used in [140], which is shown in Fig. 12.5. In this topology, we have used 5 fibers per-link, and 24 wavelengths per-fiber. The Kalman threshold for all the trials was set to $T_h = 2$. We have once again used different Keepalive update intervals K_T: $K_T = 1$, $K_T = 3$, and $K_T = 8$ in this case. Our simulations were conducted using cross Poisson traffic between different domains, and as shown in Fig. 12.6, the trials were performed for different traffic loads.

As indicator of the performance obtained with the different RWA strategies, we have once again used the percentage of blocked inter-domain lightpath requests. This is shown in Fig. 12.6.(a) for different traffic loads and for different Keepalive update intervals.

Algorithm 10 Cost+Kalman($\{P(s,d)$, Λ_i, M_{Λ_i}, z_i, $C^{adv}_{P(r_b,d)}(\Lambda_i), H^{adv}\}, T_h$)

Input: $\{P(s,d)\}$ - set of paths between nodes s and d
Λ_i - a particular wavelength on path $P(s,d)$
M_{Λ_i} - Multiplicity of wavelength Λ_i on path $P(s,d)$
z_i - ENAW of type Λ_i along the path $P(s,d)$
$C^{adv}_{P(rb,d)}(\Lambda_i)$ - the cost from the remote border OXC r_b to destination d advertised by the downstream IDRA using wavelength Λ_i
H^{adv} - number of hops between the remote border OXC r_b and d advertised by the downstream IDRA
T_h - Threshold that triggers the Kalman filter

Output: $(P^{best}, \Lambda^{best})$ - The best lightpath between s and d

1: **for** each $(P(s,d)$, $\Lambda_i)$ pair **do**
2: Compute the cost $C_{P(s,d)}(\Lambda_i)$ /* Equation (11.5) */
3: **end for**
4: /* Lightpath Selection Process */
5: **if** $T_h \leq z_i$ $\forall$ candidate path $P(s,d)$ and wavelength Λ_i **then**
6: Choose the (path, wavelength) pair with the highest estimated ENAW by Kalman
7: If the ENAWs estimated by Kalman are equal choose the (path, wavelength) pair with the minimum cost
8: If the costs are equal choose the path with the shortest number of hops H, and assign from the wavelengths provided by Kalman, the one with the lowest ID
9: If the hops H are equal prefer the path with the highest ENAW to the remote border r_b
10: If more than one path is still available run OBGP tie-breaking rules
11: **else**
12: Run the usual Lightpath Selection Process in Cost (steps 4–9 of Algorithm 9)
13: **end if**

Algorithm 11 Kalman Estimation($P(s,d)$, Λ_i, M_{Λ_i}, z_i, $\{BR\}$)

Input: $P(s,d)$ - a path between nodes s and d
Λ_i - a particular wavelength
M_{Λ_i} - Multiplicity of wavelength Λ_i
z_i - ENAW of type Λ_i along the path $P(s,d)$
(z_i is obtained from the PSI updates)
$\{BR\}$ - Set of reported blocking ratios of downstream domains for path P

Output: $x_i^-(n)$ - The estimated number of available wavelengths of type Λ_i along the path $P(s,d)$

1: $x_{i,0}^+ \leftarrow x_0$ /* Set initial condition in $x_{i,n}$ */
2: $\varepsilon_{i,0}^+ \leftarrow \varepsilon_0$ /* Set initial condition in $\varepsilon_{i,n}$ */
3: Compute $\gamma_{i,n-1}$ /* Equation (12.8) */
4: Compute $A_{i,n-1}$ /* Equation (12.9) */
5: Compute $B_{i,n-1}$ /* Equation (12.9) */
6: /* Prediction Step */
7: $x_{i,n}^- = A_{i,n-1}x_{i,n-1}^+ + B_{i,n-1}$
8: Compute Q
9: $\varepsilon_{i,n}^- = A_{i,n-1}^2\varepsilon_{i,n-1}^+ + Q$
10: /* End of Prediction Step */
11: Store $x_{i,n}^-$ and *Wait* for the next routing UPDATE
12: /* Correction Step (when an UPDATE arrives) */
13: Compute R
14: Compute $K_{i,n}$ /* Equation (12.12) */
15: $x_{i,n}^+ = x_{i,n}^- + K_{i,n}[z_{i,n} - x_{i,n}^-]$
16: $\varepsilon_{i,n}^+ = [1 - K_{i,n}]\varepsilon_{i,n}^-$
17: /* End of Correction Step */
18: Go to Step 3

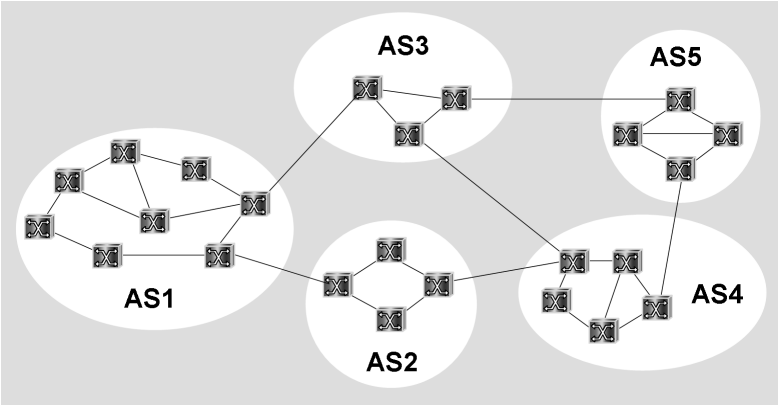


Figure 12.5: Network topology.

Clearly, Cost+Kalman outperforms OBGP+, and it supplies significant improvements when compared with the Cost RWA algorithm. As expected, these improvements are especially noticeable as the traffic load increases. In this case, the wavelength availability decreases and the Kalman-based estimation aids Cost during the lightpath decision process.

Conversely, when the traffic load is low the Kalman filter is barely used, and hence the performances of Cost and Cost+Kalman are essentially the same. Figure 12.6.(b) shows the percentage of RWA decisions that were taken using Kalman for the different traffic loads.

Table 12.4 summarizes the relative percentage of improvement in the blocking ratio for the highest simulated traffic load in the network, i.e., 133 Erlangs in this case.

Keepalive update interval	**% of improvement Cost+Kalman vs. OBGP+**	**% of improvement Cost+Kalman vs. Cost**
$K_T = 1$	67%	67%
$K_T = 3$	23%	23%
$K_T = 8$	37%	21%

Table 12.4: Relative percentage of improvement in the blocking requests. Data for the highest simulated traffic load (133 Erlangs).

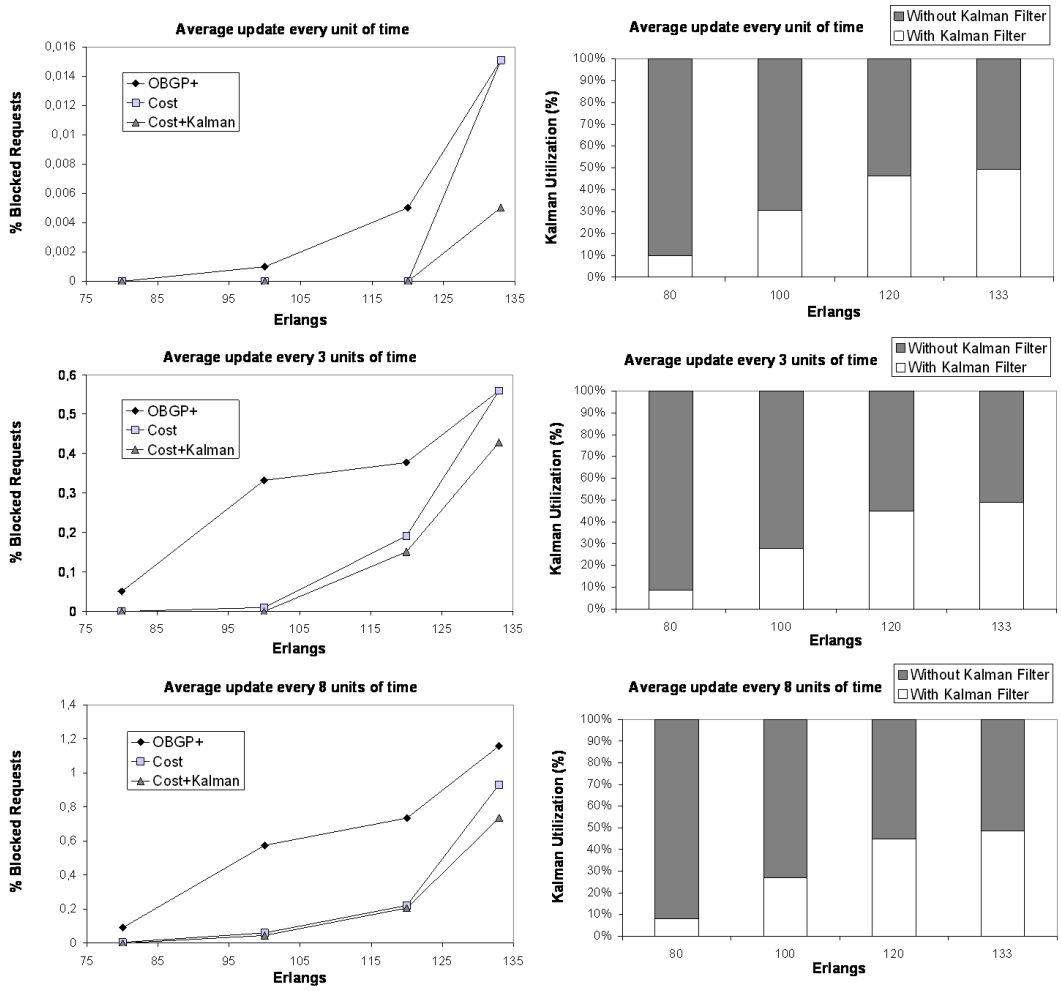


Figure 12.6: Performance evaluation.
(a) (Left) Percentage of blocked requests with OBGP+, Cost, and Cost+Kalman. (b) (Right) Percentage of RWA decisions that were taken using Kalman.

12.7 Conclusions on multi-domain optical networks

In this part of the thesis we have discussed about the lessons learned in the areas of inter-domain routing and TE control. We have argued in favor of the opportunity to change that future optical networks offer, and as a first step in

this direction we have proposed and tested a new route control model. This model blends the strengths of the ASON's Overlay model with those of the Peer model that is being developed for GMPLS networks in the framework of the IETF. We named it the Hybrid model.

We have introduced a fully distributed and physically independent control plane for the Hybrid model, supported by a new network component named IDRA. This latter is the one in charge of distributing routing and TE information between RCDs, and of making the RWA decisions within a RCD. The role of the IDRAs is comparable to that of the PCEs in IP/MPLS networks.

Furthermore, we have described the details of the information exchange between neighboring IDRAs, which essentially consists of network reachability information, path state information and TE services. Based on this, we have developed two RWA strategies for the IDRAs, namely, Cost and Cost+Kalman, and their performance was contrasted against OBGP and OBGP+. The latter is an enhanced version of OBGP that we also developed in this part of the thesis. The RWA strategies proposed here for the IDRAs inherit some of the best features of BGP, and at the same time, they drastically improve the tentative extension of BGP to optical networks.

Our evaluation results have also demonstrated the usefulness of developing simple models to roughly estimate the occupancy in multi-domain wavelength-routed optical networks, and then refine this estimation by means of predictive techniques. Our main results and conclusions apply to a rather small set of multi-domain optical networks, so further research is needed to analyze the feasibility of this kind of predictive approach in a large-scale environment. Estimation techniques like the one proposed here offer a promising line of work to address the trade-off between obtaining a low blocking ratio, and keeping the path state information exchanged between routing domains as limited as possible.

We hope that the results presented here encourage other researchers to devise novel ways of intergrating PSI in the inter-domain routing model without impacting on its capability to scale.

Part V

Conclusions and Future Work

Chapter 13

Conclusions

This thesis has studied different route control strategies for multi-domain networks, with special focus on the development of solutions aimed at improving the performance and reliability of inter-domain communications. We have analyzed the multi-domain route control problem in three different timeframes, namely, in the current IP-based context, in the near future IP/MPLS context, and in the future optical Internet.

We have shown that, at present, the introduction of cooperative and/or social route control models not only drastically reduces the penalties associated with frequent traffic relocations, but also supplies almost the same – and in several cases even better – end-to-end traffic performance, and this applies for different traffic loads in a multi-domain network.

For the near future, we have examined the existing limitations hindering the deployment of MPLS at the inter-domain level, and discussed about ways to solve them. In particular, we have broadened the IETF's concept of the PCE and proposed a distributed routing model for finding minimum-weight disjoint paths across multiple IP/MPLS domains. This route control strategy can work completely decoupled from the BGP protocol, supported precisely, by the PCE-based architecture.

Based on this, we have formulated and efficiently solved the problem of how a domain can maximize the MPLS coverage of its traffic demands with minimum cost, subject to a budget and network capacity constraints. The problem was formulated as a multi-objective integer program, and solved by means of a novel evolutionary algorithm that we proposed here, named *Maximum Coverage at minimum Cost* (MC^2). The most promising outcome of this part of our work is that the contributions are general in scope and can be applied in other problems. In particular, our proposals can be applied in settings where constrained problems considering maximum coverage vs. cost are critical, given that costs associated with concave pricing functions are widely used in practice.

Finally, we have discussed about the lessons learned in the areas of inter-domain routing and TE control, arguing in favor of exploiting the opportunity to change offered by future optical networks. As a first step in this

direction we have proposed and tested a new route control model supported by a dedicated and physically independent control plane. Our evaluation results have demonstrated that by computing, aggregating, and conveying even minor path state information through this novel control plane, it is possible to reduce almost two orders of magnitude the blocking obtained with an alternative RWA strategy like OBGP. We have also shown that these significant improvements can be achieved without needing to exchange more routing messages than with OBGP, since by decrementing the blocking, it is possible to reduce the exchange of network reachability messages and path exploration, which are typically triggered when blocking starts to occur.

Our evaluation results have shown as well the usefulness of developing simple models to roughly estimate the occupancy in multi-domain wavelength-routed optical networks, and then refine this estimation by predictive techniques.

In the next section, we outline some of the most important topics not covered by this thesis and that might be considered by other researchers as potential lines for future work.

Chapter 14

Future Work

14.1 The microview

14.1.1 At Present: in multi-domain IP networks

In the area of intelligent route control, four complex issues remain largely unsolved. First, a comprehensive understanding of the path switching dynamics carried out by IRC strategies is needed. It is necessary to deepen in the analysis and develop models characterizing the stochastic distribution of path shifts in a competitive IRC environment. This thesis has started to analyze the issue, but further research is mandatory, especially, in order to study the local and global stability aspects of IRC.

Indeed, the development of non-linear route control models guaranteeing the stability of IRC solutions is the second of the unsolved issues mentioned above. Studies like [42], [137], or this thesis, show that randomized techniques are effective in de-synchronizing route controllers when their measurement windows are sufficiently overlapped, but still, neither of these studies formally guarantees stability. Only after understanding and characterizing the distribution of *Path Shifts* (PS) in competitive IRC environments, it will be possible to formulate and study the local and global stability issues of competitive IRC.

The third open issue in IRC is the need to develop techniques to adaptively adjust the triggering condition R_{th} depending on traffic load on the network. Clearly, this can be modeled as a dynamic optimization problem, and a promising line of work is to seek for:

$$\frac{\partial PS}{\partial \dot{\tau}} = 0 \quad \wedge \quad \frac{\partial^2 PS}{\partial \dot{\tau}^2} > 0 \tag{14.1}$$

whereas $\tau = RTT$ in the standalone IRC model, or $\tau = OWD$ in the cooperative model.

The fourth and last open issue in IRC is to analyze the feasibility of coordinating – up to some extent – the route control decisions between domains

(see for example [37]), and particularly, the interactions between IRC and the usual TE actions carried out by ISPs. An initial work in this direction was recently proposed in [38].

14.1.2 Near Future: in IP/MPLS multi-domain networks

In the area of IP/MPLS multi-domain networks, three different issues need to be further investigated. First, it is necessary to devise heuristics relaxing the complexity associated with the computation of entire optimal disjoint paths directly from the source PCE. More scalable approaches aiming at finding "near-optimal" disjoint paths, supported by efficient algorithms and a compact representation of a multi-domain IP/MPLS network are still missing.

Second, it is necessary to evaluate the dynamics of the TE information exchange model proposed here, and particularly, the potential impact and limits of updating the minimum-weight paths among the PCEs.

And third, the extension of the reach of our evolutionary multi-objective algorithm (MC^2) to other areas and settings is expected. In fact, we are already working on its application in the framework of EuQoS [30], which is an European FP6 Integrated Project.

14.1.3 Future: in multi-domain optical networks

Our work in the area of multi-domain optical networks can be extended in several ways. First of all, the main results and conclusions presented here apply to a rather small multi-domain network. For instance, we have shown that the strategy of piggy-backing path state information updates on the Keepalive messages exchanged between the IDRAs, works very well on a multi-domain setting of the size of the PAN European network. Despite this, further research is needed to analyze the extension of our RWA algorithms to very large scale settings.

In particular, it is important to find the limits of both the update strategy and the Kalman-based estimation-correction technique. More specifically, to try to determine the breakpoints – both in the timescale and node scale – where the stochastic estimation of the wavelength occupancy along inter-domain paths starts showing an unrecoverable drift, i.e., the predicted occupancy is not better than a random guess. However, testing RWA strategies on a very large scale simulation environment (e.g. with thousands of domains) is not trivial at all, given the extremely large number of events that a simulator would need to process in order to determine the blocking ratios during the simulation runtime.

Another way to extend the work in this thesis is to deepen in the subject of the potential benefits of advertising nominal blocking ratios between domains. An interesting – and completely open – problem is to study the effects and propose solutions when non-cooperative transit domains report lower blocking ratios $BR_j(d)$ (see (12.8)) than those that they are actually experiencing. An appealing approach is to address the issue from the game theory viewpoint.

Another potential line of work is to study the possible application of entropy-based approaches to fine-tune the dynamic estimation of the Gaussian parameters in the Kalman filter.

Finally, one of the most promising ways to continue with the work started here is to develop some of the novel TE services for the IDRA-based model introduced in Section 11.2 (see Fig. (11.3)). In particular, to work in TE services supporting the exchange of information regarding the resilience, grooming, and wavelength conversion capabilities between routing domains.

14.2 The macroview: the big picture

In practical terms, carrying out research in multi-domain networks is at the end about two things: i) scalability; ii) understanding and modeling the interactions between large networked systems named ASs or domains. Surprisingly, we have not yet been able to characterize and formally describe any of them.

14.2.1 What exactly is scalability?

The literature is rich in terms of informally describing the meaning of scalability, but surprisingly, we lack of a closed definition that allows to univocally determine when a networked system scales, and when it does not. Informally, scalability refers to the ability of expanding a system to support a large number of users, ports or capabilities, without making major changes to system and without a major impact on its performance. What is missing is a formal abstraction of these concepts, and a way of quantifying them as the scale of a networked system grows.

Scalability is at present an intuitive conception about some properties of a system, strongly rooted in the experiences gained with well-known networked systems in the past. Despite this lack of correctness, it is widely used as a way to characterize systems, and the outcome of such characterization is almost binary in practice, i.e., a given system scales, which is a positive property, or it does not, which is a negative aspect stating that a wide deployment

of the system is not viable. It is remarkable to observe the rather binary way in which scalability is used in practice, even tough we lack of formal mechanisms to prove that a given system scales or not. We hope that this short discussion encourage other researchers to fill this gap.

14.2.2 Lack of constituent laws

During the last two decades we have witnessed the tremendous growth of the traffic on the Internet. During this process, the research and engineering communities have contributed in countless ways supporting the multi-dimensional expansion of the network. However, one fundamental piece is still missing in order to fully understand and model its machinery; we lack of constituent laws.

Even though we are now able to understand and model many different aspects of a network at a microscopic level – like the effects of queuing packets, or how to measure their OWD between two nodes – we still do not know how to treat an model a networked system at a macroscopic level. At present, we already need exabytes[1] as the unit to count the traffic on the Internet, and this unit keeps growing once every few years. The natural corollary is that in the coming years, it will become much harder, and clearly less useful, to model and understand microscopic aspects of a large networked system. Instead, we claim that understanding and modeling the macroscopic characteristics of a network should be part of our research agenda.

A similar process was faced by physicists in the past. Physicists had a large expertise in modeling the behavior of a single particle, or a rather small set of them, but not in the order of $1\text{x}10^{23}$ particles[2]. To tackle this issue, physicists developed the thermodynamic theory. The thermodynamic principles supplied the constituent laws supporting the analysis and modeling of thermomechanical phenomena at a macroscopic level, and most important of all, to anticipate and accurately control the effects of these phenomena.

Several analogies can be found between thermodynamic and networked systems. For example, if we analyze the basics of TE, we will found out that quite likely it should be possible to model its effects as an *energy* transfer, by dissipating heat from one or more overheated links to the rest of the network.

Another analogy is captured by Table 14.1. The figure in the first column shows the basics of first thermodynamic principle for open systems in a non-stationary state. The questions that arise are fundamentally the following.

---

[1] 1 exabyte = $1\text{x}10^{18}$ bytes.

[2] We recall that the Avogadro number is $N_A \simeq 6.02\text{x}10^{23}$ mol^{-1}.

Can researchers in computer science come up with an analogous theory, and fill the second column in Table 14.1?

Can we define similar concepts as the Temperature, Energy, Work, or Heat, in a networked system?

It is clear that finding expressions similar to those in Table 14.1 for networked systems, that is, constituent laws capturing the macroscopic properties of a network, would open many research paths, especially, in terms of anticipating and accurately controlling the effects of routing and TE actions. We plan to explore this path in the near future.

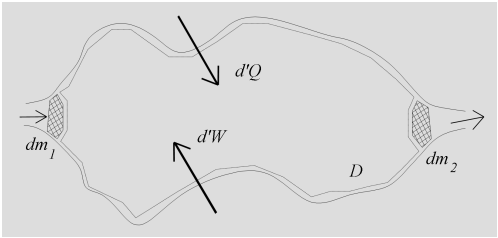
Thermodynamic Systems	Networked Systems
	
$\frac{\partial}{\partial t}\int\limits_{D}\rho e dV = \dot{Q}_{in} + \dot{W}_{ns} + \int\limits_{\partial D}\rho\left(h + \frac{u^2}{2} + gz\right)\overrightarrow{u}\,\overrightarrow{n}\,dA$	?
$\dot{M}(t) = \left(\sum\limits_{k}\dot{m}_k\right)_{in} - \left(\sum\limits_{j}\dot{m}_j\right)_{out}$	

Table 14.1: Thermodynamic analogy.

Bibliography

[1] S. Agarwal, C. Chuah, R. Katz, "OPCA: Robust Interdomain Policy Routing and Traffic Control," in IEEE Openarch, April 2003.

[2] S. Agarwal, A. Nucci, S. Bhattacharyya, "Controlling Hot Potatoes in Intradomain Traffic Engineering," SPRINT ATL Research Report RR04-ATL-070677, July 2004.

[3] A. Akella, B. Maggs, S. Seshan, A. Shaikh, R. Sitaraman, "A Measurement-Based Analysis of Multihoming," in Proceedings of ACM SIGCOMM 2003, Karlsruhe, Germany.

[4] A. Akella, J. Pang, B. Maggs, S. Seshan and A. Shaikh, "A Comparison of Overlay Routing and Multihoming Route Control," in Proceedings of ACM SIGCOMM, Portland, USA, August 2004.

[5] A. Akella, S Seshan, and A. Shaikh, "Multihoming Performance Benefits: An Experimental Evaluation of Practical Enterprise Strategies," USENIX Annual Technical Conference 2004, Boston, MA, USA.

[6] D. G. Andresen, H. Balakrishnan, M. F. Kaashoek, R. Morris, "Resilient Overlay Networks," in Proceedings, 18th of ACM SOSP, 2001.

[7] T. Anjali, C. Scoglio, J. de Oliveira, "New MPLS Network Management Techniques Based on Adaptive Learning," IEEE Transactions on Neural Networks, Vol. 16(5), pp. 1242-1255, September 2005.

[8] J. Ash, and J. L Le Roux, "PCE Communication Protocol Generic Requirements," IETF RFC4657, September 2006.

[9] R. Atkinson, Ed., S. Floyd, Ed., "IAB Concerns and Recommendations Regarding Internet Research and Evolution," RFC3869, August 2004.

[10] Avaya, Inc., "Converged Network Analyzer".

[11] A. L. Barabasi and R. Albert, "Emergence of scaling in random networks," Science journal 286:509-512, 1999.

[12] T. Bates, R. Chandra, and E. Chen, "Route Reflection - An Alternative to Full Mesh iBGP," RFC 2796, IETF, April 2000.

[13] G. Bernstein, D. Cheng, D. Pendarakis, et al, "Domain to Domain Routing using GMPLS, OSPF Extension V1.1(Draft)," OIF2002.23.06, July 2002.

[14] G. M. Bernstein, V. Sharma, L. Ong, "Inter-Domain optical routing," Journal of Optical Networking, Vol. 1, No. 2, February 2002.

[15] R. Bhandari, "Survivable Networks: Algorithms for Diverse Routing," Kluwer Academic Publishers, Norwell, MA, USA, 1999.

[16] M. Blanchet, F. Parent, B. St-Arnaud, "Optical BGP (OBGP): InterAS Lightpath Provisioning," IETF draft, ietf-draft-parent-obgp-01, March 2001.

[17] O. Bonaventure, C. de Launois, B. Quoitin, M. Yannuzzi, "Improving the quality of interdomain paths by using IP tunnels and the DNS," Technical-Report, Department of Computing Science and Engineering, Universite catholique de Louvain (UCL), Louvain-La-Neuve, Belgium, December 2004.

[18] A. Bremler-Barr, Y. Afek and S. Schwarz, "Improved BGP Convergence via Ghost Flushing," in Proceedings of IEEE INFOCOM, 2003.

[19] J. Chandrashekar, Z. Duan, Z. L. Zhang, and J. Krasky, "Limiting path exploration in BGP," in Proceedings of INFCOM, Miami, USA, 2005.

[20] R. K. C. Chang, M. Lo, "Inbound Traffic Engineering for Multihomed ASes Using AS Path Prepending," IEEE Network Magazine, March 2005.

[21] CIDR report, August 2007: `http://www.cidr-report.org/`.

[22] Cisco Systems, Inc., "Optimized Edge Routing".

[23] A. Collins, "The Detour Framework for Packet Rerouting," PhD Qualifying Examination, October 1998.

[24] Common Control and Measurement Plane (CCAMP) WG, IETF: `http://www.ietf.org/html.charters/ccamp-charter.html`.

[25] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, "Introduction to Algorithms," 2nd. ed. MIT Press, 2003.

[26] E. Crawley, R. Nair, B. Rajagopalan, H. Sandick, "A Framework for QoS-based Routing in the Internet," RFC 2386, IETF, August 1998.

[27] G. Cristallo, and C. Jacquenet, "An Approach to Inter-domain Traffic Engineering," in Proceedings of the XVIII World Telecommunications Congress (WTC2002), France, September 2002.

[28] B. S. Davie and Y. Rekhter, "MPLS: Technology and Applications," Morgan Kaufmann Series in Networking, ISBN 1558606564.

[29] Z. Duan, Z. L. Zhang, Y. T. Hou, "Service Overlay Networks: SLAs, QoS, and Bandwidth Provisioning," IEEE/ACM Transactions on Networking, Vol. 11, number 6, December 2003.

[30] End-to-end Quality of service support over heterogeneous networks (EuQoS), IST FP6-IP 004503, `http://www.euqos.org/`.

[31] J. W. Evans and C. Filsfils, "Deploying IP and MPLS QoS for Multiservice Networks: Theory & Practice," Morgan Kaufmann Series in Networking, ISBN 0123705495.

[32] M. Faloutsos, P. Faloutsos, C. Faloutsos, "On Power-Law Relationships of the Internet Topology," in Proceedings of ACM SIGCOMM, September 1999.

[33] A. Farrel, A. Satyanarayana, A. Iwata, N. Fujita, and G. Ash, "Crankback Signaling Extensions for MPLS and GMPLS RSVP-TE," draft-ietfccamp-crankback-05.txt, Internet Draft, work in progress, May 2005.

[34] A. Farrel, J. P. Vasseur, J. Ash, "A Path Computation Element (PCE)-Based Architecture," IETF RFC 4655, August 2006.

[35] A. Farrel, J. P. Vasseur, and A. Ayyangar "A Framework for Inter-Domain Multiprotocol Label Switching Traffic Engineering," IETF RFC 4726, November 2006.

[36] A. Feldmann, O. Maennel, Z. M. Mao, A. Berger, and B. Maggs "Locating Internet routing instabilities," in Proceedings ACM SIGCOMM, Portland, USA, September 2004.

[37] A. Fonte, E. Monteiro, M. Yannuzzi, X. Masip-Bruin, and J. Domingo-Pascual, "A Framework for Cooperative Interdomain QoS Routing," Springer Series: IFIP, Vol. 196, pp. 91 104, ISBN: 0-387-30815-6, March 2006.

[38] A. Fonte, M. Pedro, E. Monteiro, and F. Boavida, "Analysis of Interdomain Smart Routing and Traffic Engineering Interactions," in Proceedings of Globecom 2007, Washington DC, USA, November 2007.

[39] M. J. Francisco, L. Pezoulas, C. Huang, I. Lambadaris, "End-To-End Signaling and Routing for Optical IP Networks," in Proceedings of IEEE ICC, New York, USA, April 2002.

[40] Future INternet Design (FIND): `http://www.nets-find.net/`.

[41] Future Internet, "The Future Networked Society: A white paper from the EIFFEL Think-Tank," available from: `http://future-internet.eu/`.

[42] R. Gao, C. Dovrolis, E. W. Zegura, "Avoiding Oscillations due to Intelligent Route Control Systems," in Proceedings of INFOCOM 2006, Barcelona, Spain, April 2006.

[43] Global Environment for Network Innovations (GENI): `http://www.geni.net/`.

[44] D. K. Goldenberg, L. Qiu, H. Xie, Y. R. Yang, and Y. Zhang, "Optimizing cost and performance for multihoming," in Proceedings of ACM SIGCOMM, August 2004.

[45] T. Griffin and B. Presmore, "An Experimental Analysis of BGP convergence time," in Proceedings of IEEE ICNP, November 2001.

[46] T. G. Griffin, F. B. Shepherd, and G. Wilfong, "The Stable Paths Problem and Interdomain Routing," in IEEE/ACM Transactions on Networking, Volume 10, Issue 2, pp 232-243, April 2002.

[47] T. G. Griffin and G. T. Wilfong, "An analysis of BGP convergence properties," in Proceedings of SIGCOMM, pages 277-288, Cambridge, MA, August 1999.

[48] T. G. Griffin, and G. Wilfong, "On the correctness of iBGP configuration," in Proceedings of ACM SIGCOMM 2002, August 2002.

[49] F. Guo, J. Chen, W. Li, C., T. Chiueh, "Experiences in Building a Multihoming Load Balancing System," INFOCOM 2004, Hong Kong, China, March 2004.

[50] S. Halabi, "Internet Routing Architectures, 2nd Edition," Cisco Press, ISBN 1587054353.

[51] T. Hanne, "On the convergence of multiobjective evolutionary algorithms," European Journal of Op. Research, 117(3):553-564, 1999.

[52] C. Hedrick, "Routing Information Protocol," IETF RFC 1058, June 1988.

[53] T. Hiroyasu, M. Miki, S. Nakayama, and Y. Hanada, "Multi-Objective Optimization of Diesel Engine Emissions and Fuel Economy Using SPEA2+," in Hans-Georg Beyer et al. (editors), 2005 Genetic and Evolutionary Computation Conference (GECCO'2005), pp. 2195-2196, Vol. 2, ACM Press, New York, USA, June 2005.

[54] M. Howarth, et al , "End-to-end Quality of Service Provisioning through Inter-provider Traffic Engineering," Computer Communications, vol. 29, No. 6, pp683-702, March 2006.

[55] B. Huffaker, M. Fomenkov, D. J. Plummer, D. Moore, and k claffy, "Distance metrics in the Internet," in IEEE International Telecommunications Symposium, 2002.

[56] R. Hulsermann, A. Betker, M. Jager, S. Bodamer, M. Barry, J. Spath, C. Gauger, M. Kohn, "A Set of Typical Transport Network Scenarios for Network Modelling," ITG Fachbericht, issue 182, pages 65-72, 2004.

[57] G. Huston "Analyzing the Internet's BGP routing table," Internet Protocol Journal, vol. 4, N. 1, 2001.

[58] Internap Networks, Inc., "Flow Control Platform".

[59] International Telecommunications Union (ITU): `http://www.itu.int`.

[60] Internet Engineering Task Force (IETF): `http://www.ietf.org/`.

[61] IS-IS, IETF WG:
`http://www.ietf.org/html.charters/isis-charter.html`.

[62] ITU-T Recommendation G.8080, "Architecture for the Automatically Switched Optical Network (ASON)," November 2001.

[63] A. D. Jaggard, V. Ramachandran, "Towards the Design of Robust Inter-domain Routing Protocols," IEEE Network, Vol. 19, No. 6, November/December. 2005.

[64] J-Sim Homepage: `http://www.j-sim.org`.

[65] H. Kellerer, U. Pferschy, and D. Pisinger, "Knapsack Problems," Springer Verlag, ISBN 3-540-40286-1.

[66] A. Kolarov, A. Atai, and J. Hui, "Application of Kalman Filter in High-Speed Networks," in Proceedings of IEEE Globecom'94, San Francisco, USA, November 1994.

[67] F. A. Kuipers, "Quality of Service Routing in the Internet: Theory, Complexity and Algorithms," Ph.D. thesis, Delft University Press, The Netherlands, ISBN 90-407-2523-3, September 2004.

[68] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed Internet routing convergence," in Proceedings ACM SIGCOMM, 2000.

[69] C. Labovitz, A. Ahuja, F. Jahanian, "Experimental Study of Internet Stability and Backbone Failures," in Proceedings of FTCS-29, the 29th International Symposium on Fault-Tolerant Computing Madison, Wisconsin, pp. 278-285, June 1999.

[70] C. de Launois, "Unleashing Traffic Engineering for IPv6 Multihomed Sites," Doctoral Thesis, Computer Science and Engineering Department, Université catholique de Louvain, Belgium, September 2005.

[71] C. L. Li, T. McCormick, and D. Simchi-Levi, "The Complexity of Finding Two Disjoint Paths with Min-Max Objective Function," Discrete Applied Mathematics, 26:105-115, 1990.

[72] J. Li, A. Huang, J. Yao, I. Bitter, N. Petrick, R. M. Summers, P. J. Pickhardt, and J. R. Choi, "Automatic Colonic Ppolyp Detection using Multiobjective Evolutionary Techniques," Medical Imaging 2006: Image Processing. Ed. by J. M. Reinhardt, J. P. W. Pluim, in Proceedings of the SPIE, Vol. 6144, pp. 1742-1750, 2006.

[73] Z. Li, P. Mohapatra, "QRON: QoS-aware Routing in Overlay Networks," IEEE Journal on Selected Areas in Communications, June, 2003.

[74] G. Liu, C. Ji, V. Chan, "On the Scalability of Network Management Information for Inter-Domain Light-Path Assessment," IEEE/ACM ToN, Vol. 13, No. 1, February 2005.

[75] M. López-Ibáñez, T. D. Prasad, and B. Paechter, "Multi-Objective Optimisation of the Pump Scheduling Problem using SPEA2," in Proceedings of 2005 IEEE Congress on Evolutionary Computation (CEC'2005), pp. 435-442, Vol. 1, IEEE Service Center, Edinburgh, Scotland, September 2005.

[76] G. Malkin, "RIP Version 2 - Carrying Additional Information," IETF RFC 1723, November 1994.

[77] Z. M. Mao, R. Govindan, G. Varghese, and R. Katz, "Route Flap Damping Exacerbates Internet Routing Convergence," in Proceedings of ACM SIGCOMM, 2002.

[78] E. Marín-Tordera, X. Masip-Bruin, S. Sánchez-López, J. Solé-Pareta, and J. Domingo-Pascual, "A Hierarchical Routing Approach for Optical Transport Networks," Computer Networks, Vol. 50, No. 2, February 2006.

[79] E. Marín-Tordera, X. Masip-Bruin, S. Sánchez-López, J. Solé-Pareta, and J. Domingo-Pascual, "The Prediction-Based Routing in Optical Transport Networks," Computer Communications Journal, Elsevier, vol.19, issue 7, pp.865-878, April 2006.

[80] J. Marzo, E. Calle, C. Scoglio, T. Anjali, "QoS On-Line Routing and MPLS Multilevel Protection: a Survey," IEEE Communications Magazine, October 2003.

[81] X. Masip-Bruin, "Mechanisms to Reduce Routing Information Inaccuracy Effects: Application to MPLS and WDM Netwoks," Doctoral Thesis, Department of Computer Architecture, Technical University of Catalonia, Spain, October 2003.

[82] X. Masip-Bruin, M. Yannuzzi, J. Domingo-Pascual, A. Fonte, M. Curado, E. Monteiro, F. Kuipers, P. Van Mieghem, S. Avallone, G. Ventre,

P. Aranda-Gutierrez, M. Hollick, R. Steinmetz, L. Iannone, K. Salamatian, "Research Challenges in QoS Routing," Computer Communications, Elsevier, vol. 29, no. 5, March 2006.

[83] X. Masip-Bruin, M. Yannuzzi, R. Serral-Gracia, J. Domingo-Pascual, J. Enriquez-Gabeiras, M. Callejo, M. Diaz, F. Racaru, G. Stea, E. Mingozzi, A. Beben, W. Burakowski, E. Monteiro, L. Cordeiro, "The EuQoS System: A Solution for QoS Routing in Heterogeneous Networks," in IEEE Communications Magazine, 45(2):96-103, February 2007.

[84] MC^2: `http://www.ccaba.upc.edu/mc2-results`.

[85] A. Medina, A. Lakhina, I. Matta, and J. Byers. "BRITE: An Approach to Universal Topology Generation," in Proceedings of MASCOTS, August 2001.

[86] P. Morrissey, "Mapping Out the Best Route," NETWORK COMPUTING – MANHASSET NY – Vol. 14:(25), pp. 47-55, 2003.

[87] M. Morrow, V. Sharma, T. D. Nadeau, L. Andersson, "Challenges in Enabling Interprovider Service Quality in the Internet," IEEE Communications Magazine, vol. 43, no. 6, June 2005.

[88] T. K. Nayak, K. N. Sivarajan, "A New Approach to Dimensioning Optical Networks," IEEE JSAC, Vol. 20, No. 1, January 2002.

[89] N. Nedjah and L. D. M. Mourelle, "Pareto-optimal hardware for digital circuits using SPEA," in Proceedings of Innovations in Applied Artificial Intelligence, Springer-Verlag, Lecture Notes in Artificial Intelligence, Vol. 3533, pp. 594-604, 2005.

[90] OIF (Optical Internetworking Forum): `http://www.oiforum.com/`.

[91] OPNET Modules developed: `http://www.ccaba.upc.edu/opnet/`.

[92] OPNET Technologies, Inc:
`http://www.opnet.com/solutions/network_rd/modeler.html`.

[93] A. Orda and A. Sprintson, "Efficient Algorithms for Computing Disjoint QoS Paths," in Proceedings of IEEE INFOCOM, Hong Kong, March 2004.

[94] E. Osborne and A. Simha, "Traffic Engineering with MPLS," Cisco Press, ISBN 1587050315.

[95] OSPF, IETF WG:
`http://www.ietf.org/html.charters/ospf-charter.html`.

[96] P. P. Pan, E. L. Hahne, and H. G. Schulzrinne, "BGRP: Sink-Tree-Based Aggregation for Inter-Domain Reservations," Journal of Communications and Networks, Vol. 2, No. 2, 2000.

[97] Path Computation Element (PCE) WG, IETF:
`http://www.ietf.org/html.charters/pce-charter.html`.

[98] D. Pei, M. Azuma, D. Massey, and L. Zhang, "BGP-RCN: improving BGP convergence through root cause notification," Computer Networks, Volume 48, Issue 2, pp 175-194, 2005.

[99] C. Pelsser and O. Bonaventure, "Path Selection Techniques to Establish Constrained Interdomain MPLS LSPs," in Proceedings of Networking 2006, Coimbra, Portugal, May 2006.

[100] I. Pepelnjak, "MPLS and VPN Architectures," Cisco Press, ISBN 1587050811.

[101] B. Quoitin, "BGP-based Interdomain Traffic Engineering," Doctoral Thesis, Computer Science and Engineering Department, Université catholique de Louvain, Belgium, August 2006.

[102] B. Quoitin and O. Bonaventure, "A cooperative approach to interdomain traffic engineering," in Proceedings of NGI 2005, Rome, Italy, April 2005.

[103] B. Quoitin, C. Pelsser, L. Swinnen, O. Bonaventure, S. Uhlig, "Interdomain traffic engineering with BGP," IEEE Communications Magazine, 41(5):122-128, May 2003.

[104] B. Quoitin, S. Tandel, S. Uhlig, and O. Bonaventure, "Interdomain Traffic Engineering with Redistribution Communities," Computer Communications, 27(4), 2004.

[105] Y. Rekhter, "Constructing intra-AS path segments for an inter-AS path," ACM SIGCOMM Comput. Commun. 1991.

[106] Y. Rekhter, T. Li , "A Border Gateway Protocol 4 (BGP-4)," IETF, RFC 1771, March 1995.

[107] J. Rexford, J. Wang, Z. Xiao, and Y. Zhang, "BGP routing stability of popular destinations," Proceedings Internet Measurement Workshop, November 2002.

[108] F. Ricciato, U. Monaco, D. Ali, "Distributed Schemes for Diverse Path Computation in Multi-Domain MPLS Networks," IEEE Communications Magazine, vol. 43, n. 6, June 2005.

[109] R. Romeral and D. Larrabeiti, "Combining Border Router Policies for Disjoint LSP computation," in Proceedings of the V Workshop in G/MPLS Networks, Gerona, Spain, March 2006.

[110] S. Savage et al, "Detour: a Case for Informed Internet Routing and Transport," IEEE Micro, January, 1999.

[111] A. Shaikh, L. Kalampoukas, R. Dube, A. Varma, "Routing Stability in Congested Networks: Experimentation and Analysis," in Proceedings of ACM SIGCOMM, Stockholm, Sweden, August 2000.

[112] H. W. Sorenson (ed.), "Kalman Filtering: Theory and Application," IEEE Press, 1985.

[113] A. Sprintson, M. Yannuzzi, A. Orda, and X. Masip-Bruin, "Reliable Routing with QoS Guarantees for Multi-Domain IP/MPLS Networks," in Proceedings of IEEE INFOCOM 2007, Anchorage, Alaska, USA, May 2007.

[114] A. Sridharan, and S. B. Moon, and Christophe Diot, "On the Correlation between Route Dynamics and Routing Loops," in Proceedings of IMC, Miami, USA, October 2003.

[115] A. Sridharan, K. N. Sivarajan, "Blocking in All-Optical Networks," IEEE/ACM ToN, Vol. 12, No. 2, April 2004.

[116] L. Subramanian, S. Agarwal, J. Rexford, R. Katz, "Characterizing the Internet Hierarchy from Multiple Vantage Points," INFOCOM 2002, New York, NY, USA, June 2002.

[117] L. Subramanian, I. Stoica, H. Balakrishnan, R. H. Katz, "OverQoS: Offering Internet QoS using overlays," ACM SIGCOMM Computer Communications Review, vol. 33-1, January, 2003.

[118] J. Suurballe, "Disjoint Paths in a Network," Networks, vol. 4, no. 2,. pp. 125-145, 1974.

[119] J. Suurballe and R. Tarjan, "A Quick Method for Finding Shortest Pairs of Disjoint Paths," Networks, vol.14, pp. 325-336, 1984.

[120] T. Takeda, Y. Ikejiri, A. Farrel, J. P. Vasseur "Analysis of Inter-domain Label Switched Path (LSP) Recovery," Internet draft, draft-ietf-ccamp-inter-domain-recovery-analysis-01.txt, work in progress, July 2007.

[121] P. Traina, "Autonomous System Confederations for BGP," RFC 1965, IETF, June 1996.

[122] S. Uhlig, "Implications of Traffic Characteristics on Interdomain Traffic Engineering," Doctoral Thesis, Computer Science and Engineering Department, Université catholique de Louvain, Belgium, March 2004.

[123] S. Uhlig, V. Magnin, O. Bonaventure, C. Rapier and L. Deri, "Implications of the Topological Properties of Internet Traffic on Traffic Engineering," Proceedings of the 19th ACM Symposium on Applied Computing, Special Track on Computer Networks, Nicosia, Cyprus, March 2004.

[124] S. Uludag, K.S. Lui, K. Nahrstedt, and G. Brewster, "Comparative Analysis of Topology Aggregation Techniques and Approaches for the Scalability of QoS Routing," Technical Report TR05-010, DePaul University, Chicago, USA, May 2005.

[125] J. P. Vasseur, A. Ayyangar, and R. Zhang, "A Per-domain path computation method for establishing Inter-domain Traffic Engineering (TE) Label Switched Paths (LSPs)," Internet draft, draft-ietf-ccamp-inter-domain-pd-path-comp-05.txt, work in progress, April 2007.

[126] J. P. Vasseur and J. L. Le Roux, "Path Computation Element (PCE) communication Protocol (PCEP)," Internet draft, draft-ietf-pce-pcep-08.txt, work in progress, July 2007.

[127] C. Villamizar, R. Chandra, R. Govindan, "BGP Route Flap Damping," RFC 2439, IETF, November1998.

[128] D. Walton, A. Retana, E. Chen, "Advertisement of Multiple Paths in BGP," Internet draft, draft-walton-bgp-add-paths-04.txt, work in progress, August 2005.

[129] H. Wang, H. Xie, L. Qiu, A. Silberschatz, and Y. R. Yang, "Optimal ISP Subscription for Internet Multihoming: Algorithm Design and Implication Analysis," in Proceedings of IEEE INFOCOM 2005, Miami, FL, March 2005.

[130] L. Wang, et al, "A Novel OBGP-based mechanism for Lightpath Establishment in WDM Mesh Networks," in Proceedings of ECOC 2003, Rimini, Italy, September 2003.

[131] L. Wang H. Zhang L. Zheng, "Reducing the OBGP protection switching time in WDM mesh networks," in Proceedings of OFC, Anaheim, CA, USA, March 2006.

[132] B. Waxman, "Routing of Multipoint Connections," IEEE JSAC, December 1988.

[133] L. Xiao, K. Lui, J. Wang, K. Nahrstedt, "QoS extensions to BGP," ICNP2002, November 2002.

[134] L. Xiao, and K. Nahrstedt, "Reliability Models and Evaluation of Internal BGP Networks," in Proceedings of IEEE INFOCOM 2004, Hong Kong, China, March 2004.

[135] Y. R. Yang, H. Xie, H. Wang, A. Silberschatz, Y. Liu, L. E. Li, A. Krishnamurthy, "On Route Selection for Interdomain Traffic Engineering," IEEE Network, Vol. 19, No. 6, November/December. 2005.

[136] M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S.Sánchez-López, M. Curado, J. Domingo-Pascual, "A proposal for inter-domain QoS routing based on distributed overlay entities and QBGP," in Proceedings of QoFIS'04, LNCS 3266, pp. 257-267, Barcelona, Spain, October 2004.

[137] M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sánchez-López, M. Curado, and J. Domingo-Pascual, "On the Advantages of Cooperative and Social Smart Route Control," in Proceedings of IEEE 15th International Conference on Computer Communications and Networks (ICCCN'06), Washington DC, USA, October 2006.

[138] M. Yannuzzi and X. Masip-Bruin and O. Bonaventure, "Open Issues in Interdomain Routing: A Survey," IEEE Network, Vol. 19, No. 6, November/December. 2005.

[139] M. Yannuzzi, X. Masip-Bruin, S. Sánchez-López, J. Domingo-Pascual, A. Orda, and A. Sprintson, "On the Challenges of Establishing Disjoint QoS IP/MPLS paths across multiple domains," in IEEE Communications Magazine, 44(12):60-66, December 2006.

[140] M. Yannuzzi, X. Masip-Bruin, S. Sánchez-López, E. Marin Tordera, J. Solé-Pareta, and J. Domingo-Pascual, "Interdomain RWA based on stochastic estimation methods and adaptive filtering for optical networks," in Proceedings of IEEE Globecom 2006, San Francisco, USA, Nov./Dec. 2006.

[141] M. Yannuzzi, S. Sánchez-López, X. Masip-Bruin, J. Solé-Pareta and J. Domingo-Pascual, "A Combined Intra-Domain and Inter-Domain QoS Routing Model for Optical Networks," IFIP/IEEE ONDM 2005, Milan, Italy, February 2005.

[142] E. Zitzler, M. Laumanns, and S. Bleuler, "A Tutorial on Evolutionary Multi-objective Optimization," in X. Gandibleux et al., editors, Metaheuristics for Multi-objective Optimisation, Lecture Notes in Economics and Mathematical Systems. Springer, 2004.

[143] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multi-objective Optimization," in Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN 2001), pages 95-100. International Center for Numerical Methods in Engineering (CIMNE), 2002.

[144] E. Zitzler and L. Thiele, "Multiobjective Evolutionary Algorithms: A comparative case study and the strength Pareto approach," IEEE Transactions on Evolutionary Computation, Vol. 3, Issue 4, pp.:257-271, November 1999.

Appendix A: Publications

Main Publications

Journals

1. X. Masip-Bruin, M. Yannuzzi, R. Serral-Gracia, J. Domingo-Pascual, J. Enriquez-Gabeiras, M. Callejo, M. Diaz, F. Racaru, G. Stea, E. Mingozzi, A. Beben, W. Burakowski, E. Monteiro, L. Cordeiro, "The EuQoS System: A Solution for QoS Routing in Heterogeneous Networks," in IEEE Communications Magazine, 45(2):96-103, February 2007.

2. M. Yannuzzi, X. Masip-Bruin, S. Sanchez-Lopez, J. Domingo-Pascual, A. Orda, and A. Sprintson, "On the Challenges of Establishing Disjoint QoS IP/MPLS paths across multiple domains," in IEEE Communications Magazine, 44(12): 60-66, December 2006.

3. X. Masip-Bruin, M. Yannuzzi, J. Domingo-Pascual, A. Fonte, M. Curado, E. Monteiro, F. Kuipers, P. Van Mieghem, S. Avallone, G. Ventre, P. Aranda-Gutierrez, M. Hollick, R. Steinmetz, L. Iannone, K. Salamatian, "Research Challenges in QoS Routing," Computer Communications, Elsevier, vol. 29, no. 5, March 2006.

4. M. Yannuzzi and X. Masip-Bruin and O. Bonaventure, "Open Issues in Interdomain Routing: A Survey," IEEE Network, Vol. 19, No. 6, November/December. 2005.

Book Chapters

1. A. Fonte, E. Monteiro, M. Yannuzzi, X. Masip-Bruin, and J. Domingo-Pascual, "A Framework for Cooperative Interdomain QoS Routing," Springer Series: IFIP, Vol. 196, pp. 91 104, ISBN: 0-387-30815-6, March 2006.

Conferences

1. A. Sprintson, M. Yannuzzi, A. Orda, and X. Masip-Bruin, "Reliable Routing with QoS Guarantees for Multi-Domain IP/MPLS Networks," in Proceedings of IEEE INFOCOM 2007, Anchorage, Alaska, USA, May 2007.

2. M. Yannuzzi, X. Masip-Bruin, S. Sanchez-Lopez, E. Marin Tordera, J. Sole-Pareta, and J. Domingo-Pascual, "Interdomain RWA based on stochastic estimation methods and adaptive filtering for optical networks," in Proceedings of IEEE Globecom 2006, San Francisco, USA, Nov./Dec. 2006.

3. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sanchez-Lopez, M. Curado, and J. Domingo-Pascual, "On the Advantages of Cooperative and Social Smart Route Control," in Proceedings of IEEE 15th International Conference on Computer Communications and Networks (ICCCN'06), Washington DC, USA, October 2006.

4. M. Yannuzzi, S. Sanchez-Lopez, X. Masip-Bruin, J. Sole-Pareta, and J. Domingo-Pascual, "A combined intra-domain and inter-domain QoS routing model for optical networks," in Proceedings of the 9th Conference on Optical Network Design and Modelling (ONDM 2005), IEEE/IFIP, pp. 197-203, Milan, Italy, February 2005.

5. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sanchez-Lopez, M. Curado, J. Domingo-Pascual, "A proposal for inter-domain QoS routing based on distributed overlay entities and QBGP," in Proceedings of QoFIS'04, LNCS 3266, pp. 257-267, Barcelona, Spain, October 2004.

Complete list of Publications

1. A. Sprintson, M. Yannuzzi, A. Orda, and X. Masip-Bruin, "Reliable Routing with QoS Guarantees for Multi-Domain IP/MPLS Networks," in Proceedings of IEEE INFOCOM 2007, Anchorage, Alaska, USA, May 2007.

2. M. Yannuzzi, A. Sprintson, X. Masip-Bruin, A. Orda, S. Sanchez-Lopez, Rene Serral-Gracia, J. Sole-Pareta, J. Domingo-Pascual, "Towards an Efficient Computation of High Quality primary and backup paths in Multi-Domain IP/MPLS Networks," in Proceedings of VI Workshop in G/MPLS Networks, Gerona, Spain, April 2007.

3. X. Masip-Bruin, M. Yannuzzi, R. Serral-Gracia, J. Domingo-Pascual, J. Enriquez-Gabeiras, M. Callejo, M. Diaz, F. Racaru, G. Stea, E. Mingozzi, A. Beben, W. Burakowski, E. Monteiro, L. Cordeiro, "The EuQoS System: A Solution for QoS Routing in Heterogeneous Networks," in IEEE Communications Magazine, 45(2):96-103, February 2007.

4. M. Yannuzzi, X. Masip-Bruin, S. Sanchez-Lopez, J. Domingo-Pascual, A. Orda, and A. Sprintson, "On the Challenges of Establishing Disjoint QoS IP/MPLS paths across multiple domains," in IEEE Communications Magazine, 44(12): 60-66, December 2006.

5. M. Yannuzzi, X. Masip-Bruin, S. Sanchez-Lopez, E. Marin Tordera, J. Sole-Pareta, and J. Domingo-Pascual, "Interdomain RWA based on stochastic estimation methods and adaptive filtering for optical networks," in Proceedings of IEEE Globecom 2006, San Francisco, USA, Nov./Dec. 2006.

6. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sanchez-Lopez, M. Curado, and J. Domingo-Pascual, "On the Advantages of Cooperative and Social Smart Route Control," in Proceedings of IEEE 15th International Conference on Computer Communications and Networks (ICCCN'06), Washington DC, USA, October 2006.

7. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sanchez-Lopez, M. Curado, and J. Domingo-Pascual, "From Standalone to Collective Intelligent Route Control," in IEEE Infocom Student Workshop, Barcelona, Spain, April 2006.

8. X. Masip-Bruin, M. Yannuzzi, J. Domingo-Pascual, A. Fonte, M. Curado, E. Monteiro, F. Kuipers, P. Van Mieghem, S. Avallone, G. Ventre,

P. Aranda-Gutierrez, M. Hollick, R. Steinmetz, L. Iannone, K. Salamatian, "Research Challenges in QoS Routing," Computer Communications, Elsevier, vol. 29, no. 5, March 2006.

9. M. Yannuzzi, E. Marin Tordera, S. Sanchez-Lopez, X. Masip-Bruin, and J. F. Lobo Poyo, "Evaluation of a Combined Intra- and Interdomain Constraint-based Routing Model for Optical Networks," in Proceedings of V Workshop in G/MPLS Networks, ISBN 8493482390, Gerona, Spain, March 2006.

10. A. Fonte, E. Monteiro, M. Yannuzzi, X. Masip-Bruin, and J. Domingo-Pascual, "A Framework for Cooperative Interdomain QoS Routing," Springer Series: IFIP, Vol. 196, pp. 91 104, ISBN: 0-387-30815-6, March 2006.

11. M. Yannuzzi and X. Masip-Bruin and O. Bonaventure, "Open Issues in Interdomain Routing: A Survey," IEEE Network, Vol. 19, No. 6, November/December. 2005.

12. R. Romeral, M. Yannuzzi, D. Larrabeiti, X. Masip-Bruin, M. Urueña, "Multi-domain G/MPLS recovery paths using PCE," in Proceedings of the 10th European Conference on Networks and Optical Communications (NOC 2005), pp. 187-193, London, UK, July 2005.

13. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sanchez-Lopez, and J. Domingo-Pascual, "A self-adaptive QoS routing framework for multi-homed stub autonomous systems," in Proceedings of Eunice 2005, IFIP, pp. 241-247, Madrid, Spain, July 2005.

14. A. Fonte , E. Monteiro, M. Yannuzzi, X. Masip-Bruin, and J. Domingo-Pascual, "A cooperative approach for coordinated inter-domain QoSR decisions," in Proceedings of Eunice 2005, IFIP, pp. 133-137, Madrid, Spain, July 2005.

15. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sanchez-Lopez, M. Curado, and J. Domingo-Pascual, "Encaminamiento interdominio con calidad de servicio basado en overlay entities distribuidas y QBGP," in Novatica journal, no. 175, May/June 2005 (only available in Spanish).

16. M. Yannuzzi, X. Masip-Bruin, E. Monteiro, "Towards self-adaptive interdomain edge routing," in IEEE INFOCOM Student Workshop, Miami, USA, March 2005.

17. M. Yannuzzi, S. Sanchez-Lopez, X. Masip-Bruin, J. Sole-Pareta, and J. Domingo-Pascual, "A combined intra-domain and inter-domain QoS routing model for optical networks," in Proceedings of the 9th Conference on Optical Network Design and Modelling (ONDM 2005), IEEE/ IFIP, pp. 197-203, Milan, Italy, February 2005.

18. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sanchez-Lopez, M. Curado, J. Domingo-Pascual, and J. Sole-Pareta, "Encaminamiento inter-dominio con calidad de servicio basado en una arquitectura overlay y QBGP," XIV Jornadas de Telecom I+D, Madrid, Spain, November 2004 (only available in Spanish).

19. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sanchez-Lopez, M. Curado, J. Domingo-Pascual, "A proposal for inter-domain QoS routing based on distributed overlay entities and QBGP," in Proceedings of QoFIS'04, LNCS 3266, pp. 257-267, Barcelona, Spain, October 2004.

20. M. Yannuzzi, E. Marin-Tordera, X. Masip-Bruin, J. Domingo-Pascual, S. Sanchez-Lopez, and J. Sole-Pareta, "Virtual Private LAN Service: The new challenge for LAN/WAN connectivity," in Proceedings of the III Workshop in MPLS Networks, pp. 109-125, Gerona, Spain, March 2004.

Research Reports

21. A. Sprintson, M. Yannuzzi, A. Orda, and X. Masip, "Reliable Routing with QoS Guarantees for Multi-Domain IP/MPLS Networks," UPC-DAC-RR-CBA-2006-4, Department of Computer Architecture, Technical University of Catalonia (UPC), Barcelona, Spain, September 2006.

22. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sachez-Lopez, M. Curado, and J. Domingo-Pascual, "An Incremental QoS Routing Framework for Multihomed Stub ASs based on Distributed Overlay Entities and QBGP," UPC-DAC-RR-CBA-2005-6, Department of Computer Architecture, Technical University of Catalonia (UPC), Barcelona, Spain, December 2005.

23. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sachez-Lopez, M. Curado, and J. Domingo-Pascual, "Self-Adaptive Routing: The Inevitable Evolution of Interdomain Route Optimization Tools," UPC-DAC-RR-CBA-2005-7, Department of Computer Architecture, Technical University of Catalonia (UPC), Barcelona, Spain, December 2005.

24. M. Yannuzzi, A. Fonte, X. Masip-Bruin, E. Monteiro, S. Sachez-Lopez, M. Curado, and J. Domingo-Pascual, "A Self-adaptive Interdomain Traffic Engineering Scheme," UPC-DAC-RR- CBA-2005-8, Department of Computer Architecture, Technical University of Catalonia (UPC), Barcelona, Spain, December 2005.

25. O. Bonaventure, C. de Launois, B. Quoitin, M. Yannuzzi, "Improving the quality of interdomain paths by using IP tunnels and the DNS," Technical-Report, Department of Computing Science and Engineering, Universite catholique de Louvain (UCL), Louvain la Neuve, Belgium, December 2004.

Appendix B: European Projects

Part of the material in this thesis has been used in the following European Research Projects:

(i) **E-NEXT:** Emerging Networking Experiments and Technologies, FP6-506869, European Network of Excellence (NoE) (2004-2005).

(ii) **NOBEL I:** Next generation Optical networks for Broadband European Leadership, FP6-506760, European Integrated Project (IP) (2004-2005).

(iii) **EuQoS:** End-to-end Quality of Service support over heterogeneous networks, FP6-004503, European Integrated Project (IP) (2004-2007).

(iv) **CONTENT:** Excellence in Content Distribution Network Research, FP6-0384239, European Network of Excellence (NoE) (2006-2009).